

T-VES
1506
G216m
1997
Ej. 2

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA QUIMICA



Modelamiento Matemático y Simulación de Procesos
para Sistemas de Tratamiento de Aguas. 1a. PARTE.

PRESENTADO POR:

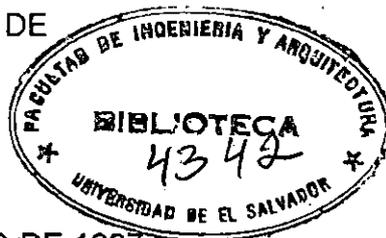
SAMUEL GARCIA FIGUEROA

CARMEN ALICIA HIDALGO ESCOBAR

15101010

15101010

PARA OPTAR AL TITULO DE
INGENIERO QUIMICO



CIUDAD UNIVERSITARIA, JUNIO DE 1997



UNIVERSIDAD DE ELSALVADOR

RECTOR : DR. BENJAMIN LOPEZ GUILLEN

SECRETARIO GENERAL : LIC. ENNIO ARTURO LUNA

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO : ING. JOAQUIN ALBERTO VANEGAS AGUILAR

SECRETARIO : ING. JOSE RIGOBERTO MURILLO CAMPOS

ESCUELA DE INGENIERIA QUIMICA

DIRECTORA : ING. DELMY DEL CARMEN RICO PEÑA

A handwritten signature in black ink, appearing to read "Delmy del Carmen Rico Peña".



UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERIA Y ARQUITECTURA
ESCUELA DE INGENIERIA QUIMICA

Trabajo de Graduación previo a la opción al grado de:
INGENIERO QUIMICO

Título:

**Modelamiento Matemático y Simulación de Procesos
para Sistemas de Tratamiento de Aguas. 1a. PARTE.**

Presentado por:

SAMUEL GARCIA FIGUEROA
CARMEN ALICIA HIDALGO ESCOBAR

Trabajo de Graduación aprobado por:

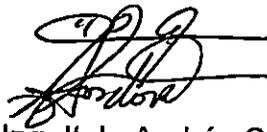
Coordinador: Ing. Italo Andrés Córdova Flamenco

Asesor: Ing. Fernando Teodoro Ramírez Zelaya

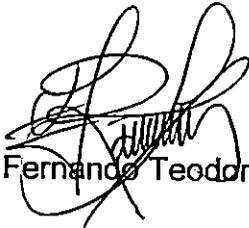
Asesor: Ing. Mario Enrique Estrada Muñoz

SAN SALVADOR, JUNIO DE 1997.

Trabajo de Graduación aprobado por:



Coordinador : Ing. Italo Andrés Córdova Flamenco



Asesor : Ing. Fernando Teodoro Ramírez Zelaya



Asesor : Ing. Marié Enrique Estrada Muñoz



AGRADECIMIENTOS

Deseamos expresar los más sinceros agradecimientos al Coordinador Ing. Italo Andrés Córdova Flamenco y a los asesores Ing. Fernando Teodoro Ramírez Zelaya e Ing. Mario Enrique Estrada Muñoz, por el valioso aporte a la realización del presente trabajo.

Así mismo, agradecemos a las siguientes personas por el apoyo y colaboración brindada en el desarrollo de este trabajo de graduación:

Lic. Ernesto Iraheta

Ing. Delmy del Carmen Rico Peña

Ing. Eugenia Gamero

Ing. Tania Torres Rivera

Ing. Daniel Mejía

Ing. Juan Rodolfo Ramírez

Ing. Oscar Lemus

Ing. Lucía Antonieta Sosa

Sr. Juan Rodolfo Mendoza

Sr. Oscar Alfredo Morán

Br. Ana Mélida Chicas

Br. Marlon Castro

Br. Gustavo Berríos

Br. Héctor Salomón Chávez

Br. Jesús Adilman Pérez

Br. José Eduardo López

Br. Juan Antonio Méndez Calixto

DEDICATORIA

A DIOS TODOPODEROSO : Por haberme permitido alcanzar este triunfo, dándome todas las fuerzas en los momentos más arduos.

A MIS PADRES : Julio y Alicia con mucho amor por sus sacrificios, por sus buenos consejos, por su apoyo incondicional durante toda mi vida y por tener siempre la confianza en mí.

A MIS HERMANOS : Julio Alberto y Gloria Mabel, por darme siempre ánimos para salir adelante, por su ayuda y cariño que me dan el deseo logren sus metas en la vida.

A MI ESPOSO : Mauricio gracias por su amor, comprensión y ayuda que me ha brindado.

A MIS HIJOS : Erick Mauricio y Carmen Geraldina, por ser la energía que me ha llevado a concluir esta meta. Los quiero mucho.

A TODA MI FAMILIA : Por su cariño y apoyo.

A MIS AMIGOS : Por la ayuda brindada durante el trayecto de mis estudios y por todos los momentos que hemos compartido.

Carmen Alicia

RESUMEN

El campo del tratamiento de aguas residuales ha experimentado en las últimas décadas, importantes avances exigidos por los continuos cambios en las características de las aguas residuales, debido a una diversa cantidad de sustancias que llevan consigo.

Con el fin de satisfacer los requerimientos de la calidad del agua para los diferentes usos, se han desarrollado muchos sistemas de tratamiento de aguas. Por ello, el hombre se ha preocupado por buscar métodos de tratamiento según el tipo de efluente para evitar el deterioro del medio ambiente.

Los sistemas de tratamiento de aguas industriales y residuales, requieren de abundante información y pruebas de cálculo; por lo que se hace necesario el uso de la computación como medio para facilitar el procesamiento de los datos.

El objetivo del presente trabajo de graduación es la elaboración de programas en lenguaje C que permitan simular los siguientes sistemas de tratamiento de aguas: sedimentación, lagunas de estabilización, adsorción en carbón activado e intercambio iónico.

Para la programación del software de los sistemas de tratamiento de aguas anteriores, se siguieron algunos pasos: 1) recolectar la información necesaria de los sistemas de tratamiento; 2) desarrollar un algoritmo de solución; 3) realizar el modelamiento matemático; 4) elaborar flujogramas para los procesos y 5) elaborar los programas en C .

Los capítulos incluidos en este trabajo presentan la secuencia de tratamiento; así se tiene la sedimentación como el tratamiento primario en que se emplean operaciones de tipo físico . Este sistema se usa para eliminar sólidos en flotación y sedimentables que se encuentran en el agua residual; como tratamiento secundario, se tienen las lagunas de estabilización que utilizan procesos biológicos para eliminar la materia orgánica.

Los tratamientos terciarios ó tratamientos avanzados son los que se utilizan para eliminar contaminantes que no han sido afectados por el tratamiento primario y secundario; ejemplos de estos están la adsorción en carbón activado e intercambio iónico. Debe hacerse notar que los vocablos primario y secundario son alternativos.

Se presenta en este trabajo el manual del usuario, donde se encontrarán los pasos para operar el programa. Aquí, el usuario podrá obtener una respuesta viable al problema que enfrente con respecto a parámetros de diseño de los sistemas de tratamiento de aguas que aquí se han considerado.

Finalmente, se encuentran conclusiones y recomendaciones referente al trabajo y referencias consultadas.

INDICE

Contenido	Página
INTRODUCCION	xvi
I. SEDIMENTACION	1
1.1 Generalidades	1
1.2 Tipos de Sedimentación	2
1.3 Sedimentación Tipo I	2
1.4 Sedimentación Tipo 2	4
1.5 Sedimentación Tipo 3 y 4	6
1.5.1 Cálculo de un Espesador Continuo	8
1.5.2 Cálculos de Diseño	9
1.5.3 Area del espesador	12
1.5.4 Profundidad del espesador	17
1.6 Planteamiento de lo que aportará la codificación del caso	17
1.6.1 Diseño del sedimentador	17
1.6.2 Definición de variables	18
1.6.3 Solucion de modelos	18
1.6.4 Cálculo de la ecuación de la bisectriz	19
1.6.5 Modelos para la parte no lineal	19
1.6.6 Primeras derivadas	20
1.6.7 Ecuaciones paramétricas	20
1.6.8 Elección del modelo	21
1.6.9 Profundidad de la zona de compresión	22
1.6.10 Cálculo del tiempo final de sedimentación	22
1.6.11 Determinación del área mínima de sedimentación	22
1.6.12 Cálculo de la altura	23
1.7 Ejemplo de cálculo para un sedimentador	25
1.7.1 Determinación del punto crítico de sedimentación	29
1.8 Programa de sedimentación	36
II. TRATAMIENTOS BIOLÓGICOS. LAGUNAS DE ESTABILIZACION.	52
2.1 Generalidades	52
2.1.1 Reseña Histórica	53
2.2 Características y estabilización biológica de las aguas	54
2.2.1 Naturaleza de la contaminación	54
2.2.2 Características de los residuos	54
2.2.3 Modo de descomposición	55
2.2.4 Reacciones biológicas	56
2.2.5 Demanda de Oxígeno	57
2.2.6 Determinación de la Demanda Bioquímica de Oxígeno.	58
2.2.7 Determinación de la Demanda Química de Oxígeno	60
2.2.8 Nitrificación	60

2.3	Clasificación de las lagunas.	61
2.4	Consideraciones sobre el diseño del proceso.	62
2.4.1	Lagunas Aerobias.	64
2.4.2	Lagunas Facultativas.	72
2.4.3	Lagunas Anaerobias.	74
2.5	Aplicación de estanques de estabilización.	76
2.6	Ejemplos de cálculo de lagunas de estabilización.	79
2.7	Planteamiento de lo que aportará la codificación del caso.	88
2.7.1	Definición de variables.	89
2.7.2	Solución de modelos.	91
2.7.3	Lagunas aerobias.	92
2.7.4	Lagunas aerobias - anaerobias.	97
2.7.5	Lagunas anaerobias.	99
2.8	Programas para lagunas de estabilización en Quick C.	108
III.	ADSORCION	132
3.1	Generalidades.	132
3.2	Uso de carbón activado y aplicaciones.	135
3.3	Adsorción.	136
3.3.1	Adsorción de colores y sabores.	140
3.4	Principios del proceso de adsorción.	140
3.4.1	Regeneración del carbón activado.	142
3.5	Ejemplos de cálculo en la adsorción de carbón activado.	143
3.6	Planteamiento de lo que aportará la codificación del caso.	148
3.6.1	Definición variables para la codificación.	149
3.7	Solución de modelos.	150
3.7.1	Ecuación de la isoterma de Freundlich.	150
3.7.2	Ecuación de la isoterma de Langmuir.	151
3.8	Programa para la resolución de ecuaciones en adsorción en carbón activado.	159
IV.	INTERCAMBIO IONICO.	173
4.0	Introducción.	173
4.1	Principios del Intercambio Ionico.	173
4.1.1	Definición de Intercambio Iónico.	173
4.1.2	Propiedades físicas y químicas de las resinas de intercambio iónico.	175
4.2	La Desmineralización.	175
4.2.1	Principios de la desmineralización total.	175
4.3	Dimensionamiento de un sistema desmineralizador.	178
4.3.1	Cálculos para determinar el tamaño del desmineralizador de lecho doble.	179
4.3.2	Datos técnicos de las resinas.	179
4.3.2.1	Resina catiónica.	179
4.3.2.2	Resina aniónica.	179

4.4	Guía de dimensionamiento de desmineralizadores.	182
4.4.1	Calidad del agua.	182
4.4.2	Capacidad.	184
4.4.3	Cálculos para la unidad catiónica.	184
4.4.4	Cálculos para la unidad aniónica.	186
4.5	Determinación de Silice.	189
4.6	Ejemplo de cálculo.	193
4.6.1	Procedimiento para el cálculo de un sedimentador.	194
4.6.1.1	Determinación de la unidad catiónica.	195
4.6.1.2	Determinación de la unidad aniónica.	196
4.7	Planteamiento de lo que aportará la codificación del caso.	198
4.7.1	Solución de modelos.	199
4.7.2	Cálculo para la unidad catiónica.	199
4.7.3	Cálculo para la unidad aniónica.	206
4.8	Programa de intercambio ionico en Quick C.	215
V.	MANUAL DEL USUARIO.	241
	Anexos para Intercambio Iónico.	259
	CONCLUSIONES Y RECOMENDACIONES.	263
	REFERENCIAS BIBLIOGRAFICAS.	266

INDICE DE FIGURAS

Num. Figura	Página
1.1 REGIMEN DE DESPLAZAMIENTOS	5
1.2 SEDIMENTACIÓN INTERMITENTE.	7
1.3 DIAGRAMA ESQUEMATICO DE UN ESPESADOR.	7
1.4 RESULTADOS DE UNA SEDIMENTACION INTERMITENTE.	8
1.5 CAPA LIMITANTE DE LA VELOCIDAD.	10
1.6 RESULTADOS DE UNA SEDIMENTACION INTERMITENTE.	12
1.7 FLUJO DE SOLIDOS PARA UN ESPESADOR CONTINUO.	14
1.8 DETERMINACION DE LA CONCENTRACION CRITICA.	15
2.1 REPRESENTACION ESQUEMATICA DE UNA LAGUNA DE ESTABILIZACION DE AGUAS RESIDUALES.	57
2.2 DESARROLLO DE LA DEMANDA BIOQUIMICA DE OXIGENO. ...	58
2.3 RELACION ENTRE LA CARGA DE DBO Y LA PROFUNDIDAD. ...	64
2.4 RELACION ENTRE K_t Y PORCENTAJE DE SUSTRATO.	70
3.1 ESQUEMA DE LA FORMA GENERALIZADA DE LA ISOTERMA DE LANGMUIR.	139
3.2 COLUMNA TIPICA DE CARBON ACTIVADO.	142
3.3 CALCULO DE CONSTANTES EMPIRICAS PARA LA ECUACION DE FREUNDLICH.	144
3.4 CALCULO DE CONSTANTES EMPIRICAS.	146
3.5 GRAFICA DE LOS DATOS DE ADSORCION DE LANGMUIR. ...	147
4.1 CAPACIDAD DE RESINA CATIONICA.	185
4.2 FUGA DE SODIO.	187
4.3 NIVELES DE REGENERACION.	189

4.4	CONDUCTIVIDAD ESPECIFICA.....	191
4.5	FUGA DE SILICE.....	192
4.6	FACTOR DE CORRECCION DE SILICE DEBIDO AL SODIO. . .	193
4.7	CAPACIDAD DE LA RESINA CATIONICA A 0% DE ALCALINI- DAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 24-60%.....	200
4.8	CAPACIDAD DE LA RESINA CATIONICA A 0% DE ALCALINI- DAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 60-100%.....	201
4.9	CAPACIDAD DE LA RESINA CATIONICA A 25% DE ALCALINI- DAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 0-24%.....	202
4.10	CAPACIDAD DE LA RESINA CATIONICA A 50% DE ALCALINI- DAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 20-100%.....	203
4.11	CAPACIDAD DE LA RESINA CATIONICA A 75% DE ALCALINI- DAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 20-100%.....	204
4.12	CAPACIDAD DE LA RESINA CATIONICA A 100% DE ALCALINI- DAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 20-100%.....	205
4.13	FACTOR DE CORRECCION PARA LA FUGA DE SILICE DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION DE 4 LIBRAS.....	209
4.14	FACTOR DE CORRECCION PARA LA FUGA DE SILICE DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION DE 6 LIBRAS Y UN RANGO DE OPERACION DE 0-4 ppm DE FUGA DE SODIO.....	210
4.15	FACTOR DE CORRECCION PARA LA FUGA DE SILICE DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION DE 6 LIBRAS Y UN RANGO DE OPERACION DE 4-8 ppm DE FUGA DE SODIO.....	211

- 4.16 FACTOR DE CORRECCION PARA LA FUGA DE SILICE
DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION
DE 8 LIBRAS Y UN RANGO DE OPERACION DE 0-4 ppm
DE FUGA DE SODIO. 212
- 4.17 FACTOR DE CORRECCION PARA LA FUGA DE SILICE
DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION
DE 8 LIBRAS Y UN RANGO DE OPERACION DE 4-8 ppm
DE FUGA DE SODIO. 213

INDICE DE TABLAS

Num. tabla	Página
1.1 SUMATORIA PARA EL CALCULO a_1 Y b_1	26
1.2 SUMATORIA PARA EL CALCULO a_2 Y b_2	27
1.3 DATOS PARA EL CALCULO DE LA ECUACION DE LA CURVA. .	27
1.4 COEFICIENTES DE DETERMINACION PARA LOS MODELOS SELECCIONADOS.	28
2.1 LAGUNAS DE ESTABILIZACION - CRITERIO DE DISEÑO.	63
2.2 ABREVIATURAS DE LAS ESTACIONES, SUS ELEVACIONES Y COORDENADAS.	66
2.3 PROMEDIOS MENSUALES Y ANUALES DE RADIACION GLOBAL $\text{Cal x cm}^2 \text{ x día}$	67
2.4 COEFICIENTES ACTIVIDAD - TEMPERATURA PARA DISTINTOS PROCESOS BIOLÓGICOS.	69
2.5 APLICACION DE DIVERSOS TIPOS DE ESTANQUES DE ESTABILIZACION Y SISTEMAS DE ESTANQUES.	77
2.5.1 CARACTERISTICAS DEL EFLUENTE DE DIVERSOS TIPOS DE ESTANQUES DE ESTABILIZACION.	78
2.6 FACTOR DE DISPERSION PARA UNA LAGUNA DE ESTABILIZACION.	96
3.1 MATERIA PRIMA QUE HA SIDO ESTUDIADA PARA LA PRODUCCION DE CARBON ACTIVADO.	134
3.2 CARACTERISTICAS DE CARBON ACTIVADO.	134
3.3 CARBONES EMPLEADOS EN EL TRATAMIENTO DE AGUAS. .	135
4.1 ALGUNOS TIPOS DE RESINAS ANIONICAS DE BASE FUERTE DISPONIBLES COMERCIALMENTE.	176
4.2 ALGUNOS TIPOS DE RESINA DE ACIDO FUERTE DISPONIBLES COMERCIALMENTE.	177

4.3	CARACTERISTICAS FISICAS DE LA RESINA CATIONICA DE ACIDO FUERTE.	180
4.4	CONDICIONES DE OPERACION SUGERIDAS PARA LA RESINA CATIONICA DE ACIDO FUERTE.	180
4.5	PROPIEDADES FISICAS Y QUIMICAS DE LA RESINA ANIONICA DE BASE FUERTE.	181
4.6	CONDICIONES DE OPERACION SUGERIDOS PARA LA RESINA ANIONICA FUERTEMENTE BASICA.	181
4.7	EJEMPLO DE ANALISIS FISICOQUIMICOS PARA EL DIMENSIONAMIENTO DE SISTEMAS DESMINERALIZADORES.	183
4.8	DATOS DE CALIDAD PARA UN AGUA DESMINERALIZADA EN BASE A LAS FUGAS ESPERADAS.	183
4.9	FACTORES DE CONVERSION PARA LA CAPACIDAD DE RESINA ANIONICA DE BASE FUERTE.	190
4.10	CORRECCION DEL pH.	192
4.11	ANALISIS FISICOQUIMICO DEL AGUA CRUDA SUMINISTRADA POR ANDA A LAS INSTALACIONES DE LA PLANTA PILOTO. ...	194

INDICE DE DIAGRAMAS

Num. Diagrama	Página
1.A	CALCULO DE LA ECUACION DE LA BISECTRIZ. 33
1.B	PROCEDIMIENTO DE CALCULO PARA LA RECTA TANGENTE. . 34
1.C	CALCULO DEL AREA DE UN SEDIMENTADOR. 35
2.A	DETERMINACION DE PRODUCCION DE OXIGENO Y CELULAS DE ALGAS EN LAGUNAS AEROBIAS. 101
2.B	DETERMINACION DEL PORCENTAJE DE SUSTRATO REMANENTE EN LAGUNAS AEROBIAS. 102
2.C	PARAMETROS DE DISEÑO PARA LAGUNAS AEROBIAS. 103
2.D	DETERMINACION DE AIREACION. 104
2.E	DETERMINACION DE LA DBO EN UNA LAGUNA FACULTATIVA 105
2.F	PARAMETROS DE DISEÑO PARA LAGUNAS FACULTATIVAS. . 106
2.G	PARAMETROS DE DISEÑO PARA LAGUNAS ANAEROBIAS. . . . 107
3.A	DETERMINACION DE CONSTANTES EMPIRICAS EN LA ECUACION DE FREUNDLICH (CASO 1). 154
3B	DETERMINACION DE CONSTANTES EMPIRICAS EN LA ECUACION DE FREUNDLICH (CASO 2). 155
3.C	RESOLUCION MATRICIAL POR EL METODO DE CRAMER (FREUNDLICH). 156
3.D	DETERMINACION DE CONSTANTES EMPIRICAS EN LA ECUACION DE LANGMUIR 157
3.E	RESOLUCION MATRICIAL POR EL METODO DE CRAMER (CRAMER). 158

INTRODUCCIÓN

El agua residual recogida de pueblos y ciudades, debe devolverse a la tierra ó a las aguas de nuestro planeta ; por lo que se debe de eliminar los contaminantes ó la presencia de sustancias extrañas contenidas en el agua residual, para la protección del medio ambiente.

En la actualidad, la mayoría de los sistemas de tratamiento de aguas están siendo investigados intensamente desde el punto de ejecución y aplicación. Es por eso, que al diseñar una planta de tratamiento de aguas es importante conocer el tipo de sustancias que contiene el agua , para poder hacer una adecuada selección de procesos y operaciones unitarias que han de considerarse en el sistema.

Uno de los propósitos de este trabajo de graduación es presentar algunos sistemas de tratamiento de aguas mediante los cuales es posible obtener agua de la calidad deseada. Los sistemas de tratamiento de aguas que se exponen en el trabajo son : sedimentación, lagunas de estabilización, adsorción en carbón activado e intercambio iónico.

La sedimentación consiste en la eliminación de partículas de mayor tamaño y en la reducción de la concentración de sólidos en suspensión para poder obtener un efluente clarificado.

Las lagunas de estabilización, tienen como finalidad el tratamiento biológico del agua residual en un estanque de configuración controlada. Los residuos orgánicos descomponibles son estabilizados por microorganismos debido a un período de detención requerido.

El tratamiento de agua residual con carbón activado, se considera como un proceso de acabado del agua ya tratada por procesos biológicos y se utiliza para eliminar olor, color, compuestos tóxicos, materia orgánica residual, etc.

La operación de intercambio iónico comprende el intercambio entre los iones contaminantes de una solución y los iones de un sólido (resina) ; se aplica tanto para el ablandamiento del agua como para la desmineralización.

Para cubrir el amplio campo de práctica de la ingeniería, el propósito principal del presente trabajo de graduación es la elaboración de programas de computadora aplicando el lenguaje C de programación para el manejo de la información.

El C es un lenguaje de programación de empleo general, es muy eficaz y conveniente para muchas tareas; en este caso se especifican variables para el diseño de procesos para sistemas de tratamiento de aguas.

CAPITULO I

SEDIMENTACION

1.1 GENERALIDADES

Las aguas residuales contienen una serie de partículas en suspensión: arenas, particularmente en sistemas mixtos de alcantarillado; sólidos orgánicos suspendidos y coloidales, sedimentables por si mismos o mediante su coagulación-floculación; y materia flotante, sólida o líquida que al no haber sido removida previamente en el sistema (por rejillas, cedazos o micromallas), se requiere separar del agua residual para obtener un cierto grado de tratamiento tendiente a reducir su carácter ofensivo para con las unidades posteriores de los mismos procesos de acondicionamiento y para reducir, básicamente, la demanda bioquímica de oxígeno que acusan esas materias precedentes.

Particularmente, para el caso que nos ocupa de líquidos residuales, la separación de sólidos más pesados que el agua se obtiene mediante una sedimentación; obteniéndose suspensiones o emulsiones más concentradas que deben ser evacuadas de las unidades separadoras.

Los principios de asentamiento vertical de partículas más pesadas que el líquido que las contiene, serán expuestos, a través de formulaciones clásicas establecidas.

Estos principios y formulaciones permiten entender, a la vez que aplicar, en forma más racional, las operaciones envueltas a través de la utilización de las cámaras desarenadoras (remoción por decantación de arenas y otros sólidos de densidad notoriamente mayor a la del líquido en proceso de tratamiento), de los estanques sedimentadores, clarificadores y densificadores (remoción por sedimentación de sólidos suspendidos sedimentables de naturaleza floculante y de densidades mayores que la del agua, pero menores que las de las arenas), y los estanques desnataores o desgrasadores (remoción de partículas sólidas o líquidas de densidades menores a las del líquido que las contiene) (Rivas Mijares, 1978).

La sedimentación es la separación de las partículas suspendidas más pesadas que el agua, mediante la acción de la gravedad. Es una de las operaciones unitarias más utilizadas en el tratamiento de aguas residuales.

La sedimentación se utiliza para la eliminación de : arena, materia particulada en el tanque de eliminación primaria, flóculos químicos cuando se emplea la coagulación química y para la concentración de sólidos en los espesadores de fango.

En la mayoría de los casos, el propósito es obtener un efluente clarificado, pero también es necesario producir un fango con una concentración de sólidos que pueda ser manejado y tratado con facilidad (Metcalf, 1977).

1.2 TIPOS DE SEDIMENTACION

En base a la concentración y tendencia a la interacción de las partículas pueden efectuarse cuatro clasificaciones generales sobre la forma en que dichas partículas se depositan. Es frecuente que se produzca más de un tipo de decantación en un momento dado durante la sedimentación y también es posible que los cuatro tipos tengan lugar simultáneamente.

El tipo 1 se refiere a la sedimentación de partículas discretas en una suspensión de sólidos de concentración baja. Las partículas se depositan como entidades individuales y no existe interacción significativa con las partículas próximas. Un ejemplo típico es una suspensión diluida de partículas de arena. Este tipo se conoce como sedimentación libre.

El tipo 2 se refiere a una suspensión bastante diluida de partículas que se agregan, o flocculan, durante la sedimentación. Al agregarse, las partículas aumentan de masa y se depositan más rápidamente.

El tipo 3 tiene lugar en suspensiones de concentración intermedia en las que las fuerzas interparticulares son suficientes para retardar la sedimentación de las partículas vecinas. Estas tienden a permanecer entre sí en posiciones fijas y la masa de las mismas se deposita como una unidad. En la parte superior del fango que sedimenta, se desarrolla una interfase sólido-líquido bien diferenciada. Se conoce este tipo como sedimentación zonal.

El tipo 4 tiene lugar cuando las partículas alcanzan tal concentración que se forma una estructura y solo puede producirse un nuevo asentamiento por compresión de aquella. La compresión ocurre por el peso de las partículas que continuamente se van añadiendo a la estructura por sedimentación del líquido sobrenadante. Este tipo se conoce con el nombre de sedimentación por compresión y suele presentarse en las capas inferiores de las masas profundas de fango (Metcalf, 1977).

1.3 SEDIMENTACION TIPO 1

La teoría del desplazamiento de partículas discretas, entendiéndose por estas aquellas que no cambian sensiblemente de tamaño, peso y forma en un medio líquido, dice que: " La velocidad de asentamiento de partículas discretas en un fluido en reposo tiende a acelerarse hasta tanto la resistencia friccional o arrastre del fluido, que actúa de inmediato, iguale a la fuerza de impulsión que actúa sobre la partícula." Ello provoca, en consecuencia, un desplazamiento de esa partícula a velocidad uniforme. (Metcalf, 1977).

La fuerza de impulsión mencionada es la que corresponde al peso efectivo de la partícula (su peso en el fluido donde esta suspendida) y puede expresarse así:

primero, como la expresión de la diferencia entre la fuerza gravitacional (Ec.1.1) y el empuje (Ec.1.2) :

$$F_a = \rho_s \cdot g \cdot V \quad \text{Ec.1.1}$$

$$F_e = \rho \cdot g \cdot V \quad \text{Ec.1.2}$$

Obteniendo, la fuerza de impulsión (F_i) :

$$F_i = (\rho_s - \rho) \cdot g \cdot V \quad \text{Ec.1.3}$$

Donde :

- F_i = Fuerza de impulsión
- ρ_s = densidad de masa de la partícula
- g = aceleración debida a la gravedad
- ρ = densidad del fluido
- V = volumen de la partícula.

Por otra parte, la fuerza de resistencia friccional (o arrastre) del fluido, primeramente propuesto por Newton y luego desarrollada por Stokes, aparece como una función de los siguientes parámetros: densidad de masa y viscosidad dinámica del líquido: (ρ) y (μ) respectivamente; y de expresiones del diámetro y velocidad de la partícula (d) y (v_s) respectivamente. El diámetro, por su puesto, se proyecta normalmente a la dirección de su desplazamiento, y su superficie friccional relativa (en contacto con el fluido), la correspondiente a su área superficial respecto a su propio volumen. Por lo anterior, la fuerza puede ser expresada así :

$$F_D = (C_D \cdot A \cdot \rho \cdot v_s) \quad \text{Ec.1.4}$$

en la cual :

- F_D = Fuerza de arrastre (de resistencia friccional)
- C_D = Coeficiente de arrastre de Newton
- A = Area de proyección, normal a la dirección del movimiento
- ρ = densidad del fluido
- v_s = velocidad de desplazamiento de la partícula.

A su vez, C_D , considerado por Newton como constante para cualquier tipo de flujo, es en realidad variable, función del número de Reynolds (R) :

$$R = (v \cdot d) / \eta \quad \text{Ec.1.5}$$

Donde :

- v = velocidad de la partícula en el agua
- d = diámetro de la partícula
- η = viscosidad cinemática del fluido (μ / ρ)
- μ = viscosidad absoluta
- ρ = densidad de masa

Para el caso de esferas, discos y cubos, el coeficiente de arrastre C_D varia de acuerdo al gráfico de la figura 1.1; se calcula primero, el número de Reynolds, luego se intercepta con la gráfica que represente la forma de la partícula y se encuentra C_D .

Utilizando las ecuaciones 1.3 y 1.4 anteriores se ha establecido una ecuación general para la velocidad de asentamiento o de alzamiento de partículas discretas libres (en medio de baja concentración) :

$$v_s = \sqrt{\frac{2g (\rho_s - \rho)V}{C_D \rho \cdot A}} \quad \text{Ec.1.6}$$

Donde :

- v_s = velocidad de asentamiento

El resto de la simbología mantiene su definición.

1.4 SEDIMENTACION TIPO 2

Las partículas en soluciones relativamente diluidas, a veces no actuarán como partículas discretas sino que se agregarán durante la sedimentación. Conforme se produce la coalescencia o floculación, la masa de las partículas aumentará y se depositarán rápidamente.

La floculación depende de las oportunidades de contacto que tengan las partículas, lo que varía con la carga de superficie, profundidad del tanque, gradientes de velocidad, concentración de las partículas y gama de tamaños de las mismas

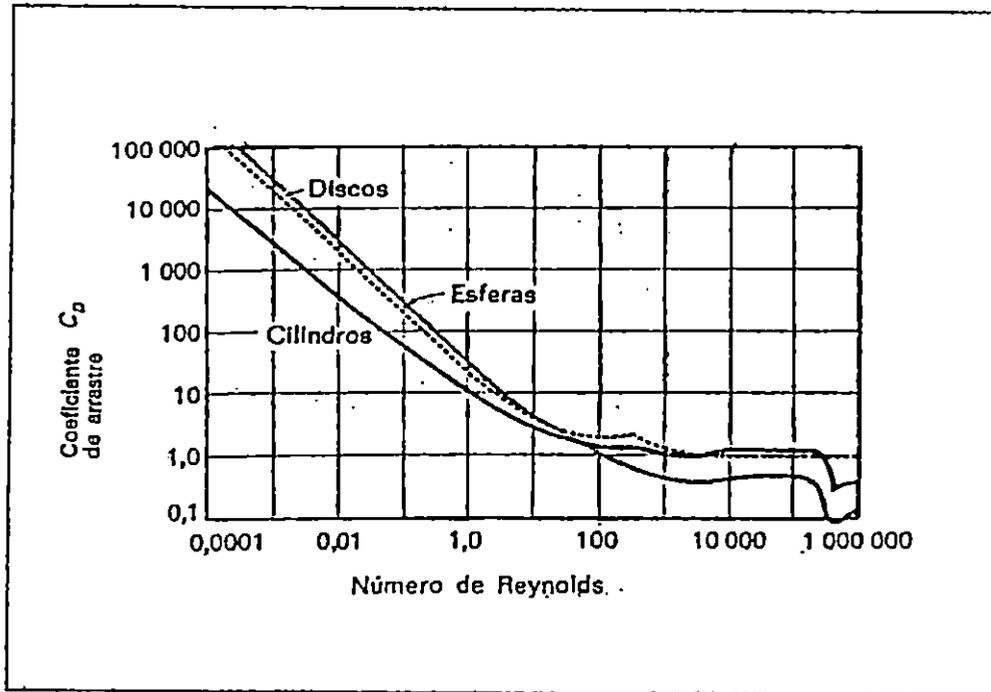


FIGURA 1.1 REGIMEN DE DESPLAZAMIENTOS (METCALF,1977)

La sedimentación se da en condiciones de reposo y se tiene cuidado en mantener una temperatura uniforme a lo largo del proceso (Metcalf,1977).

A distintos intervalos de tiempo, se retiran muestras y se analizan para conocer el número de sólidos suspendidos. Se calcula en cada muestra la eliminación porcentual y el resultado se sitúa en una gráfica tiempo vrs profundidad.

La determinación de la eliminación porcentual se realiza gráficamente y los cálculos se efectúan de la siguiente manera (Metcalf,1977) :

$$\text{Eliminación porcentual} = (\Delta h_{i+1} / h_f). (Pr_{i+1} + Pr_{i+2}) / 2 \quad \text{Ec.1.7}$$

Donde :

h = profundidad
Pr = porcentaje

1.5 SEDIMENTACION TIPO 3 Y TIPO 4

En los sistemas que contienen elevadas concentraciones de sólidos suspendidos, además de la sedimentación libre se suele producir la sedimentación zonal (tipo 3) y la sedimentación por compresión (tipo 4).

El mecanismo de sedimentación puede describirse observando lo que ocurre durante una prueba de sedimentación intermitente en una probeta de vidrio, a medida que los sólidos se sedimentan a partir de una suspensión preparada recientemente, con una concentración uniforme de partículas sólidas uniformes en toda la probeta. En cuanto se inicia el proceso, todas las partículas empiezan a sedimentarse y se supone que alcanzan con rapidez las velocidades terminales bajo condiciones de sedimentación obstaculizada. Se establecerán varias zonas de concentración, figura 1.2 b.

La zona D (región de sedimentación tipo 4) de sólidos sedimentados incluirá de manera predominante las partículas más pesadas, que se sedimentan más rápido. En una zona de transición poco definida situada por encima del material sedimentado, existen canales a través de los cuales debe subir el fluido. Este fluido es forzado desde la zona D al comprimirse.

La zona C (región de sedimentación zonal tipo 3) es una región de distribución de tamaño variable y concentración no uniforme.

La zona B (región de sedimentación retardada tipo 2) es una zona de concentración uniforme, de casi la misma concentración y distribución que había al inicio. En la parte superior de la región B existe un límite por encima del cual está el líquido transparente de la región A. Si la suspensión original tiene un tamaño cercano a las partículas más pequeñas, la línea entre A y B es definida (Foust, 1990).

A medida que continúa la sedimentación, las zonas varían como se ilustra en la figura 1.2 b, c, d. Nótese que A y D crecen a expensas de B. por último alcanza un punto en donde B y C desaparecen y todos los sólidos aparecen en D; esto se conoce como punto crítico de sedimentación (figura 1.2 e) es decir, el punto en el cual se forma una sola interfase discernible entre el líquido clarificado y el sedimento. El proceso de sedimentación desde este punto consiste en una compresión lenta de los sólidos, con el líquido desde la capa límite de cada partícula siendo forzado hacia arriba a través de los sólidos al interior de la zona clarificada. Las velocidades de sedimentación son muy lentas en esta suspensión densa. La fase final es un caso extremo de sedimentación obstaculizada (Foust, 1990).

En la operación ilustrada de sedimentación intermitente, las alturas de las diferentes zonas varían con el tiempo, las mismas zonas estarán presentes en un equipo operando con régimen continuo. Sin embargo, una vez que se ha alcanzado un estado estable (cuando la alimentación al espesador de la suspensión por unidad de tiempo es igual a la velocidad de eliminación de lodos y líquido clarificado), las alturas de todas las zonas serán constantes (Foust, 1990).

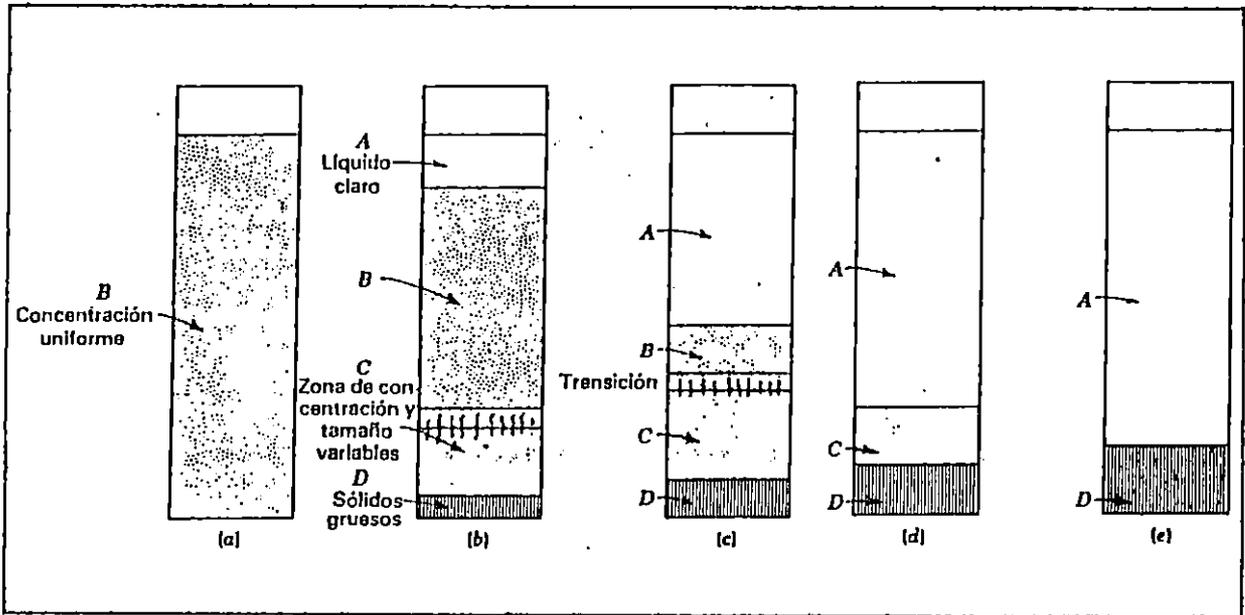


FIGURA 1.2 SEDIMENTACIÓN INTERMITENTE (FOUST,1990)

Las operaciones industriales de sedimentación pueden llevarse a cabo con equipos intermitentes o continuos que se conocen como espesadores. El equipo es sólo un tanque cilíndrico con aberturas para la alimentación de la suspensión y la eliminación del producto. El tanque se llena con una suspensión diluida y se permite que sedimente. Después de un cierto período se decanta un líquido clarificado hasta que aparece el lodo en el drenaje. El lodo se extrae del tanque por una abertura del fondo, como se indica en la figura 1.3.

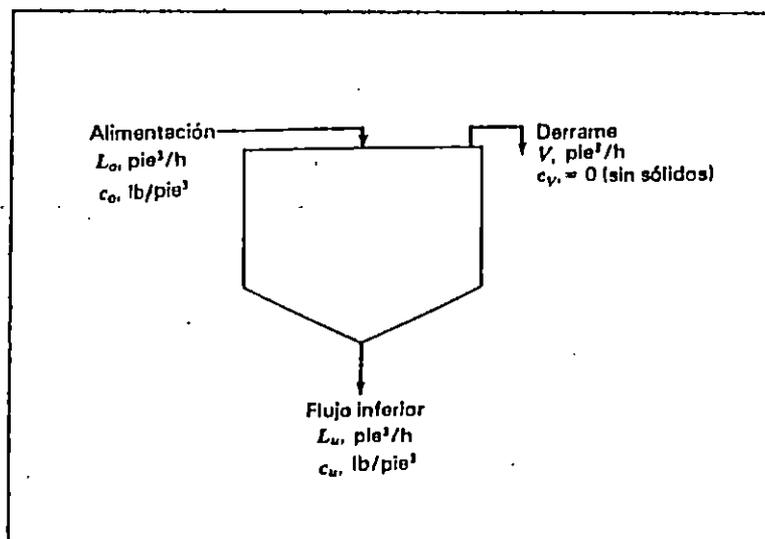


FIGURA 1.3 DIAGRAMA ESQUEMATICO DE UN ESPESADOR (FOUST,1990)

Los espesadores continuos son tanques de gran diámetro y baja profundidad con rastrillos giratorios de baja velocidad para eliminar el lodo. La suspensión se alimenta por el centro del tanque. Alrededor de la parte superior del tanque está un derrame de líquido clarificado. El rastrillo sirve para raspar el lodo hacia el centro de la parte inferior para su descarga. El movimiento del rastrillo "agita" tan sólo la capa de lodo. Esta agitación moderada ayuda a la separación del agua de los lodos (Foust, 1990).

1.5.1 CALCULO DE UN ESPESADOR CONTINUO.

El propósito de un espesador continuo es obtener una concentración mayor a partir de una suspensión de cierta concentración inicial de sólidos y a través del proceso de sedimentación. Los cálculos necesarios para el diseño de un espesador continuo están controlados por las características de sedimentación de los sólidos en la suspensión. El diseño de un espesador requiere de las especificaciones del área de sección transversal y la profundidad. Es posible diseñar una unidad para obtener un producto específico de manera continua usando información de sedimentación intermitente. Los siguientes párrafos describen los procedimientos del cálculo (Foust, 1990).

Las mediciones de laboratorio son las pruebas más satisfactorias para determinar las características de sedimentación de una cierta suspensión o lodos. Aunque estas pruebas son intermitentes, su análisis es útil para el diseño de espesadores continuos. La Figura 1.4 muestra los resultados de esta prueba cuando se obtiene la altura de la interfase líquido-sólido en función del tiempo.

Las pendientes de esta curva en cualquier momento representan velocidades de sedimentación de la suspensión en ese instante y son características de una concentración específica de sólidos.

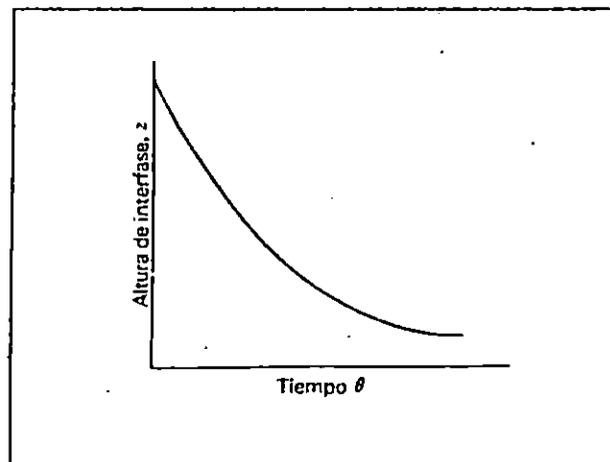


FIGURA 1.4 RESULTADOS DE UNA SEDIMENTACIÓN INTERMITENTE (FOUST, 1990)

La primera porción de la curva tiende a ser lineal, correspondiente a una velocidad constante de sedimentación de los lodos a la concentración inicial. En el espesamiento, esta región inicial representa una parte insignificante del tiempo total de espesamiento. A medida que transcurre el tiempo, la velocidad de sedimentación disminuye. Coe y Clevenger en 1916 y Kynch en 1952, propusieron un modelo para explicar esto, suponiendo que la velocidad de sedimentación es proporcional a la concentración de sólidos suspendidos. Una vez que se pasa la región de velocidad constante de sedimentación de la figura 1.4, cada punto de la curva corresponde a una concentración diferente de sólidos.

Debe enfatizarse que la curva de sedimentación determinada en el laboratorio sólo es aplicable a la suspensión considerada y que los resultados pueden tener cierta incertidumbre. Por ejemplo, debe usarse una probeta del mayor diámetro posible para minimizar los efectos de las paredes. La prueba también debe llevarse a cabo con profundidades comparables a la de la unidad propuesta (Foust, 1990).

1.5.2 CALCULOS DE DISEÑO.

Casi todos los métodos actuales para cálculos de diseño parten del trabajo de Coe y Clevenger. Como se mencionó antes, su principal suposición fue que la velocidad de descenso de la interfase sólido-líquido era una función de la concentración local. Los resultados de pruebas intermitentes muestran con claridad que la velocidad de sedimentación disminuye al aumentar la concentración. Sin embargo, la disminución es menos rápida que el incremento de concentración. Aunque este balance de velocidad lineal decreciente y flujo de masa creciente como consecuencia de la mayor densidad de sólidos en la capa que se compacta, es impredecible, tiene importancia en el procedimiento de diseño (Foust, 1990)

Una condición necesaria para la operación de un espesador continuo, es que la velocidad de sedimentación de los sólidos en cada zona debe ser, por lo menos, lo suficientemente rápida para acomodar al sólido que está llegando a ese nivel. En la parte superior del espesador, la suspensión está muy diluida y la sedimentación es muy rápida. En el fondo, la densidad y concentración de sólidos son en extremo altas y la velocidad de sedimentación es lenta.

En un espesador que opere de manera continua, los sólidos pasan a través de capas de todas las concentraciones entre la de alimentación y la del flujo inferior. Si los sólidos que entran a cualquier capa no pasan a través de ella a una velocidad igual a la de llegada de sólidos a la capa, se incrementará el espesor de ésta y crecerá hacia arriba hasta que, por último, aparecen sólidos en el derrame. El diseño del espesador se basa en la identificación de la concentración de la capa que tiene la menor capacidad para el paso de sólidos a través de ella en las condiciones de operación. Esta capa particular se conoce como capa limitante de la velocidad (Figura, 1.5). Se debe contar con un área suficiente para asegurar que el flujo de sólido especificado no exceda la capacidad de la capa limitante de la velocidad (Foust, 1990).

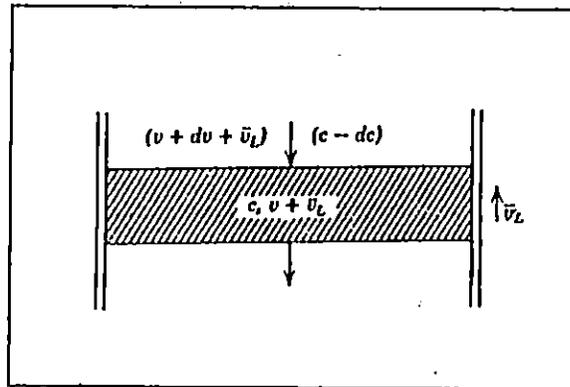


FIGURA 1.5 CAPA LIMITANTE DE LA VELOCIDAD (FOUST, 1990)

La figura 1.5 describe una capa de concentración c en una prueba intermitente. Se supone que esta capa es la limitante de la velocidad, de manera que puede considerarse que asciende a una velocidad v_L . Los sólidos se sedimentan en esta capa provenientes de la inmediatamente superior, que tiene una concentración $(c-dc)$ y una velocidad $(v+dv)$ con respecto a la columna y $(v+dv+ \bar{v}_L)$ con respecto a la capa. Los sólidos se sedimentan de esta capa a una velocidad v con respecto a las paredes y $v+ \bar{v}_L$ con respecto a la capa. Si se supone que la capa tiene una concentración constante de sólidos, entonces por medio de un balance de materiales,

$$(c-dc) S t (v+dv+ \bar{v}_L) = c S t (v+ \bar{v}_L) \quad \text{Ec.1.8}$$

donde S es el área perpendicular al flujo de sólidos y t el tiempo. Resolviendo la Ec. 1.8 se obtiene :

$$\bar{v}_L = c \cdot \frac{dv}{dc} - v - dv \quad \text{Ec.1.9}$$

y sin tomar en cuenta dv ,

$$\bar{v}_L = c \cdot \frac{dv}{dc} - v \quad \text{Ec.1.10}$$

Se ha supuesto que la velocidad de sedimentación es una función de la concentración [es decir, $v= f(c)$] de manera que $dv/dc=f'(c)$. Por tanto, la ecuación 1.10 resulta en.

$$\bar{v}_L = c \cdot f'(c) - f(c) \quad \text{Ec.1.11}$$

Dado que c es constante para esta capa, $f'(c)$ y $f(c)$ tampoco varían, de manera que v_L también es constante. El hecho de que v_L sea constante en la zona limitante de la velocidad, puede usarse para determinar la concentración de sólidos en el límite superior de la capa, a partir de una sola prueba de sedimentación intermitente. Sean c_0 y z_0 la concentración y altura iniciales de los sólidos suspendidos en una prueba de sedimentación intermitente. El peso total de sólidos en la suspensión es $c_0 z_0 S$, donde S es el área de sección transversal de la probeta en la cual se lleva a cabo la prueba. Si existe una capa limitante, debe formarse primero en el fondo y moverse hacia arriba en dirección a la interfase del líquido clarificado. Si la concentración de la capa limitante es c_L y el tiempo para alcanzar la interfase es t_L , la cantidad de sólidos que pasa a través de esta capa es $c_L - St_L (v_L + \bar{v}_L)$. Esta cantidad debe ser igual al total de sólidos presentes, dado que la capa que tiene esta concentración límite comenzó a formarse en el fondo y ascendió a la interfase. Por tanto

$$c_L St_L (v_L + \bar{v}_L) = c_0 z_0 S \quad \text{Ec.1.12}$$

Si z_L es la altura de la interfase en t_L con \bar{v}_L constante, de acuerdo con la ecuación 1.11, entonces :

$$\bar{v}_L = \frac{z_L}{t_L} \quad \text{Ec.1.13}$$

Sustituyendo el valor de v_L de la ecuación 1.13 en la ecuación 1.12 y simplificando se obtiene :

$$c_L = \frac{c_0 z_0}{z_L + v_L t_L} \quad \text{Ec.1.14}$$

Los datos de prueba del laboratorio pueden manejarse graficando la altura de la interfase en función del tiempo, como en la figura 1.6. En esta gráfica, el valor de \bar{v}_L es la pendiente de la curva en $t = t_L$. La tangente a la curva en t_L interseca a la ordenada en z_1 . La pendiente de esta línea es

$$v_L = \frac{z_1 - z_L}{t_L} \quad \text{Ec.1.15}$$

ó

$$z_1 = z_L + \theta_L v_L \quad \text{Ec.1.16}$$

Combinando la ecuaciones 1.16 y 1.14 resulta en :

$$c_L z_i = c_0 z_0$$

Ec.1.17

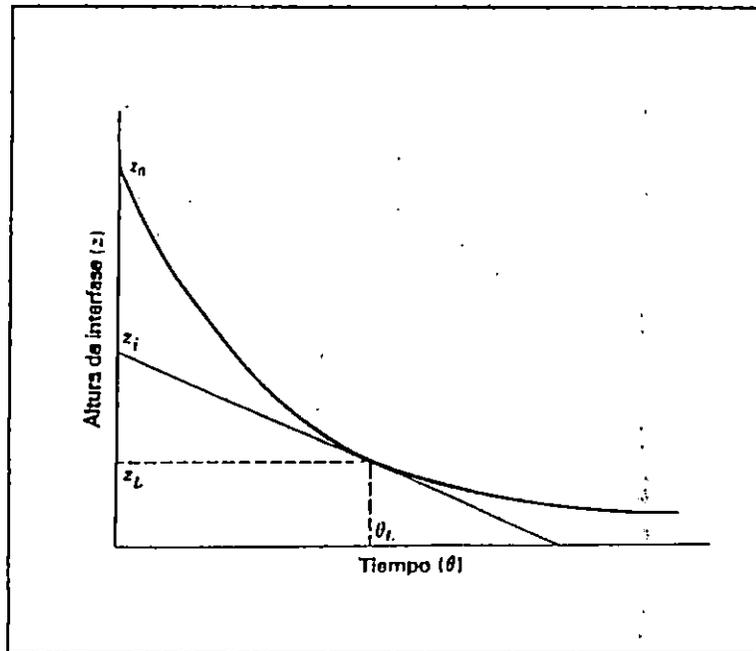


FIGURA 1.6 RESULTADOS DE UNA SEDIMENTACIÓN INTERMITENTE (FOUST,1990)

Por consiguiente, z_i es la altura que la suspensión ocuparía si todos los sólidos presentes fueran una concentración c_L . En términos del modelo antes postulado c_L es la concentración mínima para la que las capas límite producen interferencia.

La velocidad de sedimentación es una función de la concentración, determinada con una sola prueba de sedimentación como sigue. Para diversos valores arbitrarios de t 's, la pendiente de la tangente y su intersección en $t = 0$ se determinan con una gráfica de z en función de t . El valor de esta intersección se utiliza en la ec.1.17 para obtener la concentración correspondiente. A partir de esto, se obtiene v_L en función de c .

1.5.3 AREA DEL ESPESADOR

El área del espesador requerida está fijada por la capa que necesita el área máxima para el paso de una cantidad unitaria de sólidos. Por consiguiente, es necesario determinar la concentración de dicha capa. Considere la prueba de sedimentación intermitente en la que el flujo total de sólidos es la suma del flujo de las partículas que

se sedimentan y el específico para suministrar un flujo global a la salida inferior. Este flujo total es :

$$F = F_B + F_u = cv + \frac{L_u c}{A} \quad \text{Ec.1.18}$$

donde :

- c = composición de la capa.
- v = velocidad de sedimentación de los sólidos en c.
- L = velocidad volumétrica del flujo inferior.
- A = área de sección transversal perpendicular al flujo.

El primer término del lado derecho de la ec. 1.18 es el flujo de sedimentación de las partículas que se conoce como flujo intermitente y se obtiene a partir de datos de prueba por lotes. El segundo término es el flujo asociado con la eliminación de sólidos por medio del flujo inferior. En la figura 1.7 se presenta una gráfica de la ecuación 1.18 mostrando las contribuciones relativas de los dos flujos.

En la figura 1.7 el flujo de sólidos F_L con una concentración c_L es menor que el flujo de sólidos a otras concentraciones entre la alimentación y el flujo inferior. Por consiguiente, F_L es la capacidad limitante de flujo de sólidos y debe suministrarse un área de tal manera que el flujo no exceda de F_L . El área requerida por el espesador es :

$$A = \frac{L_0 C_0}{F_L} \quad \text{Ec.1.19}$$

El área calculada mediante la ec. 1.19 es la mínima que debe suministrarse para un espesador continuo operando en estado estable.

Talmadge y Fitch presentaron un método para calcular el tiempo y la concentración necesaria para alcanzar el flujo crítico, así como un medio para determinar el área mínima. De nuevo la prueba intermitente con una probeta es la base para el procedimiento. Al principio de la prueba, los sólidos se encuentran dispersos de manera uniforme a través del cilindro con una concentración c_0 . La masa total de sólidos en la probeta es $c_0 A z_0$ donde A es el área de sección transversal de la probeta y z_0 es la altura inicial de la interfase que, en este caso, es la profundidad del líquido. Sea t_c , el tiempo necesario para lograr la concentración crítica c_2 . Físicamente, esta concentración se alcanza en el momento en que la velocidad de sedimentación disminuye con rapidez.

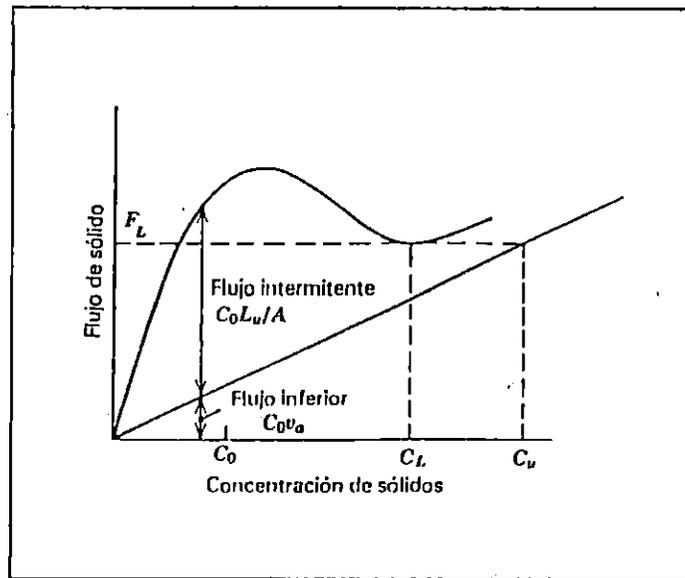


FIGURA 1.7 FLUJO DE SÓLIDOS PARA UN ESPESADOR CONTINUO (FOUST, 1990)

De nuevo se grafican los datos de prueba. Esta curva se utiliza para determinar t_2 y c_2 de la siguiente manera: La primera parte de la curva representa la sedimentación libre a una velocidad casi constante y se traza una tangente a esta parte de la curva. Al final de la prueba cuando las concentraciones son altas y las velocidades son bajas, la curva también muestra una velocidad casi constante, se prolongan las dos tangentes hasta que se interceptan. En la intersección se traza el ángulo de bisección, en la intersección de la línea de este ángulo de bisección con la curva de sedimentación produce la estimación de t_c para que los sólidos entren a la zona de compresión y la concentración en t_c es c_c .

Durante la prueba, la masa de sólidos en la probeta es constante. Un balance de materiales para los sólidos da lugar a:

$$c_0 A z_0 = c_c A z_c = c_u A z_u \quad \text{Ec.1.20}$$

ó

$$c_0 z_0 = c_c z_c = c_u z_u \quad \text{Ec.1.21}$$

Donde el subíndice u denota los valores del flujo inferior. Si el proceso de espesamiento se concibe como si pasara de la concentración c_c a c_u , debe eliminarse una cierta cantidad de agua para lograr este cambio de concentración. El volumen del agua que se separa es .

$$V = A(z_c - z_u) \quad \text{Ec.1.22}$$

Y el tiempo requerido para descargar este volumen de agua es $(t_u - t_c)$. De manera que el flujo volumétrico es :

$$L = \frac{V}{(t_u - t_c)} = \frac{A(z_c - z_u)}{t_u - t_c} \quad \text{Ec.1.23}$$

resolviendo para $(t_u - t_c)$:

$$(t_u - t_c) = \frac{A(z_c - z_u)}{L} \quad \text{Ec.1.24}$$

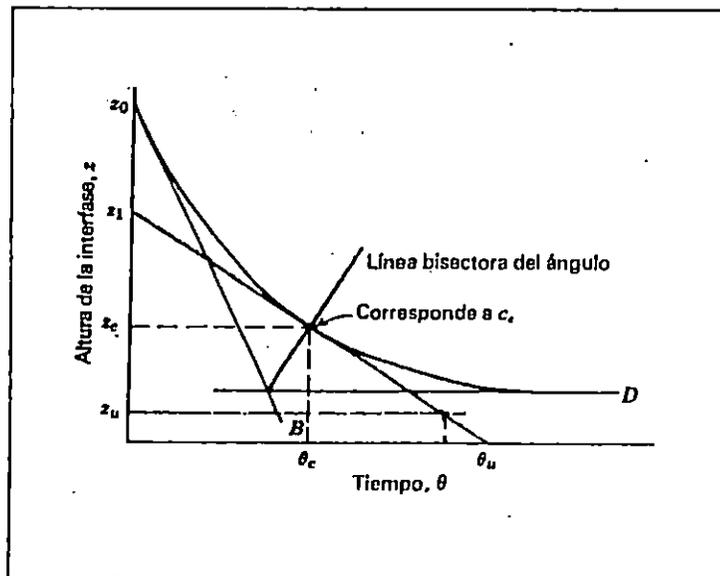


FIGURA 1.8 DETERMINACIÓN DE LA CONCENTRACIÓN CRÍTICA (FOUST, 1990)

Con la figura 1.8, se obtiene la velocidad de sedimentación en θ_c con la pendiente de curva en t_c o con la siguiente ecuación:

$$v_c = \frac{z_1 - z_c}{t_c} \quad \text{Ec.1.25}$$

En condiciones de flujo continuo, el derrame del líquido clarificado no puede exceder de v_c para que no produzca un espesamiento. Por consiguiente, el flujo en t_c cuando comienza el espesamiento, es :

$$L = A v_c = A \left(\frac{z_1 - z_c}{t_c} \right) \quad \text{Ec.1.26}$$

sustituyendo L de la Ec. 1.26 se obtiene .

$$\frac{z_c - z_u}{t_u - t_c} = \frac{z_1 - z_c}{t_c} \quad \text{Ec.1.27}$$

Se utiliza la ec. 1.26 para calcular t_u de la Figura 1.8 como sigue: se puede calcular z_u mediante la ec. 1.21:

$$z_u = \frac{z_0 c_0}{c_u} \quad \text{Ec.1.28}$$

En la figura 1.8 se localiza z_u y t_u se puede calcular mediante la ec. 1.27 u obtenerse a partir de la gráfica en donde la línea horizontal que pasa a través de z_u intersecta a la tangente de la curva a través de t_c .

El procedimiento de Talmadge y Fitch aplicado al análisis de pruebas intermitentes antes descrito, se utiliza para determinar el área mínima para un espesador continuo. La velocidad promedio de masa a la que se forma la capa de concentración c_u en el espesador es $c_u z_u A / t_u$. Pero, dado que de acuerdo con la ec. 1.21, $c_u z_u = c_0 z_0$, la velocidad promedio de masa es $c_0 z_0 A / t_u$. Para operaciones continuas, la velocidad de formación de la capa de concentración c_u debe ser igual a la entrada de sólidos a la unidad.

$$L_0 c_0 = c_0 z_0 A / t_u \quad \text{Ec.1.29}$$

resolviendo para el área, se tiene

$$A = \frac{L_0 t_u}{z_0} \quad \text{Ec.1.30}$$

1.5.4 Profundidad del espesador

Comings, en una serie de pruebas con un espesador continuo, determinó el efecto de la velocidad del flujo del inferior sobre la capacidad del espesador. En todas las corridas se obtuvo esencialmente un derrame transparente. La profundidad de la zona de espesamiento aumentó a medida que se disminuyó la velocidad del flujo inferior. Concluyó que la profundidad de la zona de espesamiento para suspensiones incompresibles es menos importante que el tiempo de retención de las partículas dentro del espesador. Es decir, el área de un espesador puede predecirse dependiendo del límite determinado para obtener el tiempo de retención necesaria. Es posible determinar el tiempo de retención a partir de datos de laboratorio con una prueba intermitente. Por tanto, dado que, en las pruebas intermitentes se utilizaron las profundidades representativas de las esperadas en el espesador a escala industrial, no se requiere un cálculo separado para el volumen del lodo espesado.

Además de la profundidad de la zona de sedimentación debe hacerse alguna provisión para inventario y aspecto de diseño mecánico. Se puede estimar la profundidad total del espesador sumando la profundidad de las zonas de sedimentación y espesamiento.

Espaciamiento del fondo	1 a 2 pies
Capacidad de almacenamiento	1 a 2 pies
Inmersión de alimentación	1 a 3 pies

Es importante notar que si la concentración del flujo inferior requerida es menor que el valor crítico determinado a partir de la prueba intermitente, los cálculos del área se basan en las velocidades de sedimentación no obstaculizada.

1.6 PLANTEAMIENTO DE LO QUE APORTARA LA CODIFICACION DEL CASO

La sedimentación, es la separación de las partículas suspendidas más pesadas que el agua mediante la acción de la gravedad.

El propósito del presente trabajo es el diseño de un espesador o sedimentador continuo, en el que a partir de una suspensión de cierta concentración inicial de sólidos se logra obtener mediante el proceso de sedimentación una concentración mayor.

1.6.1 DISEÑO DEL SEDIMENTADOR

El diseño de un espesador continuo requiere de especificaciones del área de sección transversal y la profundidad.

Las mediciones de laboratorio, son pruebas intermitentes y su análisis es útil para el diseño de espesadores continuos.

Durante la prueba se obtienen datos de altura de la interfase en función del tiempo. Luego, por medio de diferentes cálculos se encuentran: El área del sedimentador y la profundidad.

1.6.2 DEFINICION DE VARIABLES

- A = área del espesador (m^2)
 a = coeficiente de regresión
 b = coeficiente de regresión
 C_0 = concentración inicial (kg/m^3)
 C_u = concentración final de compresión (kg/m^3)
 L_0 = caudal volumétrico (m^3/h)
 t_c = tiempo necesario para alcanzar la concentración crítica (h)
 t_u = tiempo final de sedimentación, en la fase de compresión de los sólidos (h).
 v_c = velocidad de sedimentación en el punto crítico (m/h).
 Z_c = altura de la interfase a una concentración crítica (m)
 Z_L = altura de suspensión a una concentración C_L (m)
 Z_0 = altura inicial (m)
 Z_u = altura final de la interfase, en la fase de compresión de los sólidos (m)

1.6.3 SOLUCION DE MODELOS

Los datos de prueba de laboratorio se manejan graficando la altura de la interfase en función del tiempo

Una vez recolectados los datos y conociendo la variable independiente (tiempo) y la variable dependiente (altura), se obtendrá el diagrama de dispersión.

De acuerdo a la forma que toma dicho diagrama para la sedimentación se observa que la primera porción de la curva tiende a ser lineal, ya que posee una velocidad constante de sedimentación. Por lo tanto, se aplicará una función lineal:

$$Z_1 = a_1 + b_1 t \quad \text{Ec.1.31}$$

donde a y b son coeficientes de regresión que se calculan de la siguiente manera:

$$a_1 = \frac{\sum X^2 \sum Y - \sum X \sum XY}{n \sum X^2 - (\sum X)^2} \quad \text{Ec.1.32}$$

$$b_1 = \frac{n \sum XY - \sum Y \sum X}{n \sum X^2 - (\sum X)^2} \quad \text{Ec.1.33}$$

La última porción de la curva de sedimentación muestra de igual manera una velocidad casi constante debido a las altas concentraciones de la suspensión.

Por lo tanto se aplicará también una función lineal:

$$Z_2 = a_2 + b_2 t \quad \text{Ec.1.34}$$

Donde los cálculos de a_2 y b_2 se obtienen de igual forma que a_1 y b_1

1.6.4 CALCULO DE LA ECUACION DE LA BISECTRIZ

Con el objeto de llegar a conocer el tiempo final de sedimentación (t_u) donde hay compresión de sólidos, se encuentra la ecuación de la bisectriz.

Como se sabe que la distancia de un punto de la bisectriz a las rectas tangentes a la curva encontradas debe ser la misma, se igualan ambas distancias. así:

$$\frac{|A_1X + B_1Y + C_1|}{\sqrt{A_1^2 + B_1^2}} = \frac{|A_2X + B_2Y + C_2|}{\sqrt{A_2^2 + B_2^2}} \quad \text{Ec.1.35}$$

Luego, por varios cálculos matemáticos se obtiene la ecuación de la bisectriz:

$$A_3X + B_3Y + C_3 = 0 \quad \text{Ec.1.36}$$

donde A_3 , B_3 y C_3 son las constantes de la ecuación.

El diagrama 1.A muestra los pasos para el cálculo de la ecuación de la bisectriz tangente a la curva en el punto crítico.

1.6.5 MODELOS PARA LA PARTE NO LINEAL

La parte curvilínea de la gráfica de sedimentación, tendrá como modelos de predicción los siguientes:

$$1. Y = a + bX + cX^2 + dX^3 \quad \text{Ec.1.37}$$

$$2. Y = a + bX + cX^2 + d/X \quad \text{Ec.1.38}$$

$$3. Y = a + bX + cX \ln X \quad \text{Ec.1.39}$$

$$4. Y = a + b/X + c/X^2 \quad \text{Ec.1.40}$$

$$5. Y = a + bX + c\sqrt{X} \quad \text{Ec.1.41}$$

Los modelos anteriores se seleccionaron luego de introducir datos provenientes de ejemplos tomados de la bibliografía consultada, así como también de datos provenientes de pruebas realizados por estudiantes de Ingeniería Química de la Universidad de El Salvador como parte de la asignatura Procesos de Separación y Manejo de Sólidos. Los modelos elegidos son los que mejor se ajustaban a los datos de prueba.

1.6.6 PRIMERAS DERIVADAS

$$1. \frac{dY}{dX} = b + 2cX + 3dX^2 \quad \text{Ec.1.42}$$

$$2. \frac{dY}{dX} = b + 2cX - \frac{d}{X^2} \quad \text{Ec.1.43}$$

$$3. \frac{dY}{dX} = b + c(1 + \ln X) \quad \text{Ec.1.44}$$

$$4. \frac{dY}{dX} = -\frac{b}{X^2} - \frac{2c}{X^3} \quad \text{Ec.1.45}$$

$$5. \frac{dY}{dX} = b + \frac{c}{2\sqrt{X}} \quad \text{Ec.1.46}$$

1.6.7 ECUACIONES PARAMETRICAS

Las ecuaciones paramétricas correspondientes a las ecuaciones 1.42-1.46 son las siguientes:

$$1. Y = a + bX + cX^2 + dX^3$$

$$\begin{aligned} \bullet \sum Y &= an & + b\sum X & + c\sum X^2 & + d\sum X^3 \\ \bullet \sum XY &= a\sum X & + b\sum X^2 & + c\sum X^3 & + d\sum X^4 \\ \bullet \sum X^2Y &= a\sum X^2 & + b\sum X^3 & + c\sum X^4 & + d\sum X^5 \\ \bullet \sum X^3Y &= a\sum X^3 & + b\sum X^4 & + c\sum X^5 & + d\sum X^6 \end{aligned}$$

$$2. Y = a + bX + cX^2 + \frac{d}{X}$$

$$\begin{aligned} \bullet \sum Y &= an & + b\sum X & + c\sum X^2 & + d\sum(1/X) \\ \bullet \sum XY &= a\sum X & + b\sum X^2 & + c\sum X^3 & + dn \\ \bullet \sum X^2Y &= a\sum X^2 & + b\sum X^3 & + c\sum X^4 & + d\sum X \\ \bullet \sum(Y/X) &= a\sum(1/X) & + bn & + c\sum X & + d\sum(1/X^2) \end{aligned}$$

$$3. Y = a + bX + cX \cdot \ln X$$

- $\sum Y = an + b \sum X + c \sum X \cdot \ln X$
- $\sum XY = a \sum X + b \sum X^2 + c \sum X^2 \cdot \ln X$
- $\sum X \cdot \ln X = a \sum X \cdot \ln X + b \sum X^2 \cdot \ln X + c \sum (X^2 / (\ln X)^2)$

$$4. Y = a + b/X + c/X^2$$

- $\sum Y = an + b \sum (1/X) + c \sum (1/X^2)$
- $\sum (Y/X) = a \sum (1/X) + b \sum (1/X^2) + c \sum (1/X^3)$
- $\sum (Y/X^2) = a \sum (1/X^2) + b \sum (1/X^3) + c \sum (1/X^4)$

$$5. Y = a + bX + c\sqrt{X}$$

- $\sum Y = an + b \sum X + c \sum \sqrt{X}$
- $\sum XY = a \sum X + b \sum X^2 + c \sum X \sqrt{X}$
- $\sum Y \sqrt{X} = a \sum \sqrt{X} + b \sum X \sqrt{X} + c \sum X$

1.6.8 ELECCION DEL MODELO

Como modelo matemático para la parte curvilínea, se toma aquel que presente una menor diferencia entre los datos ingresados por el usuario y la curva. El mejor modelo puede ser conocido por medio del coeficiente de determinación (r^2). El coeficiente r^2 expresa la proporción de la varianza Z_i asociada con la varianza de t_i

$$r^2 = \frac{\sum (Z_c - \bar{Z})^2}{\sum (Z_i - \bar{Z})^2} \quad \text{Ec.1.47}$$

donde Z_c son los valores para Z calculados a partir de la ecuación resultante, Z_i los valores introducidos por el usuario, y \bar{Z} es el valor promedio de todos los Z_i introducidos que no se toman en cuenta en el cálculo de las rectas tangentes.

El modelo elegido (con base en un mejor r^2) se simultanea con la ecuación de la bicetriz para encontrar el punto (Z_c, t_c) .

Por este punto pasa una recta tangente, cuya ecuación es:

$$Z_i = v_s t_c + (Z_c - v_s t_c) \quad \text{Ec.1.48}$$

donde v_s es la pendiente de la recta tangente, calculada a partir de la evaluación de la primera derivada del modelo elegido en el punto de intersección entre la bisectriz y la curva, el cual es el punto crítico de sedimentación (t_c, Z_c).

1.6.9 PROFUNDIDAD DE LA ZONA DE COMPRESION

El calculo para la profundidad de la zona de compresión esta dada por la ecuación 1.28.

$$Z_u = \frac{C_0 Z_0}{C_u}$$

donde las variables C_0 , Z_0 y C_u son datos que el usuario deberá proporcionar.

1.6.10 CALCULO DEL TIEMPO FINAL DE SEDIMENTACION (t_u)

El t_u se obtiene de la igualación la ecuación de la recta tangente a la curva en el punto crítico de sedimentación (t_c, Z_c) con Z_u , así:

$$Z_u = v_s t_u + (Z_c - v_s t_c) \quad \text{Ec.1.49}$$

despejando t_u , se obtiene la siguiente ecuación

$$t_u = \frac{Z_u - Z_c + v_s t_c}{v_s} \quad \text{Ec.1.50}$$

donde t_u es el tiempo final de sedimentación.

1.6.11 DETERMINACION DEL AREA MINIMA DE SEDIMENTACION

Para el calculo del área mínima de un espesador continuo, se hace uso de la ecuación 1.30

$$A = \frac{L_0 t_u}{Z_0}$$

donde L_0 es el caudal que se someterá al proceso de sedimentación, Z_0 es un dato constante introducido por el usuario que representa la altura inicial y t_u es el tiempo de sedimentación final calculado previamente.

El diagrama 1.B muestra los pasos a seguir para el cálculo del área de un sedimentador.

1.6.12 CALCULO DE LA ALTURA

Para poder conocer la altura del estanque de sedimentación se requiere del calculo de la ecuación:

$$h = \frac{V}{A} \quad \text{Ec.1.51}$$

donde:

h= altura del sedimentador
 V= volumen de la zona de compresión
 A= área del sedimentador

El volumen de la zona de compresión se calcula por la ecuación:

$$V = \frac{Q_0 C_0}{\rho_s} (t_u - t_c) + \frac{Q_0 C_0}{\rho} \int_{t=t_c}^{t=0} \frac{1}{C_L} dt \quad \text{Ec.1.52}$$

donde:

Q_0 = Caudal a la entrada del sedimentador
 C_0 = Concentración inicial del liquido a la entrada del sedimentador.
 ρ_s = densidad del sólido.
 t_u = tiempo crítico.
 ρ = densidad del agua.

El valor de $\frac{1}{C_L}$, primero se encuentra por medio de la ecuación 1.17

$$C_L = \frac{C_0 Z_0}{Z_i}$$

resolviendo para $\frac{1}{C_L}$, se tiene

$$\frac{1}{C_L} = \frac{1}{\frac{C_0 Z_0}{Z_i}} = \frac{Z_i}{C_0 Z_0} \quad \text{Ec.1.53}$$

donde C_0 y Z_0 son la concentración inicial y altura inicial respectivamente (constantes para un mismo caso), y Z_i es la altura de la interfase. Además, geoméricamente Z_i es la intersección con el eje Y que resulta de tirar líneas tangentes a la curva de sedimentación en cualquier punto.

La evaluación para Z_i comienza para un tiempo igual a cero ($t=0$), y finaliza cuando este iguale el valor del punto crítico de sedimentación ($t=t_c$).

Debido a que no existe una sola función para la gráfica de sedimentación, ya que los primeros puntos se utilizan para la evaluación de la recta tangente al inicio de la curva de sedimentación, es necesario calcular el valor de Z_i .

El cálculo de Z_i se realiza en dos etapas:

primero se evalúa la primera derivada tanto de la recta tangente al inicio de la curva, como de la ecuación de la curva (que resulto con un mejor r^2) en un enésimo punto dado (t_n, Z_n).

Luego con el punto (t_n, Z_n) y el valor de la pendiente, se puede calcular tantas rectas tangentes a la curva como puntos seleccionados. Es recomendable tomar una cantidad considerable de puntos, ya que estos serán utilizados en el cálculo del área representada por la integral de la ecuación 1.52.

En el caso de la curva, los valores de Z_i , son calculados derivando la función seleccionada y determinando su valor para un enésimo punto (t_n, Z_n). Luego aplicando la ecuación punto pendiente, se obtiene la ecuación

$$Z_i = mt + (Z_n - v_s t_n) \quad \text{Ec.1.54}$$

Una vez calculados los valores de Z_i tanto para la recta como para la curva, se puede calcular el valor $\frac{1}{C_L}$ a partir de la ecuación 1.53.

Con los valores obtenidos de $\frac{1}{C_L}$ se puede evaluar la integral presente en la ecuación 1.52

$$\int_{t=t_c}^{t=0} \frac{1}{C_L} dt$$

Una vez calculado el valor de la integral, y sustituyendolo en la ecuación 1.56 puede determinarse el volumen para la zona de compresión.

1.7 EJEMPLO DE CALCULO PARA UN SEDIMENTADOR

Se desea diseñar un tanque de sedimentación para una suspensión con una concentración inicial $C_0 = 4 \text{ kg/m}^3$. En una probeta de laboratorio se ha estudiado la sedimentación discontinua de esta suspensión, obteniéndose los resultados de la tabla (h,t). Calcular el área y la altura necesaria para conseguir una concentración de sólido espesado $C_u = 24 \text{ kg/m}^3$ para un caudal de $16.67 \text{ m}^3/\text{h}$. $\rho_s = 1,200 \text{ kg/m}^3$.

TIEMPO (h)	ALTURA (m)
0.00	0.39
5.00	0.32
10.00	0.246
15.00	0.180
20.00	0.132
25.00	0.096
30.00	0.071
35.00	0.055
40.00	0.042
45.00	0.034
50.00	0.026
55.00	0.020
60.00	0.018
65.00	0.016

SOLUCION:

Para poder solucionar el problema en base al modelamiento matemático presentado en la sección 1.6, se sigue el siguiente procedimiento:

1. Calculo de las ecuaciones tangentes a la curva de sedimentación

La sedimentación en un inicio es libre y se desarrolla a una velocidad casi constante, por lo que puede representarse por medio de una ecuación lineal como se muestra en la ecuación 1.30

$$Z_1 = a_1 + b_1 t$$

Las constantes de regresión a, b se calculan usando las ecuaciones 1.32 y 1.33 respectivamente

$$a_1 = \frac{\sum X^2 \sum Y - \sum X \sum XY}{n \sum X^2 - (\sum X)^2}$$

$$b_1 = \frac{n \sum XY - \sum Y \sum X}{n \sum X^2 - (\sum X)^2}$$

Según estas ecuaciones necesitamos calcular las columnas que se presentan en el siguiente tabla 1.1

Para este caso se ha considerado que los tres primeros puntos se aproximan a la linealidad, y es a los cuales se les aplica el análisis de regresión para el cálculo de la ecuación lineal inicial.

TABLA 1.1 SUMATORIAS PARA EL CALCULO DE a_1 Y b_1

t	Z _t	tZ _t	t ²	Z _t ²
0.00	0.39	0.00	0.00	0.1521
5.00	0.32	1.60	25.00	0.1024
10.00	0.246	2.46	100.00	0.060516
$\sum t=15$	$\sum Z=0.956$	$\sum tZ=4.06$	$\sum t^2=125$	$\sum Z^2=0.315016$

Sustituyendo los valores de las sumatorias en la ecuación 1.30, se obtiene que:

$$a_1 = 0.391$$

$$b_1 = -0.014$$

entonces, la ecuación inicial de la recta tangente a la curva es,

$$Y = 0.391 - 0.144X$$

Ec.1.55

La parte final de la curva, al igual que la parte inicial muestra una tendencia lineal, debido a que las concentraciones son altas y las velocidades bajas. La ecuación lineal en esta parte se calcula de la misma forma que para la parte inicial.

Utilizando las mismas ecuaciones y el mismo análisis de regresión que el caso inicial, y considerado que los cuatro últimos puntos se aproximan a la linealidad, se obtiene los siguientes datos para las sumatorias .

TABLA 1.2 SUMATORIAS PARA EL CALCULO DE a_2 Y b_2

t	Z	tZ	t ²	Z ²
50.00	0.026	1.300	2500	0.000676
55.00	0.020	1.100	3025	0.0004
60.00	0.018	1.080	3600	0.000324
65.00	0.016	1.040	4225	0.000256
$\Sigma t=230$	$\Sigma Z=0.08$	$\Sigma tZ=4.52$	$\Sigma t^2=13350$	$\Sigma Z^2=0.001656$

Sustituyendo los valores de las sumatorias en la ecuación 1.34 se obtienen los datos,

$$a_2 = 0.0568$$

$$b_2 = -6.4 \times 10^{-4}$$

entonces, se tiene que la ecuación final de la recta tangente a la curva es,

$$Y = 0.0568 - 6.4 \times 10^{-4} X \quad \text{Ec.1.56}$$

La ecuación de la bisectriz entre las ecuaciones 1.55 y 1.56 se obtiene al sustituir los valores de las variables A,B,C correspondientes a la ecuación 1.36 de la distancia de un punto a una recta, para ambas ecuaciones lineales.

Por diferentes cálculos matemáticos se obtiene la ecuación de la bisectriz:

$$Y = 133X - 4324.7 \quad \text{Ec.1.57}$$

Luego, se calcula la ecuación de la curva para la parte no lineal, donde las velocidades de sedimentación no son constantes. Para este calculo se utilizan los valores ingresados por el usuario que no se tomaron en cuenta para la determinación de las ecuaciones lineales.

TABLA 1.3 DATOS PARA EL CALCULO DE LA ECUACION DE LA CURVA

Tiempo (h)	Altura (m)
15	0.18
20	0.132
25	0.096
30	0.071
35	0.055
40	0.042
45	0.034

A continuación se presentan los resultados obtenidos de todas las sumatorias, según las ecuaciones normales, las cuales se sustituyen en dichas ecuaciones con la finalidad de encontrar los valores de las constantes.

$$\begin{aligned} \Sigma X &= 210 \\ \Sigma Y &= 0.61 \\ \Sigma XY &= 15.005 \\ \Sigma X^2 Y &= 15.005 \\ \Sigma X^2 Y^2 &= 420.625 \\ \Sigma X^3 Y &= 13224.875 \\ \Sigma X^2 &= 7000 \\ \Sigma X^3 &= 252000 \\ \Sigma X^4 &= 9572500 \\ \Sigma X^5 &= 377475000 \\ \Sigma X^6 &= 15286562500 \\ \Sigma (1/X) &= 0.265793 \\ \Sigma (1/X^2) &= 0.01159 \\ \Sigma X^3 &= 0.00057217 \\ \Sigma X^4 &= 0.000031041 \\ \Sigma (Y/X) &= 0.02811 \\ \Sigma (Y/X^2) &= 0.0052528 \\ \Sigma \sqrt{X} &= 37.771173 \\ \Sigma (X\sqrt{X}) &= 1198.76837 \\ \Sigma Y\sqrt{X} &= 2.975432 \\ \Sigma X \ln X &= 726.3342 \\ \Sigma XY \ln X &= 49.05586 \\ \Sigma X^2 \ln X &= 24846.477 \\ \Sigma X^2 \ln X^2 &= 88727.6327 \end{aligned}$$

El modelo matemático que resulte será el que presente el coeficiente de determinación (r^2) más cercano a la unidad. Este cálculo se realiza utilizando la ecuación 1.47, y los resultados obtenidos para cada modelo se muestran en la tabla. 1.4

TABLA 1.4 COEFICIENTES DE DETERMINACION PARA LOS MODELOS SELECCIONADOS.

Número de Ecuación	Ecuación	coeficiente de determinación (r^2)
1.37	$Y = a + bX + cX^2 + dX^3$	0.999803
1.38	$Y = a + bX + cX^2 + d/X$	0.999039
1.39	$Y = a + bX + cX \ln X$	0.997448
1.40	$Y = a + b/X + c/X^2$	0.999134
1.41	$Y = a + bX + c\sqrt{X}$	0.998328

Puede observarse que de las cinco expresiones la que posee un mejor r^2 es la ecuación 1.37, por tanto es el modelo que se utilizará en el cálculo del punto de intersección con la ecuación de la bisectriz. El punto de intersección corresponde al punto crítico de sedimentación (t_c , Z_c).

Los valores de las constantes a,b,c y d de la ecuación seleccionada se determinan Sustituyendo las sumatorias en las ecuaciones paramétricas presentadas para cada ecuación en la parte no lineal del gráfico, obteniéndose los siguientes valores,

$$a= 0.455143$$

$$b=-0.026155$$

$$c= 0.000599$$

$$d=-5.111E-06$$

1.7.1 DETERMINACION DEL PUNTO CRITICO DE SEDIMENTACION

Con las ecuaciones calculas tanto para la bisectriz como para la parte no lineal, se procede al cálculo del punto crítico.

Igualando y realizando una prueba y error entre las ecuaciones mencionadas se puede determinar el punto exacto donde se unen ambas ecuaciones. Este procedimiento da como resultado el valor para el punto crítico (t_c , Z_c), así:

$$\text{ecuación de la bisectriz } Y=133X-4324.7$$

$$\text{ecuación de la parte no lineal } Y= 0.455143-0.026155+0.000599-5.111E-06$$

igualando, se tiene:

$$133X-4324.7= 0.455143-0.026155+0.000599-5.111E-06$$

de donde resulta que

$$(t_c, Z_c)= (X,Y)= (32.52,0.06)$$

Una vez obtenido el valor del punto crítico de sedimentación, se procede al cálculo de la ecuación de la línea recta, tangente a la curva en el punto crítico (intersección de la bisectriz con la curva), calculado en el procedimiento anterior.

Para la determinación de la recta tangente en el punto crítico, es necesario evaluar la primera derivada ecuación 1.42 de la ecuación seleccionada (Ec.1.37), en el punto

crítico. El valor resultante corresponderá al valor de la pendiente (v_s) de la recta tangente en el punto crítico.

La primera derivada para la ecuación 1.42, es

$$dY/dX = b + 2c X + 3d X^2$$

donde b, c y d son los mismos valores correspondientes a las constantes del modelo matemático elegido en la parte no lineal y X corresponde al valor del tiempo crítico (t_c).

sustituyendo los valores de las constantes y el valor de X (mismo t_c) en la ecuación de la primera derivada se obtiene que,

$$v_s = \text{pendiente} = -0.0034$$

donde v_s es la pendiente de la recta tangente en el punto crítico, y representa la velocidad de sedimentación en ese momento para dicho punto.

ahora que se conoce el valor de la pendiente en el punto crítico, puede calcularse la ecuación de la recta tangente a la curva en ese punto.

Utilizando la ecuación punto pendiente para la determinación de la recta tangente y resolviendo para Z se obtiene la expresión,

$$Z = v_s t + (Z_i, v_s t_i)$$

donde, Z representa la altura para un tiempo dado

Sustituyendo valores y operando matemáticamente se obtiene la ecuación de la recta tangente a la curva,

$$Y = -0.0034X + 0.1706 \quad \text{Ec. 1.58}$$

Ahora igualando la ecuación 1.28 con la ecuación 1.58 y realizando diferentes operaciones matemáticas, se llega a obtener un valor para el tiempo t_u de,

$$t_u = 31.06 \text{ min}$$

luego sustituyendo en la ecuación 1.30, se obtiene un valor de área de,

$$A = 22.14 \text{ m}^2$$

La altura del espesador se encuentra, en base a los siguientes criterios:

Profundidad de:

Zona de compresión	0.065 m
Inclinación del fondo	0.61 m
Capacidad de almacenamiento	0.61 m
Sumersión de la alimentación	0.61 m

Por lo tanto la profundidad total es igual a:

profundidad total= 1.895 m.

DIAGRAMAS DE FLUJO
PARA
SEDIMENTACION

DIAGRAMA 1.A	CALCULO DE LA ECUACION DE LA BISECTRIZ
DIAGRAMA 1.B	PRODEDIMIENTO DE CALCULO PARA LA RECTA TANGENTE
DIAGRAMA 1.C	CALCULO DEL AREA DE UN SEDIMENTADOR

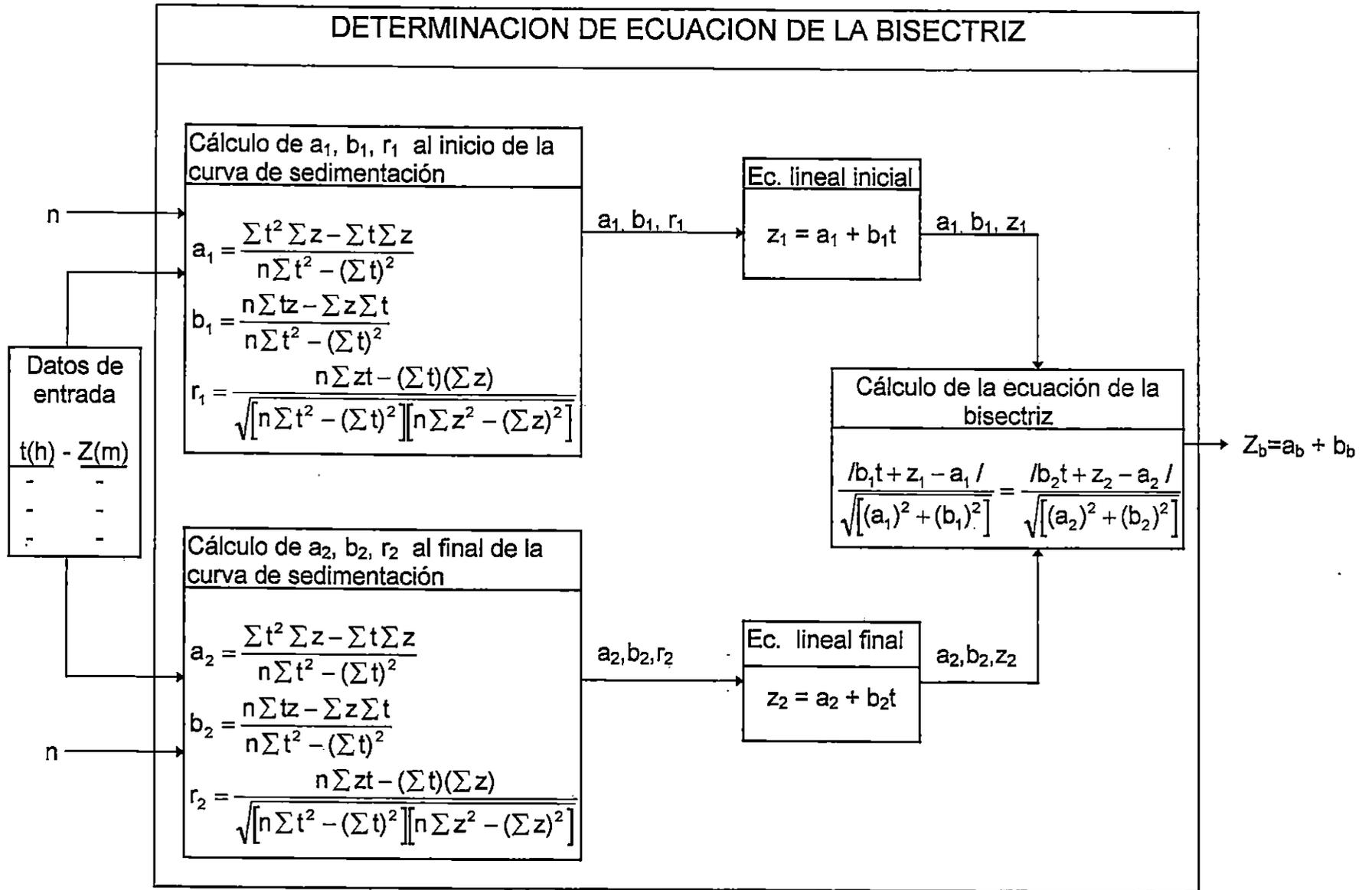


DIAGRAMA 1.A CALCULO DELA ECUACION DE LA BISECTRIZ

CALCULO DE LA PENDIENTE DE LA RECTA TANGENTE EN EL PUNTO DE INTERSECCION DE LA BISECTRIZ CON LA ECUACION DE LA CURVA

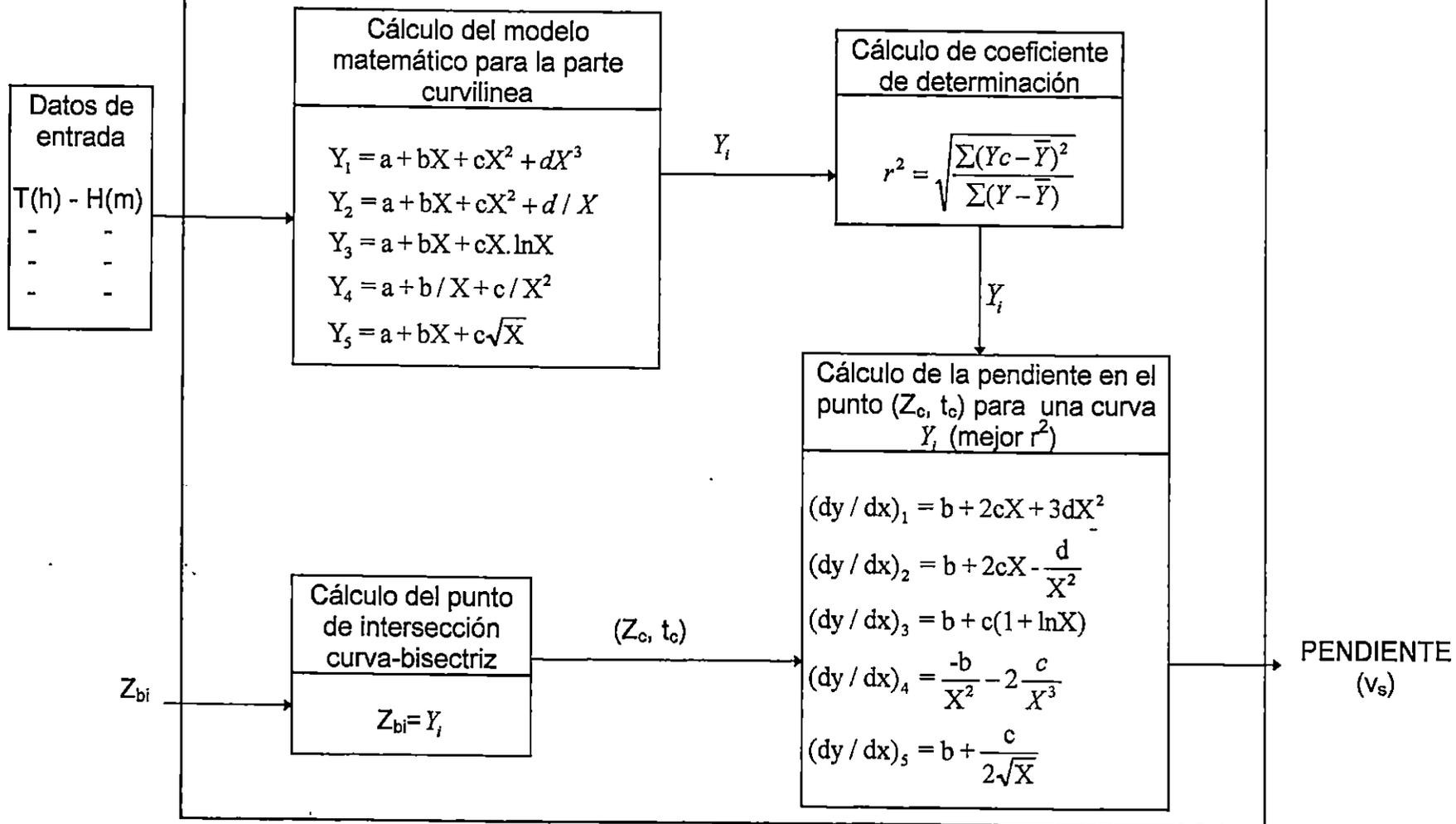


DIAGRAMA 1.B PROCEDIMIENTO DE CALCULO PARA LA RECTA TANGENTE

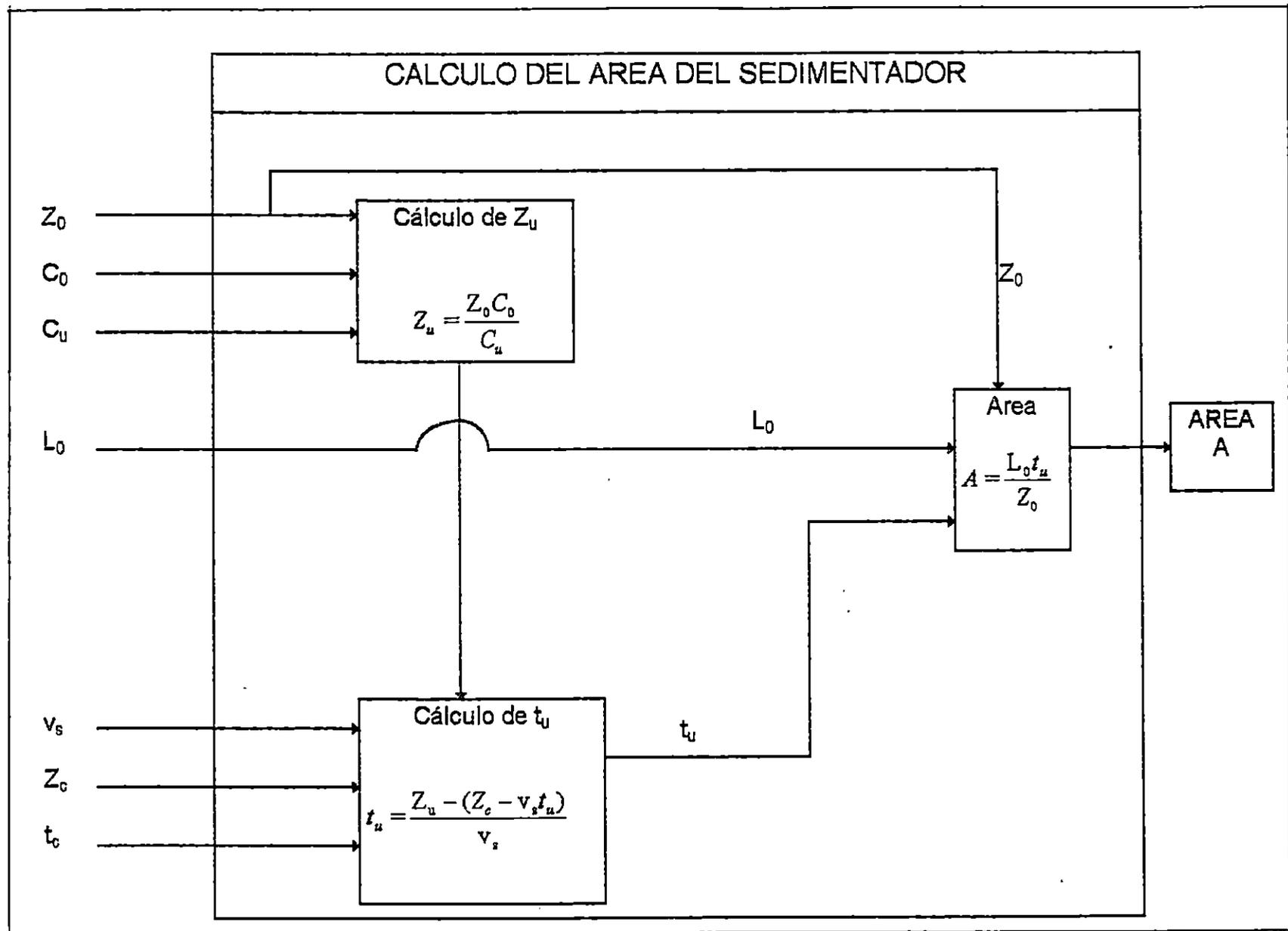


DIAGRAMA 1.C CALCULO DEL AREA PARA UN SEDIMENTADOR

1.8 PROGRAMA DE SEDIMENTACION EN QUICK C

```

#include "conio.h"
#include "stdio.h"
#include "stdlib.h"
#include "dos.h"
#include "stdarg.h"
#include "math.h"

/* ----- FUNCIONES COMUNES ----- */
int CreaVentanaMensaje(Int MensajeUnico, int NumeroPalabras,...);
void Cursor(int row, int col);
void BorraBloque(int top, int left, int btm, int right, int attr);
void DespliegaCadena(char *str, int attr, int row, int col);
void pca(char c, int attr, int row, int col);
void Ventana(int trow, int lcol, int brow, int rcol, int attr, int border,
             char *title);
int getattr(void);
void cls(int attr);
int getvmode(int *iscolor);

#define IZQUIERDA    75
#define DERECHA     77
#define ARRIBA      72
#define ABAJO       80
// #define F3        61
// #define F1        59
#define ENTER       13
#define ESCAPE      27
#define WHITE       15
#define BLACK       0
#define BLUE        1
/* ----- Define's propios ----- */
#define MENSAJE_ERR01 "ERR01: El Dato debe ser >= 0 y <= 100. Presione Cualquier Tecla."
#define MENSAJE_ERR02 "ERR02: El Dato debe ser > 0. Presione Cualquier Tecla."
#define MENSAJE_ERR03 "ERR03: El Dato debe ser > 0 y <= 12. Presione Cualquier Tecla."
#define MENSAJE_ERR04 "ERR04: Memoria Insuficiente para esta Opcion. Presione ESCAPE."
#define MAGENTA     5
#define CONST_A     0.90132612
#define CONST_B     -0.0095455207
#define CONST_C     0.000022472713
#define CONST_D     26.485764
#define EXP         2.718
#define DIF         0.1
/* ----- */

int i, PosicionMenu, j;
int attr0, attr1, attr2, attr3, iscolor;
char temp[80], Imprimir[1];
FILE *pfile5;
/* ----- Variables propias ----- */
char Respuesta[1];
int NumeroCorrecto, attr4, NumeroDatos, Linea, ColumnaTiempo, ColumnaAltura;
int LineaInicio, LineaFin, DatosCurva, FilaR, k;
char NumeroEnCadena[80], temp1[80];
double Tiempo[50], Altura[50], CoefA[3], CoefB[3], CoefR[3], Factor1;
double SumaTiempoR1, SumaAlturaR1, ProductoTAR1, ProductoT2R1, ProductoA2R1;
double SumaTiempoR2, SumaAlturaR2, ProductoTAR2, ProductoT2R2, ProductoA2R2;
double Total[35], MCramer[4][4], Const[4], CoefSolucion[5][5], PromedioAltura;
double AlturaCalculada[50], SumaDif1, SumaDif2, ValorR, ValorYRecta, ValorYCurva;
double ValorX, DifIntersec, Pendiente, AlturaInicial, AlturaFinal, Dif;
double ConcentracionInicial, ConcentracionFinal, TiempoBuscado, Caudal, Area;
double ProfundidadTotal;
double A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, A12;
double B1, B2, B3, B4, B5, B6, B7, B8, B9, B10, B11, B12;
double D1, D2, D3, D4, D5, D6, D7, D8, D9, D10, D11, D12;
double E1, E2, E3, E4, E5, E6, E7, E8, E9, E10, E11, E12;
double F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12;

```

```

double D,DA,DB,DC,DD;

/* ----- Funciones propias ----- */
void Sedimentacion(void);
int PideDatos(void);
int IniciaArreglos(void);
int Cramer(int NumFila);
int Cramer3(int NumFila);
int Calculos();

/* ----- */
void Cursor(int row, int col)
/* ----- */
{
    /* envja el Cursor a la Posicjñ (row=0,col=0 Origin) */
    union REGS rin,rout;
    rin.h.ah=2;
    rin.h.bh=0;
    rin.h.dh=row;
    rin.h.dl=col;
    int86(0x10,&rin,&rout);
}

/* ----- */
void DespliegaCadena(char *str, int attr, int row, int col)
/* ----- */
{
    /* Escribe una Cadena con Atributo y coordenadas */
    while (*str)
        pca(*str++,attr,row,col++);
}

/* ----- */
void BorraBloque(int top, int left, int btm, int right, int attr)
/* ----- */
{
    /* Clears Screen Block w/Attribute, Cursor inside */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.al=0;
    rin.h.bh=attr;
    rin.h.ch=top; /* top row */
    rin.h.cl=left; /* top column */
    rin.h.dh=btm; /* bottom row */
    rin.h.dl=right; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(top+1,left+1);
}

/* ----- */
int getattr(void)
/* ----- */
{
    union REGS rin,rout;
    rin.h.ah=8; /* get attribute at Cursor */
    rin.h.bh=0;
    int86(0x10,&rin,&rout);
    return((int)rout.h.ah);
}

/* ----- */
void pca(char c, int attr, int row, int col)
/* ----- */
{
    union REGS rin,rout;
    Cursor(row,col); /* set cursor */
    rin.h.ah=9; /* print char with attribute */
    rin.h.al=c;
    rin.h.bh=0;
    rin.h.bl=attr;
    rin.x.cx=1;
    int86(0x10,&rin,&rout);
}

```

```

/*-----*/
void Ventana(int trow, int lcol, int brow, int rcol, int attr,
int border, char *title)
/*-----*/
{
    /* Set Simple Window and Title */
    int tl,tr,bl,br,lr,tb,i,length;
    BorraBloque(trow,lcol,brow,rcol,attr);
    if(border==1){ /* Single Border */
        tl=218; tr=191; bl=192; br=217; lr=196; tb=179;
    }
    else{ /* Double Border */
        tl=201; tr=187; bl=200; br=188; lr=205; tb=186;
    }
    pca(tl,attr,trow,lcol);
    pca(bl,attr,brow,lcol);
    pca(tr,attr,trow,rcol);
    pca(br,attr,brow,rcol);
    for(i=lcol+1; i<rcol; i++){
        pca(lr,attr,trow,i);
        pca(lr,attr,brow,i);
    }
    for(i=trow+1; i<brow; i++){
        pca(tb,attr,i,lcol);
        pca(tb,attr,i,rcol);
    }
    i=0;
    while(title[i++]!='\0');
    length=i-1;
    DespliegaCadena(title,attr,trow,(rcol+lcol)/2-length/2);
}

/*-----*/
void cls(int attr)
/*-----*/
{
    /* Clears Screen w/Attribute, Cursor to Origin */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.ai=0;
    rin.h.bh=attr;
    rin.h.ch=0; /* top row */
    rin.h.cl=0; /* top column */
    rin.h.dh=24; /* bottom row */
    rin.h.dl=79; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(0,0);
}

/*-----*/
int getvmode(int *iscolor)
/*-----*/
{
    union REGS rin,rout;
    int mode;
    rin.h.ah=15; /* get display mode */
    int86(0x10,&rin,&rout);
    mode=(int)rout.h.ah;
    *iscolor=1;
    if(mode==0 || mode==2 || mode==7)
        *iscolor=0;
    return(mode); /* sets color flag, returns video mode */
}

/*-----*/
int CreaVentanaMensaje(int Unico, int NumeroPalabras, ...)
/*-----*/
{
    int Bandera = 1;
    int Longitud;
    char *Rotulo;
}

```

```

if (malloc(205 * sizeof(double)) == NULL)
{
    DespliegaCadena("ALTO!!!!",atr4,20,20);
    while(!kbhit());
    return (-1);
}

for (i = 0; i <= 49; i++)
{
    Tiempo[i] = 0;
    Altura[i] = 0;
}

for (i = 0; i <= 3; i++)
    for (j = 0; j <= 3; j++)
        MCramer[j][i] = 0;

for (i = 0; i <= 3; i++)
    for (j = 0; j <= 5; j++)
        CoefSolucion[j][i] = 0;

for (i = 0; i <= 34; i++)
    Total[i] = 0;

return(0);
}

/*-----*/
int Cramer3(int NumFila)
/*-----*/
{
    /* float A1,A2,A3;
    float B1,B2,B3;
    float D1,D2,D3;
    float E1,E2,E3;
    float F1,F2,F3;
    float D,DA,DB,DC; */
    // float C[3];

    A1=B1=D1=E1=0.00;
    A2=B2=D2=E2=0.00;
    A3=B3=D3=E3=0.00;

    A1=MCramer[0][0]*(MCramer[1][1]*MCramer[2][2]-MCramer[2][1]*MCramer[1][2]);
    A2=-MCramer[1][0]*(MCramer[0][1]*MCramer[2][2]-MCramer[2][1]*MCramer[0][2]);
    A3=MCramer[2][0]*(MCramer[0][1]*MCramer[1][2]-MCramer[1][1]*MCramer[0][2]);

    D=A1+A2+A3;

    B1=Const[0]*(MCramer[1][1]*MCramer[2][2]-MCramer[2][1]*MCramer[1][2]);
    B2=-Const[1]*(MCramer[0][1]*MCramer[2][2]-MCramer[2][1]*MCramer[0][2]);
    B3=Const[2]*(MCramer[0][1]*MCramer[1][2]-MCramer[1][1]*MCramer[0][2]);

    DA=B1+B2+B3;

    D1=-Const[0]*(MCramer[1][0]*MCramer[2][2]-MCramer[2][0]*MCramer[1][2]);
    D2=Const[1]*(MCramer[0][0]*MCramer[2][2]-MCramer[2][0]*MCramer[0][2]);
    D3=-Const[2]*(MCramer[0][0]*MCramer[1][2]-MCramer[1][0]*MCramer[0][2]);

    DB=D1+D2+D3;

    E1=Const[0]*(MCramer[1][0]*MCramer[2][1]-MCramer[2][0]*MCramer[1][1]);
    E2=-Const[1]*(MCramer[0][0]*MCramer[2][1]-MCramer[2][0]*MCramer[0][1]);
    E3=Const[2]*(MCramer[0][0]*MCramer[1][1]-MCramer[1][0]*MCramer[0][1]);

    DC=E1+E2+E3;

    CoefSolucion[NumFila-1][0] = DA / D;
    CoefSolucion[NumFila-1][1] = DB / D;
    CoefSolucion[NumFila-1][2] = DC / D;

```

```

return(0);
}

/*-----*/
int Cramer(int NumFila)
/*-----*/
{
A1=MCramer[2][2]*(MCramer[3][3]*MCramer[4][4]-MCramer[4][3]*MCramer[3][4]);
A2=MCramer[3][2]*(MCramer[2][3]*MCramer[4][4]-MCramer[4][3]*MCramer[2][4]);
A3=MCramer[4][2]*(MCramer[2][3]*MCramer[3][4]-MCramer[3][3]*MCramer[2][4]);
A4=MCramer[2][1]*(MCramer[3][3]*MCramer[4][4]-MCramer[4][3]*MCramer[3][4]);
A5=MCramer[3][1]*(MCramer[2][3]*MCramer[4][4]-MCramer[4][3]*MCramer[2][4]);
A6=MCramer[4][1]*(MCramer[2][3]*MCramer[3][4]-MCramer[3][3]*MCramer[2][4]);
A7=MCramer[2][1]*(MCramer[3][2]*MCramer[4][4]-MCramer[4][2]*MCramer[3][4]);
A8=MCramer[3][1]*(MCramer[2][2]*MCramer[4][4]-MCramer[4][2]*MCramer[2][4]);
A9=MCramer[4][1]*(MCramer[2][2]*MCramer[3][4]-MCramer[3][2]*MCramer[2][4]);
A10=MCramer[2][1]*(MCramer[3][2]*MCramer[4][3]-MCramer[4][2]*MCramer[3][3]);
A11=MCramer[3][1]*(MCramer[2][2]*MCramer[4][3]-MCramer[4][2]*MCramer[2][3]);
A12=MCramer[4][1]*(MCramer[2][2]*MCramer[3][3]-MCramer[3][2]*MCramer[2][3]);

D =MCramer[1][1]*(A1-A2+A3)+MCramer[1][2]*(-A4+A5-A6)+MCramer[1][3]*(A7-A8+A9)+MCramer[1][4]*(-A10+A11-A12);

B1=MCramer[2][2]*(MCramer[3][3]*MCramer[4][4]-MCramer[4][3]*MCramer[3][4]);
B2=MCramer[3][2]*(MCramer[2][3]*MCramer[4][4]-MCramer[4][3]*MCramer[2][4]);
B3=MCramer[4][2]*(MCramer[2][3]*MCramer[3][4]-MCramer[3][3]*MCramer[2][4]);
B4=MCramer[1][2]*(MCramer[3][3]*MCramer[4][4]-MCramer[4][3]*MCramer[3][4]);
B5=MCramer[3][2]*(MCramer[1][3]*MCramer[4][4]-MCramer[4][3]*MCramer[1][4]);
B6=MCramer[4][2]*(MCramer[1][3]*MCramer[3][4]-MCramer[3][3]*MCramer[1][4]);
B7=MCramer[1][2]*(MCramer[2][3]*MCramer[4][4]-MCramer[4][3]*MCramer[2][4]);
B8=MCramer[2][2]*(MCramer[1][3]*MCramer[4][4]-MCramer[4][3]*MCramer[1][4]);
B9=MCramer[4][2]*(MCramer[1][3]*MCramer[2][4]-MCramer[2][3]*MCramer[1][4]);
B10=MCramer[1][2]*(MCramer[2][3]*MCramer[3][4]-MCramer[3][3]*MCramer[2][4]);
B11=MCramer[2][2]*(MCramer[1][3]*MCramer[3][4]-MCramer[3][3]*MCramer[1][4]);
B12=MCramer[3][2]*(MCramer[1][3]*MCramer[2][4]-MCramer[2][3]*MCramer[1][4]);

DA = (Const[1]*(B1-B2+B3))- (Const[2]*(B4-B5+B6))+ (Const[3]*(B7-B8+B9))- (Const[4]*(B10-B11+B12));

D1=MCramer[2][1]*(MCramer[3][3]*MCramer[4][4]-MCramer[4][3]*MCramer[3][4]);
D2=MCramer[3][1]*(MCramer[2][3]*MCramer[4][4]-MCramer[4][3]*MCramer[2][4]);
D3=MCramer[4][1]*(MCramer[2][3]*MCramer[3][4]-MCramer[3][3]*MCramer[2][4]);
D4=MCramer[1][1]*(MCramer[3][3]*MCramer[4][4]-MCramer[4][3]*MCramer[3][4]);
D5=MCramer[3][1]*(MCramer[1][3]*MCramer[4][4]-MCramer[4][3]*MCramer[1][4]);
D6=MCramer[4][1]*(MCramer[1][3]*MCramer[3][4]-MCramer[3][3]*MCramer[1][4]);
D7=MCramer[1][1]*(MCramer[2][3]*MCramer[4][4]-MCramer[4][3]*MCramer[2][4]);
D8=MCramer[2][1]*(MCramer[1][3]*MCramer[4][4]-MCramer[4][3]*MCramer[1][4]);
D9=MCramer[4][1]*(MCramer[1][3]*MCramer[2][4]-MCramer[2][3]*MCramer[1][4]);
D10=MCramer[1][1]*(MCramer[2][3]*MCramer[3][4]-MCramer[3][3]*MCramer[2][4]);
D11=MCramer[2][1]*(MCramer[1][3]*MCramer[3][4]-MCramer[3][3]*MCramer[1][4]);
D12=MCramer[3][1]*(MCramer[1][3]*MCramer[2][4]-MCramer[2][3]*MCramer[1][4]);

DB=(Const[1]*(-D1+D2-D3))+ (Const[2]*(D4-D5+D6))+ (Const[3]*(-D7+D8-D9))+ (Const[4]*(D10-D11+D12));

E1=MCramer[2][1]*(MCramer[3][2]*MCramer[4][4]-MCramer[4][2]*MCramer[3][4]);
E2=MCramer[3][1]*(MCramer[2][2]*MCramer[4][4]-MCramer[4][2]*MCramer[2][4]);
E3=MCramer[4][1]*(MCramer[2][2]*MCramer[3][4]-MCramer[3][2]*MCramer[2][4]);
E4=MCramer[1][1]*(MCramer[3][2]*MCramer[4][4]-MCramer[4][2]*MCramer[3][4]);
E5=MCramer[3][1]*(MCramer[1][2]*MCramer[4][4]-MCramer[4][2]*MCramer[1][4]);
E6=MCramer[4][1]*(MCramer[1][2]*MCramer[3][4]-MCramer[3][2]*MCramer[1][4]);
E7=MCramer[1][1]*(MCramer[2][2]*MCramer[4][4]-MCramer[4][2]*MCramer[2][4]);
E8=MCramer[2][1]*(MCramer[1][2]*MCramer[4][4]-MCramer[4][2]*MCramer[1][4]);
E9=MCramer[4][1]*(MCramer[1][2]*MCramer[2][4]-MCramer[2][2]*MCramer[1][4]);
E10=MCramer[1][1]*(MCramer[2][2]*MCramer[3][4]-MCramer[3][2]*MCramer[2][4]);
E11=MCramer[2][1]*(MCramer[1][2]*MCramer[3][4]-MCramer[3][2]*MCramer[1][4]);
E12=MCramer[3][1]*(MCramer[1][2]*MCramer[2][4]-MCramer[2][2]*MCramer[1][4]);

DC=(Const[1]*(E1-E2+E3))- (Const[2]*(E4-E5+E6))+ (Const[3]*(E7-E8+E9))- (Const[4]*(E10-E11+E12));

F1=MCramer[2][1]*(MCramer[3][2]*MCramer[4][3]-MCramer[4][2]*MCramer[3][3]);
F2=MCramer[3][1]*(MCramer[2][2]*MCramer[4][3]-MCramer[4][2]*MCramer[2][3]);

```

```

F3=MCramer[4][1]*(MCramer[2][2]*MCramer[3][3]-MCramer[3][2]*MCramer[2][3]);
F4=MCramer[1][1]*(MCramer[3][2]*MCramer[4][3]-MCramer[4][2]*MCramer[3][3]);
F5=MCramer[3][1]*(MCramer[1][2]*MCramer[4][3]-MCramer[4][2]*MCramer[1][3]);
F6=MCramer[4][1]*(MCramer[1][2]*MCramer[3][3]-MCramer[3][2]*MCramer[1][3]);
F7=MCramer[1][1]*(MCramer[2][2]*MCramer[4][3]-MCramer[4][2]*MCramer[2][3]);
F8=MCramer[2][1]*(MCramer[1][2]*MCramer[4][3]-MCramer[4][2]*MCramer[1][3]);
F9=MCramer[4][1]*(MCramer[1][2]*MCramer[2][3]-MCramer[2][2]*MCramer[1][3]);
F10=MCramer[1][1]*(MCramer[2][2]*MCramer[3][3]-MCramer[3][2]*MCramer[2][3]);
F11=MCramer[2][1]*(MCramer[1][2]*MCramer[3][3]-MCramer[3][2]*MCramer[1][3]);
F12=MCramer[3][1]*(MCramer[1][2]*MCramer[2][3]-MCramer[2][2]*MCramer[1][3]);

DD=(Const[1]*(-F1+F2-F3))+(Const[2]*(F4-F5+F6))+(Const[3]*(-F7+F8-F9))+(Const[4]*(F10-F11+F12));

CoefSolucion[NumFila-1][0] = DA / D;
CoefSolucion[NumFila-1][1] = DB / D;
CoefSolucion[NumFila-1][2] = DC / D;
CoefSolucion[NumFila-1][3] = DD / D;

return(0);
}

/*-----*/
int PideDatos()
/*-----*/
{
    k = 0;
    k = IniciaArreglos();
    sprintf(NumeroEnCadena,"k: %i",k);
    DespliegaCadena(NumeroEnCadena,atrr4,20,20);
    NumeroDatos = -1;
    NumeroCorrecto = -1;
    while(NumeroDatos <= 0 || NumeroDatos > 50)
    {
        Ventana(18,0,20,79,atrr1,1,temp);
        DespliegaCadena("Número de Pares de Datos:",atrr1,19,3);
        NumeroCorrecto = CapturaNumero(NumeroEnCadena,19,29,3,atrr0);
        if (NumeroCorrecto >= 0) NumeroDatos = atoi(NumeroEnCadena);
        else if (NumeroCorrecto===-2) return(-1);
    }

    BorraBloque(0,0,23,79,atrr2);
    Ventana(0,0,18,79,atrr1,1,temp);
    DespliegaCadena("Tiempo  Altura",atrr1,1,1);

    Linea = 1;
    ColumnaTiempo = 1;
    ColumnaAltura = 11;
    for(i = 0; i <= (NumeroDatos - 1); i++)
    {
        if (i == 16)
        {
            Linea = 2;
            ColumnaTiempo = 21;
            ColumnaAltura = 31;
            DespliegaCadena("Tiempo  Altura",atrr1,1,21);
        }
        else
        {
            if (i == 32)
            {
                Linea = 2;
                ColumnaTiempo = 41;
                ColumnaAltura = 51;
                DespliegaCadena("Tiempo  Altura",atrr1,1,41);
            }
            else
            {
                Linea ++;
            }
        }
    }
}

```

```

    }
    Tiempo[i] = RetornaDbl(1,Linea,ColumnaTiempo,9);
    if (NumeroCorrecto===-2) return(-1);
    Altura[i] = RetornaDbl(1,Linea,ColumnaAltura,9);
    if (NumeroCorrecto===-2) return(-1);
}

DatosCurva = 0;
while (DatosCurva == 0)
{
    BorraBloque(0,0,20,79,atr2);
    Ventana(0,0,5,79,atr1,1,temp);
    DespliegaCadena("Pares de Datos que se aproximan a la linealidad:",atr1,1,3);
    NumeroCorrecto = -1;
    LinealInicio = 0;
    while(LinealInicio <= 0 || LinealInicio > NumeroDatos)
    {
        DespliegaCadena("Al Inicio:",atr1,2,3);
        NumeroCorrecto = CapturaNumero(NumeroEnCadena,2,14,3,atr0);
        if (NumeroCorrecto >= 0) LinealInicio = atoi(NumeroEnCadena);
        else if (NumeroCorrecto===-2) return(-1);
    }
    NumeroCorrecto = -1;
    LinealFin = 0;
    while(LinealFin <= 0 || LinealFin > (NumeroDatos - LinealInicio))
    {
        DespliegaCadena("Al Final:",atr1,3,3);
        NumeroCorrecto = CapturaNumero(NumeroEnCadena,3,14,3,atr0);
        if (NumeroCorrecto >= 0) LinealFin = atoi(NumeroEnCadena);
        else if (NumeroCorrecto===-2) return(-1);
    }
    DatosCurva = NumeroDatos - (LinealInicio + LinealFin);
    if (DatosCurva == 0)
    {
        BorraBloque(0,0,20,79,atr2);
        Ventana(10,0,15,79,atr1,1,temp);
        DespliegaCadena("El número de datos que se acercan a la linealidad iguala el número",atr1,11,3);
        DespliegaCadena("total de pares de datos ingresados. Reintente ingresar el valor de",atr1,12,3);
        DespliegaCadena("los pares de datos que se acercan a la linealidad.",atr1,13,3);
        while(!kbhit());
    }
}

BorraBloque(0,0,20,79,atr2);
Ventana(0,0,11,79,atr1,1,temp);
DespliegaCadena("Calculando Coeficientes en Recta Bisectriz",atr1,1,3);
for (i = LinealInicio; i <= (LinealInicio + DatosCurva - 1); i++)
{
    PromedioAltura = PromedioAltura + Altura[i];
}
PromedioAltura = PromedioAltura / DatosCurva;

SumaTiempoR1 = SumaAlturaR1 = ProductoTAR1 = ProductoT2R1 = ProductoA2R1 = 0;
SumaTiempoR2 = SumaAlturaR2 = ProductoTAR2 = ProductoT2R2 = ProductoA2R2 = 0;
for (i = 0; i <= (LinealInicio - 1); i++)
{
    SumaTiempoR1 = SumaTiempoR1 + Tiempo[i];
    SumaAlturaR1 = SumaAlturaR1 + Altura[i];
    ProductoTAR1 = ProductoTAR1 + (Tiempo[i] * Altura[i]);
    ProductoT2R1 = ProductoT2R1 + (Tiempo[i] * Tiempo[i]);
    ProductoA2R1 = ProductoA2R1 + (Altura[i] * Altura[i]);
}
CoefA[0] = (ProductoT2R1 * SumaAlturaR1) - (SumaTiempoR1 * ProductoTAR1);
CoefA[0] = CoefA[0] / ((LinealInicio * ProductoT2R1) - (SumaTiempoR1 * SumaTiempoR1));
CoefB[0] = (LinealInicio * ProductoTAR1) - (SumaTiempoR1 * SumaAlturaR1);
CoefB[0] = CoefB[0] / ((LinealInicio * ProductoT2R1) - (SumaTiempoR1 * SumaTiempoR1));
CoefR[0] = (LinealInicio * ProductoTAR1) - (SumaTiempoR1 * SumaAlturaR1);
CoefR[0] = CoefR[0] / sqrt(((LinealInicio * ProductoT2R1) - pow(SumaTiempoR1,2)) * ((LinealInicio * ProductoA2R1) - pow(SumaAlturaR1,2)));

```

```

for(i = (NumeroDatos - 1); i >= (NumeroDatos - LinealFin); i-)
{
    SumaTiempoR2 = SumaTiempoR2 + Tiempo[i];
    SumaAlturaR2 = SumaAlturaR2 + Altura[i];
    ProductoTAR2 = ProductoTAR2 + (Tiempo[i] * Altura[i]);
    ProductoT2R2 = ProductoT2R2 + (Tiempo[i] * Tiempo[i]);
    ProductoA2R2 = ProductoA2R2 + (Altura[i] * Altura[i]);
}
CoefA[1] = (ProductoT2R2 * SumaAlturaR2) - (SumaTiempoR2 * ProductoTAR2);
CoefA[1] = CoefA[1] / ((LinealFin * ProductoT2R2) - (SumaTiempoR2 * SumaTiempoR2));
CoefB[1] = (LinealFin * ProductoTAR2) - (SumaTiempoR2 * SumaAlturaR2);
CoefB[1] = CoefB[1] / ((LinealFin * ProductoT2R2) - (SumaTiempoR2 * SumaTiempoR2));
CoefR[1] = (LinealFin * ProductoTAR2) - (SumaTiempoR2 * SumaAlturaR2);
CoefR[1] = CoefR[1] / sqrt(((LinealFin * ProductoT2R2) - pow(SumaTiempoR2,2)) * ((LinealFin * ProductoA2R2) -
pow(SumaAlturaR2,2)));

Factor1 = sqrt((pow(CoefB[1],2) + 1)/(pow(CoefB[0],2) + 1));
CoefA[2] = CoefB[0] * Factor1 + CoefB[1];
CoefB[2] = -1 * Factor1 - 1;
CoefR[2] = CoefA[0] * Factor1 + CoefA[1];

if ((CoefA[2] / (-1 * CoefB[2])) < 0)
{
    CoefA[2] = CoefB[0] * Factor1 - CoefB[1];
    CoefB[2] = -1 * Factor1 + 1;
    CoefR[2] = CoefA[0] * Factor1 - CoefA[1];
}

DespliegaCadena("Calculando Sumatorias",atr1,2,3);
for(i = LinealInicio; i < (NumeroDatos - LinealFin); i++)
{
    Total[0] = Total[0] + Tiempo[i];           /* at */
    Total[1] = Total[1] + Altura[i];          /* aa */
    Total[2] = Total[2] + pow(Tiempo[i],2);   /* aty */
    Total[3] = Total[3] + pow(Tiempo[i],3);   /* at3 */
    Total[4] = Total[4] + pow(Tiempo[i],4);   /* at4 */
    Total[5] = Total[5] + pow(Tiempo[i],5);   /* at5 */
    Total[6] = Total[6] + pow(Tiempo[i],6);   /* at6 */
    Total[7] = Total[7] + (1 / Tiempo[i]);     /* a(1/t) */
    Total[8] = Total[8] + (1 / pow(Tiempo[i],2)); /* a(1/t^2) */
    Total[9] = Total[9] + (1 / pow(Tiempo[i],3)); /* a(1/t^3) */
    Total[10] = Total[10] + (1 / pow(Tiempo[i],4)); /* a(1/t^4) */
    Total[11] = Total[11] + (Tiempo[i] * Altura[i]); /* a(ta) */
    Total[12] = Total[12] + (pow(Tiempo[i],2) * Altura[i]); /* a(t^2*a) */
    Total[13] = Total[13] + (pow(Tiempo[i],3) * Altura[i]); /* a(t^3*a) */
    Total[14] = Total[14] + (Altura[i] / Tiempo[i]); /* a(a/t) */
    Total[15] = Total[15] + (Tiempo[i] * log(Tiempo[i])); /* a(t*ln(t)) */
    Total[16] = Total[16] + (pow(Tiempo[i],2) * log(Tiempo[i])); /* a(t^2*ln(t)) */
    Total[17] = Total[17] + (Tiempo[i] * log(Tiempo[i]) * Altura[i]); /* a(t*ln(t)*a) */
    Total[18] = Total[18] + (pow(log(Tiempo[i],2) * pow(Tiempo[i],2)); /* a(tylny(t)) */
    Total[19] = Total[19] + (Altura[i] / pow(Tiempo[i],2)); /* a(a/ty) */
    Total[20] = Total[20] + sqrt(Tiempo[i]); /* a(ut) */
    Total[21] = Total[21] + (Tiempo[i] * sqrt(Tiempo[i])); /* a(tut) */
    Total[22] = Total[22] + (Altura[i] * sqrt(Tiempo[i])); /* a(aut) */
}

DespliegaCadena("Aplicando Cramer en ecuación y= a + bx + cdx(x).",atr1,3,3);
/* Ecuación y= a + bx + cdx(x) NEW!!! */
Const[0] = Total[1];
MCramer[0][0] = NumeroDatos - LinealInicio - LinealFin;
MCramer[0][1] = Total[0];
MCramer[0][2] = Total[15];
MCramer[0][3] = 0;
Const[1] = Total[11];
MCramer[1][0] = Total[0];
MCramer[1][1] = Total[2];
MCramer[1][2] = Total[16];
MCramer[1][3] = 0;
Const[2] = Total[17];
MCramer[2][0] = Total[15];

```

```

MCramer[2][1] = Total[16];
MCramer[2][2] = Total[18];
MCramer[2][3] = 0;
Const[3] = 0;
MCramer[3][0] = 0;
MCramer[3][1] = 0;
MCramer[3][2] = 0;
MCramer[3][3] = 0;
k = Cramer3(1);

DespliegaCadena("Aplicando Cramer en ecuaci3n y = a + bx + cx y + d/x",attr1,4,3);
/* Ecuaci3n y = a + bx + cx y + d/x NEW!!! */
Const[0] = Total[1];
MCramer[0][0] = NumeroDatos - LinealInicio - LinealFin;
MCramer[0][1] = Total[0];
MCramer[0][2] = Total[2];
MCramer[0][3] = Total[7];
Const[1] = Total[11];
MCramer[1][0] = Total[0];
MCramer[1][1] = Total[2];
MCramer[1][2] = Total[3];
MCramer[1][3] = NumeroDatos - LinealInicio - LinealFin;
Const[2] = Total[12];
MCramer[2][0] = Total[2];
MCramer[2][1] = Total[3];
MCramer[2][2] = Total[4];
MCramer[2][3] = Total[0];
Const[3] = Total[14];
MCramer[3][0] = Total[7];
MCramer[3][1] = NumeroDatos - LinealInicio - LinealFin;
MCramer[3][2] = Total[0];
MCramer[3][3] = Total[8];
k = Cramer(2);

DespliegaCadena("Aplicando Cramer en ecuaci3n y = a + bx + cx y + dx^3",attr1,5,3);
/* Ecuaci3n y = a + bx + cx y + dx^3 NEW!!! */
Const[0] = Total[1];
MCramer[0][0] = NumeroDatos - LinealInicio - LinealFin;
MCramer[0][1] = Total[0];
MCramer[0][2] = Total[2];
MCramer[0][3] = Total[3];
Const[1] = Total[11];
MCramer[1][0] = Total[0];
MCramer[1][1] = Total[2];
MCramer[1][2] = Total[3];
MCramer[1][3] = Total[4];
Const[2] = Total[12];
MCramer[2][0] = Total[2];
MCramer[2][1] = Total[3];
MCramer[2][2] = Total[4];
MCramer[2][3] = Total[5];
Const[3] = Total[13];
MCramer[3][0] = Total[3];
MCramer[3][1] = Total[4];
MCramer[3][2] = Total[5];
MCramer[3][3] = Total[6];
k = Cramer(3);

DespliegaCadena("Aplicando Cramer en la ecuaci3n y = a + b/x + c/x y",attr1,6,3);
/* Ecuaci3n y = a + b/x + c/x y NEW!!! */
Const[0] = Total[1];
MCramer[0][0] = NumeroDatos - LinealInicio - LinealFin;
MCramer[0][1] = Total[7];
MCramer[0][2] = Total[8];
MCramer[0][3] = 0;
Const[1] = Total[14];
MCramer[1][0] = Total[7];
MCramer[1][1] = Total[8];
MCramer[1][2] = Total[9];
MCramer[1][3] = 0;

```

```

Const[2] = Total[19];
MCramer[2][0] = Total[8];
MCramer[2][1] = Total[9];
MCramer[2][2] = Total[10];
MCramer[2][3] = 0;
Const[3] = 0;
MCramer[3][0] = 0;
MCramer[3][1] = 0;
MCramer[3][2] = 0;
MCramer[3][3] = 0;
k = Cramer3(4);

DespliegaCadena("Aplicando Cramer en ecuaci3n y = a + bx + c0x",attr1,7,3);
/* Ecuaci3n y = a + bx + c0x NEW!!! */
Const[0] = Total[1];
MCramer[0][0] = NumeroDatos - LinealInicio - LinealFin;
MCramer[0][1] = Total[0];
MCramer[0][2] = Total[20];
MCramer[0][3] = 0;
Const[1] = Total[11];
MCramer[1][0] = Total[0];
MCramer[1][1] = Total[2];
MCramer[1][2] = Total[21];
MCramer[1][3] = 0;
Const[2] = Total[22];
MCramer[2][0] = Total[20];
MCramer[2][1] = Total[21];
MCramer[2][2] = Total[0];
MCramer[2][3] = 0;
Const[3] = 0;
MCramer[3][0] = 0;
MCramer[3][1] = 0;
MCramer[3][2] = 0;
MCramer[3][3] = 0;
k = Cramer3(5);

DespliegaCadena("Determinaci3n del r m s cercano.",attr1,8,3);
/* C lculo de r para la primera ecuaci3n NEW!!! */
SumaDif1 = SumaDif2 = 0;
for (i = LinealInicio; i <= (LinealInicio + DatosCurva - 1); i++)
{
  AlturaCalculada[i] = CoefSolucion[0][0] + (CoefSolucion[0][1] * Tiempo[i])
                    + (CoefSolucion[0][2] * Tiempo[i] * log(Tiempo[i]));
  SumaDif1 = SumaDif1 + pow((AlturaCalculada[i] - PromedioAltura),2);
  SumaDif2 = SumaDif2 + pow((Altura[i] - PromedioAltura),2);
}
CoefSolucion[0][4] = SumaDif1 / SumaDif2;

/* C lculo de r para la segunda ecuaci3n NEW!!! */
SumaDif1 = SumaDif2 = 0;
for (i = LinealInicio; i <= (LinealInicio + DatosCurva - 1); i++)
{
  AlturaCalculada[i] = CoefSolucion[1][0] + (CoefSolucion[1][1] * Tiempo[i])
                    + (CoefSolucion[1][2] * pow(Tiempo[i],2))
                    + (CoefSolucion[1][3] / Tiempo[i]);
  SumaDif1 = SumaDif1 + pow((AlturaCalculada[i] - PromedioAltura),2);
  SumaDif2 = SumaDif2 + pow((Altura[i] - PromedioAltura),2);
}
CoefSolucion[1][4] = SumaDif1 / SumaDif2;

/* C lculo de r para la tercera ecuaci3n NEW!!! */
SumaDif1 = SumaDif2 = 0;
for (i = LinealInicio; i <= (LinealInicio + DatosCurva - 1); i++)
{
  AlturaCalculada[i] = CoefSolucion[2][0] + (CoefSolucion[2][1] * Tiempo[i])
                    + (CoefSolucion[2][2] * pow(Tiempo[i],2))
                    + (CoefSolucion[2][3] * pow(Tiempo[i],3));
  SumaDif1 = SumaDif1 + pow((AlturaCalculada[i] - PromedioAltura),2);
  SumaDif2 = SumaDif2 + pow((Altura[i] - PromedioAltura),2);
}

```

```

CoefSolucion[2][4] = SumaDif1 / SumaDif2;

/* C lculo de r para la cuarta ecuaci3n NEW!!! */
SumaDif1 = SumaDif2 = 0;
for (i = LinealInicio; i <= (LinealInicio + DatosCurva - 1); i++)
{
    AlturaCalculada[i] = CoefSolucion[3][0] + (CoefSolucion[3][1] / Tiempo[i])
                        + (CoefSolucion[3][2] / pow(Tiempo[i],2));
    SumaDif1 = SumaDif1 + pow((AlturaCalculada[i] - PromedioAltura),2);
    SumaDif2 = SumaDif2 + pow((Altura[i] - PromedioAltura),2);
}
CoefSolucion[3][4] = SumaDif1 / SumaDif2;

/* C lculo de r para la quinta ecuaci3n NEW!!! */
SumaDif1 = SumaDif2 = 0;
for (i = LinealInicio; i <= (LinealInicio + DatosCurva - 1); i++)
{
    AlturaCalculada[i] = CoefSolucion[4][0] + (CoefSolucion[4][1] * Tiempo[i])
                        + (CoefSolucion[4][2] * sqrt(Tiempo[i]));
    SumaDif1 = SumaDif1 + pow((AlturaCalculada[i] - PromedioAltura),2);
    SumaDif2 = SumaDif2 + pow((Altura[i] - PromedioAltura),2);
}
CoefSolucion[4][4] = SumaDif1 / SumaDif2;

/* Escogemos el r m s cercano a 1 */
ValorR = 0;
Dif = fabs(CoefSolucion[0][4] - 1);
for (i = 1; i <= 4; i++)
{
    if (fabs(CoefSolucion[i][4] - 1) < Dif)
    {
        Dif = fabs(CoefSolucion[i][4] - 1);
        ValorR = CoefSolucion[i][4];
        FilaR = i;
    }
}

DespliegaCadena("Calculando la intersecci3n de la bisectriz y la curva.",atr1,9,3);
/* Hallamos la intersecci3n entre la bisectriz y la curva */
DifIntersec = 1;
ValorX = 0;
ValorYRecta = ValorYCurva = 0;
while (DifIntersec > 0.01)
{
    ValorX = ValorX + 0.0001;
    switch(FilaR)
    {
        case 0:
        {
            ValorYCurva = CoefSolucion[0][0] + (CoefSolucion[0][1] * ValorX)
                        + (CoefSolucion[0][2] * ValorX * log(ValorX));
            break;
        }
        case 1:
        {
            ValorYCurva = CoefSolucion[1][0] + (CoefSolucion[1][1] * ValorX)
                        + (CoefSolucion[1][2] * pow(ValorX,2))
                        + (CoefSolucion[1][3] / ValorX);
            break;
        }
        case 2:
        {
            ValorYCurva = CoefSolucion[2][0] + (CoefSolucion[2][1] * ValorX)
                        + (CoefSolucion[2][2] * pow(ValorX,2))
                        + (CoefSolucion[2][3] * pow(ValorX,3));
            break;
        }
        case 3:
        {
            ValorYCurva = CoefSolucion[3][0] + (CoefSolucion[3][1] / ValorX)

```

```

        + (CoefSolucion[3][2] / pow(ValorX,2));
    }
    break;
}
case 4:
{
    ValorYCurva = CoefSolucion[4][0] + (CoefSolucion[4][1] * ValorX)
        + (CoefSolucion[4][2] * sqrt(ValorX));
    break;
}
}
ValorYRecta = (CoefA[2] * ValorX + CoefR[2]) / CoefB[2];
DifIntersec = fabs(ValorYRecta - ValorYCurva);
}

DespliegaCadena("Calculando la pendiente en el punto de intersección.",attr1,10,3);
/* Hallamos la pendiente de la curva en el punto de intersección */
switch(FilaR)
{
    case 0:
    {
        Pendiente = CoefSolucion[0][1] + CoefSolucion[0][2] * (1 + log(ValorX));
        break;
    }
    case 1:
    {
        Pendiente = CoefSolucion[1][1] + (2 * CoefSolucion[1][2] * ValorX)
            - (CoefSolucion[1][3] / pow(ValorX,2));
        break;
    }
    case 2:
    {
        Pendiente = CoefSolucion[2][1] * (2 * CoefSolucion[2][2] * ValorX)
            + (3 * CoefSolucion[2][3] * pow(ValorX,2));
        break;
    }
    case 3:
    {
        Pendiente = (-1 * CoefSolucion[3][1] / pow(ValorX,2))
            - (2 * CoefSolucion[3][2] / pow(ValorX,3));
        break;
    }
    case 4:
    {
        Pendiente = CoefSolucion[4][1] + (CoefSolucion[4][2] / (2*sqrt(ValorX)));
        break;
    }
}

BorraBloque(0,0,23,79,atr2);
Ventana(0,0,11,79,atr1,1,temp);
DespliegaCadena("Altura Inicial de la Interfase (m):",atr1,1,3);
AlturaInicial = RetomaDb(1,1,39,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Concentración Inicial de la solución (Kg/m3):",atr1,2,3);
ConcentracionInicial = RetomaDb(1,2,50,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Concentración final de la solución (Kg/m3):",atr1,3,3);
ConcentracionFinal = RetomaDb(1,3,48,9);
if (NumeroCorrecto==2) return(-1);

AlturaFinal = ConcentracionInicial * AlturaInicial / ConcentracionFinal;

TiempoBuscado = (AlturaFinal - (ValorYRecta - (Pendiente * ValorX))) / Pendiente;
DespliegaCadena("Caudal (m3/hora):",atr1,4,3);
Caudal = RetomaDb(1,4,21,9);
if (NumeroCorrecto==2) return(-1);
Area = Caudal * TiempoBuscado / AlturaInicial;
sprintf(NumeroEnCadena,"Area %5.3f m²",Area);
DespliegaCadena(NumeroEnCadena,atr4,5,3);
sprintf(NumeroEnCadena,"Zona de Compresión: %5.3f (m)",AlturaFinal);

```

```

DespliegaCadena(NumeroEnCadena,attr1,6,3);
DespliegaCadena("Inclinaci3n de Fondo: 0.61 (m)",attr1,7,3);
DespliegaCadena("Capacidad de Almacenamiento: 0.61 (m)",attr1,8,3);
DespliegaCadena("Sumersi3n de la Alimentaci3n: 0.61 (m)",attr1,9,3);
ProfundidadTotal = AlturaFinal + 3 * 0.61;
sprintf(NumeroEnCadena,"Profundidad Total: %5.3f m",ProfundidadTotal);
DespliegaCadena(NumeroEnCadena,attr4,10,3);

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,attr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getche();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(19,3,19,70,attr2);
        DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5,"\n\n***** Sedimentacion *****\n");
    fprintf(pfile5,"Area          : %5.3f m2 \n",Area);
    fprintf(pfile5,"Zona de Compresion      : %5.3f m \n",AlturaFinal);
    fprintf(pfile5,"Inclinacion de Fondo   : 0.61 m \n");
    fprintf(pfile5,"Capacidad de Almacenamiento : 0.61 m \n");
    fprintf(pfile5,"Sumersion de la Alimentacion: 0.61 m \n");
    fprintf(pfile5,"Profundidad Total      : %5.3f m \n",ProfundidadTotal);
    fclose(pfile5);
}

CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
return(0);
}

/*-----*/
void Sedimentacion(void)
/*-----*/
{
    /* Definicion temporal de atributos --- */
    attr3=WHITE|(5<<4); /* Atributos del Mensaje abajo en la Pantalla */
    attr2=WHITE|(BLACK<<4); /* Atributos del Marco de Ventana Blanco/Negro */
    attr1=WHITE|(BLACK<<4); /* Atributos del Texto Desactivado Blanco/Negro */
    attr0=WHITE|(BLUE<<4); /* Atributos del Texto Activado Blanco/Azul */
    BorraBloque(0,0,23,79,attr1);
    /* Este atributo es propio ----- */
    attr4=WHITE|(MAGENTA<<4); /* Atributos del Texto para respuestas Blanco/Cyan */
    /*-----*/
    Respuesta[0] = ' ';
    strcpy(temp,"Advertencia!!!");
    while(Respuesta[0] != 'S' && Respuesta[0] != 'N')
    {
        Ventana(5,0,11,79,attr1,1,temp);
        DespliegaCadena("Los puntos que el usuario ingresar en esta opci3n deben haber sido",attr1,6,3);
        DespliegaCadena("previamente graficados y evaluados, preferiblemente en papel",attr1,7,3);
        DespliegaCadena("milimetrado, para rechazar aquellos puntos que no se ajusten a la",attr1,8,3);
        DespliegaCadena("Curva de Sedimentaci3n y que los resultados sean congruentes.",attr1,9,3);
        DespliegaCadena("Desea Continuar (S/N)?",attr1,10,27);
        Cursor(10,51);
        Respuesta[0] = getche();
    }
    if (Respuesta[0] == 'S')

```

```
{
  strcpy(temp,"");
  PideDatos();
}
BorraBloque(0,0,23,79,attr2);
Ventana(0,0,10,79,attr1,1,temp);
}

void main(void)
{
  Sedimentacion();
  BorraBloque(0,0,23,79,attr2);
}
```

CAPITULO II

TRATAMIENTOS BIOLÓGICOS - LAGUNAS DE ESTABILIZACIÓN

2.1 GENERALIDADES

Se reconoce generalmente que el tratamiento biológico en alguna de sus formas es la solución más económica en el caso de las aguas domésticas y en la mayoría de las aguas industriales (Gloyna, 1973).

Durante los últimos 15 años se ha venido desarrollando, perfeccionando y utilizando una forma de tratamiento biológico. Las denominadas lagunas de estabilización son la solución más adecuada para lugares donde el suelo no es caro, las cargas orgánicas fluctúan, existen restricciones económicas y hay escasez de personal preparado (Rivas Mijares, 1978).

El tipo de laguna de estabilización depende de los objetivos de tratamiento. Un sistema de lagunas o estanques se proyecta usualmente para recubrir aguas domésticas o industriales sin un tratamiento previo, pero también puede preverse para tratar efluentes primarios o secundarios de fábricas, fangos activados en exceso o fango diluido de pozos negros.

Una inspección de costos demuestra que es considerablemente más barato tratar aguas residuales en lagunas de estabilización que por cualquier otro procedimiento, con tal de que los costos del terreno no sean prohibitivos. Instalaciones de este tipo prestan hoy servicios en industrias como refinería de petróleo, mataderos, centrales lecheras, plantas de preparación de aves, etc.

Las lagunas de estabilización como su nombre lo indica, son en realidad estanques o reservorios excavados en el suelo y provistos con una protección, si este fuera el caso, para las paredes y fondo que limitan su volumen. En consecuencia, a pesar de los volúmenes y de las áreas requeridas, estas unidades de tratamiento pueden ser realizadas a muy bajos costos para su operación y mantenimiento (Rivas Mijares, 1978).

Otra definición para las lagunas de estabilización es que son estructuras sencillas de tierra, abiertas al sol y al aire, son poco profundas y la finalidad es el tratamiento del agua residual (Fair, 1993).

Las lagunas ó estanques pueden utilizarse para el pretratamiento de aguas residuales, para reducir la mayor parte de la demanda bioquímica de oxígeno (DBO), así como para reducir la concentración de agentes patógenos (Metcalf, 1977).

En las lagunas de estabilización de aguas residuales los residuos orgánicos a descomponer son estabilizados por microorganismos y el número de agentes patógenos

se reduce de un modo notable, principalmente debido al largo período de detención requerido por la estabilización (Gloyna, 1973).

2.1.1 RESEÑA HISTORICA

Las lagunas han sido utilizadas desde hace siglos para el tratamiento de las aguas residuales de origen humano o animal. Sin embargo, tan sólo en las últimas décadas se han formulado criterios de cálculo a base de las exigencias volumétricas de las cantidades adecuadas de carga orgánica y de períodos de detención.

En 1901 en Texas se construyó un estanque conocido como "Mitchell Lake" de 275 ha de superficie y 1.4 m de profundidad.

En 1924 en Santa Rosa, California se hizo un embalse de aguas residuales que tenía unos 90 cm de profundidad. Parece ser que el efluente de este estanque era muy semejante al que se hubiera obtenido de un filtro percolador.

El primer estanque de Dakota del Norte se puso en funcionamiento en 1928 en vista de que no existía ningún curso de agua que pudiera diluir o transportar el agua residual recogida por el alcantarillado. Este estanque continuó en explotación durante más de 30 años.

Hacia 1960 había en Dakota del Norte más de cien instalaciones y en Texas alrededor de 300 sistemas de estanques, en su mayor parte del tipo de tratamiento secundario. En 1962, había en Estados Unidos 1647 estanques de estabilización en explotación para el tratamiento de aguas residuales de origen industrial o agrícola.

También se han hecho experiencias con éxito en Australia (1962); Nueva Zelanda (1965); Israel (1962); Brasil (1967); Sudafrica (1962); India; Canada (1968).

En una encuesta se demostró que por lo menos 39 países hace uso de estanques de estabilización de aguas residuales. Entre ellos los del continente americano: Canada, Estados Unidos, México, Guatemala, Costa Rica, Nicaragua, Argentina, Brasil, Colombia, Ecuador, Perú, Venezuela, Bolivia y Cuba.

En El Salvador, se tiene conocimiento del funcionamiento de lagunas de estabilización en: "Ciudad de San José, Las Flores", al norte de San Salvador (1977-1978), "Mataderos de El Salvador", Km. 22 carretera a Quezaltepeque (1971-1972), "Textiles de San Andrés, S.A.", carretera a Santa Ana, Km 32, departamento de La Libertad (1974-1975), Un Km al noroeste del Batallón Atlacatl, Km 28 de la carretera Panamericana, departamento de La Libertad (1968-1969), "Regimiento de Caballería", Km 30 1/2, carretera a Santa Ana, departamento de La Libertad (1979), "Brigada de Artillería", carretera a Opico (1977-1978), Un Km al suroeste de Santiago Nonualco, departamento de La Paz (1976-1977).

Actualmente a los sistemas de lagunaje se les conoce como tecnologías blandas.

Una laguna debería ser impermeable para evitar contaminar las aguas subterráneas. Para impermeabilizar de una forma más barata se puede hacer compactando con arcilla o lámina de PVC.

2.2 CARACTERISTICAS Y ESTABILIZACION BIOLOGICA DE LAS AGUAS

2.2.1 NATURALEZA DE LA CONTAMINACION

Los contaminantes pueden dividirse en biodegradables y no biodegradables. Ciertos contaminantes, por ejemplo los inorgánicos, no se degradan biológicamente y una vez que entran en las aguas receptoras pueden diluirse aunque no se reducen necesariamente en cantidad.

Otros contaminantes experimentan modificaciones por la acción de factores biológicos, químicos y físicos, y, a consecuencia de esas modificaciones, pueden convertirse en sustancias inofensivas. De igual modo, algunas aguas residuales de origen industrial pueden transformarse en efluentes estables.

Las sustancias y los microorganismos presentes en las aguas residuales pueden ser: a) agentes infecciosos, b) residuos con demanda de oxígeno, c) nutrientes de plantas, d) compuestos químicos orgánicos, e) productos químicos inorgánicos y sustancias minerales, f) sedimentos, y g) sustancias radiactivas.

2.2.2 CARACTERISTICAS DE LOS RESIDUOS

Las condiciones climáticas y las costumbres sociales no son más que una muestra de los muchos factores que hacen variar el tipo y el volumen de las aguas residuales.

La cantidad de agua que sirve de vehículo a la materia orgánica es tanto mayor cuanto más abundante es el abastecimiento público de agua y más avanzada la industrialización de la colectividad. El contenido de sólidos del agua residual es mayor en las localidades con limitaciones en el suministro de agua que en los casos en que existe una buena red de alcantarillado para arrastrar los residuos.

En las aguas residuales domésticas la materia orgánica puede dividirse en tres tipos principales: proteínas, hidrato de carbono y grasas. Las proteínas, que constituyen el 40 - 50% de la materia orgánica, son complejos de aminoácidos y proporcionan la mayor parte de los nutrientes bacterianos. Los hidratos de carbonos están constituidos por almidones y azúcares fácilmente degradables y también por celulosa, que se degrada con menos facilidad (Gloyna, 1973).

Las grasas, con inclusión de los ácidos grasos, no suelen ser muy solubles y se degradan más lentamente.

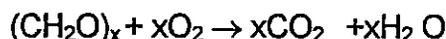
2.2.3 MODO DE DESCOMPOSICION

La descomposición de la materia orgánica puede producirse en condiciones aerobias o anaerobias (Gloyna y Eckenfeder, 1968). El proceso aerobio requiere una aportación continua de oxígeno disuelto libre y es el método más eficaz para reducir el contenido orgánico de los residuos líquidos diluidos. Sin embargo, cuando hay sólidos que han de pasar al estado líquido o cuando la concentración de residuos es muy grande, como en el caso de la materia orgánica sólida sedimentada procedente de aguas domésticas, fangos de pozos negros o residuos de mataderos, el proceso anaerobio resulta extremadamente efectivo.

Proceso aerobio

En el metabolismo aerobio de la materia orgánica, gran parte del carbono sirve de fuente de energía para los microorganismos y, al ser respirado, produce anhídrido carbónico (CO₂). Los microorganismos que intervienen son en su mayoría bacterias, pero también hay hongos y protozoos.

Las principales reacciones que probablemente ocurren en un sistema de estanques aerobios de estabilización son las siguientes (Gloyna, 1973):



Proteínas (N orgánico) → amoníaco → nitritos → nitratos

Azufre orgánico → sulfatos

Fosfatos orgánicos → H₃PO₄ → CaPO₄

La cantidad de oxígeno requerida para estabilizar la materia orgánica de los residuos depende de la DBO satisfecha durante el tratamiento. Esta DBO representa el oxígeno que debe de ser suministrado al estanque mediante proceso de fotosíntesis, la rapidez con que se utiliza el oxígeno constituye un importante factor para el proyecto de una planta de tratamiento.

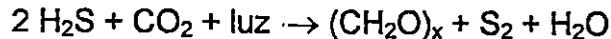
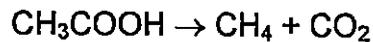
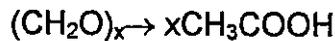
Parte de la materia orgánica se utiliza para formar el protoplasma mientras que otra parte de los residuos se degrada dando lugar a compuestos de bajo poder energético.

Proceso anaerobio

La descomposición de la materia orgánica por putrefacción tiene lugar durante la fermentación anaerobia. Este proceso es bifásico en primer lugar, un grupo especial de bacterias descomponen la materia orgánica en ácidos grasos, aldehídos, alcoholes, etc. Otro grupo de bacterias metanogénicas transforman los productos intermedios en gas metano (CH_4), amoníaco (NH_3), anhídrido carbónico (CO_2) e hidrógeno (H_2). El proceso anaerobio, lo mismo que el aerobio, convierte el carbono, el nitrógeno, el fósforo y otros nutrientes en protoplasma celular (Gloyna, 1973).

El proceso anaerobio también requiere oxígeno, pero este procede de los compuestos químicos y no del oxígeno libre disuelto. En la descomposición anaerobia los productos finales son muy complejos; las reacciones son más lentas y los productos pueden emitir malos olores.

Las reacciones bioquímicas que se producen en la descomposición anaerobia de residuos pueden expresarse como sigue (Gloyna, 1973):



Los estanques en los que se efectúa la estabilización de aguas residuales mediante una combinación de bacterias facultativas, anaerobias y aerobias, se conocen con el nombre de estanques de estabilización aerobios-anaerobios. Como se muestra en la figura 2.1.

2.2.4 REACCIONES BIOLÓGICAS

Las principales reacciones biológicas que se observan en un estanque de estabilización comprenden los siguientes fenómenos: a) oxidación de materia orgánica por bacterias aerobias, b) nitrificación de las proteínas y de otros compuestos nitrogenados por bacterias aerobias, c) reducción de la materia orgánica por bacterias anaerobias presentes en los depósitos y estratos líquidos del fondo, y d) oxigenación de los estratos líquidos superficiales por las algas.

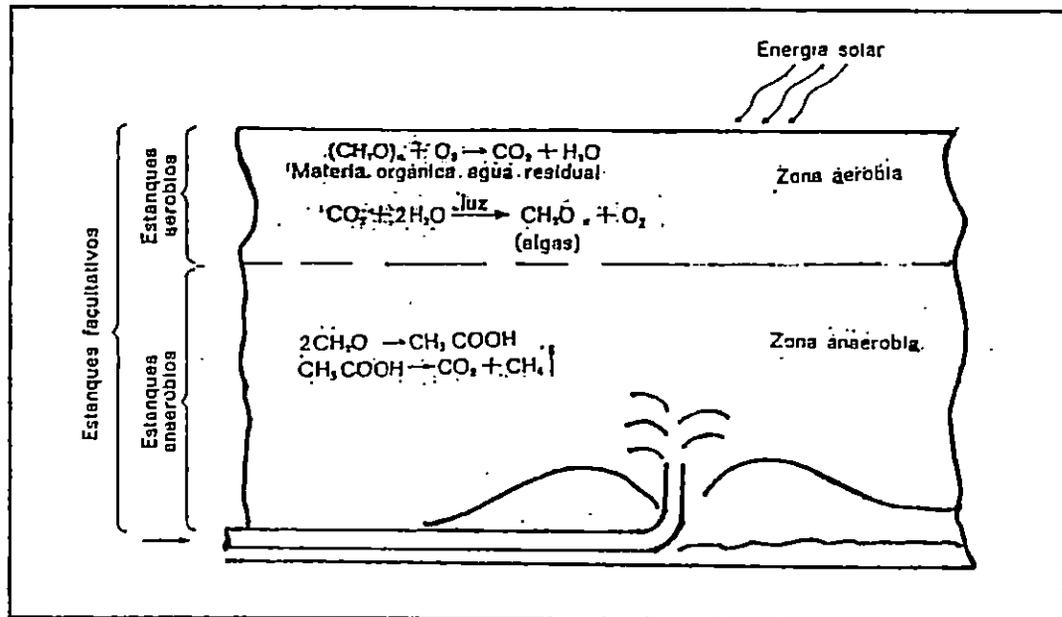


FIGURA 2.1 REPRESENTACION ESQUEMATICA DE UNA LAGUNA DE ESTABILIZACION DE AGUAS RESIDUALES (METCALF, 1977)

Actualmente se aceptan ciertos hechos básicos relativos a la oxidación de los residuos orgánicos: a) el oxígeno disuelto se reduce durante la estabilización de la materia orgánica; b) la rapidez de la oxidación es independiente de la cantidad de oxígeno disuelto disponible; c) el tipo y el número de organismos presentes es importante; y d) los cambios del contenido de oxígeno pueden servir para medir la cantidad y el carácter de la materia orgánica oxidable.

2.2.5 DEMANDA DE OXIGENO

La demanda de oxígeno del agua residual puede expresarse como demanda bioquímica de oxígeno (DBO) y como demanda química de oxígeno (DQO). Aunque estos valores no miden necesariamente lo mismo, ambos dan una indicación muy útil de la demanda total de oxígeno que puede imponer una carga de residuos a una planta de tratamiento o a un curso de agua. La demanda de oxígeno de las materias carbonosas y las nitrogenadas se miden usualmente mediante la determinación de la DBO, mientras que la demanda total se expresa en general en forma de DQO (Gloyne, 1973).

Con el ensayo de la DQO se oxidan todas las sustancias contenidas en el agua residual a excepción de algunos compuestos aromáticos que no son oxidados completamente. En cambio la DBO solo determina la cantidad de carbono orgánico biodegradable por lo que numericamente este ensayo resulta menor que la DQO. Aproximadamente, la DBO es igual al 60% de la DQO (Gloyne, 1973).

2.2.6 DETERMINACION DE LA DEMANDA BIOQUIMICA DE OXIGENO

La determinación de la DBO indica la cantidad de oxígeno disuelto requerida por los microorganismos vivos para la utilización o la destrucción aeróbica de la materia orgánica. En consecuencia, la DBO representa la cantidad de oxígeno disuelto que puede ser necesaria para desarrollar y sostener la actividad biológica precisa para degradar una determinada cantidad de residuos; si el tratamiento aerobio se prolonga, la DBO puede comprender además el oxígeno requerido para degradar parte de la materia biológica celular producida a partir de la carga inicial de residuos.

En las aguas residuales recientes la DBO se desarrolla en dos fases. Durante la primera las materias carbonosas se oxidan intensamente, mientras que durante la segunda etapa se produce una nitrificación. En la primera fase se supone que la tasa de descomposición orgánica en un momento dado es directamente proporcional a la cantidad de material biodegradable presente en ese momento (Figura 2.2).

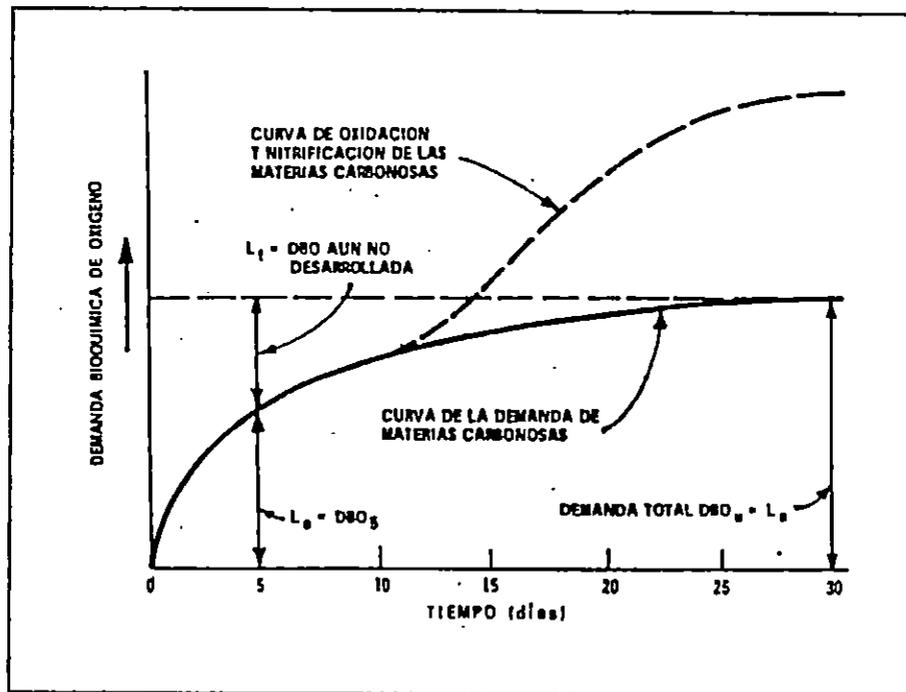


FIGURA 2.2 . DESARROLLO DE LA DEMANDA BIOQUIMICA DE OXIGENO
(GLOYNA, 1973)

La estabilización biológica de una agua residual puede durar mucho tiempo y en la práctica se ha aceptado como valor de referencia la DBO_5 que es la DBO correspondiente a un período de incubación de 5 días.

Con objeto de obtener el valor máximo de DBO es importante también emplear la dilución apropiada para asegurarse que cualquier efecto tóxico queda eliminado y que no se utilizará todo el oxígeno disuelto. Cuando con disoluciones crecientes se obtengan valores

también crecientes de DBO, habrá que aumentar las diluciones hasta conseguir que el nivel de DBO se estabilice en su valor máximo.

La DBO última, o total (DBO_u) es la suma de la DBO de las materias carbonosas ya satisfecha y de la remanente (ecuación 2.1).

$$L_a = L_s + L_t \quad \text{Ec.2.1}$$

En la que

$$L_a = \text{DBO total, DBO}_u \text{ (mg / m }^3 \text{)}$$

$$L_s = \text{DBO satisfecha después de t días (mg / m }^3 \text{)}$$

$$L_t = \text{DBO aún no satisfecha después de t días (mg / m }^3 \text{)}$$

Las ecuaciones 2.2 - 2.4 dan la tasa de crecimiento a la que se satisface la DBO de las materias carbonosas. La constante k depende de las características del agua residual y de la temperatura. En las ecuaciones 2.3 y 2.4 los valores típicos pueden variar de 0.10 a 0.30 para las aguas residuales domésticas, de 0.15 a 0.25 para las aguas residuales domésticas no tratadas o para efluentes de sistemas de tratamiento rápido, de 0.04 a 0.06 para las aguas de corrientes poco contaminadas.

$$-\frac{dL_t}{dt} = K_1 L_t \quad \text{Ec.2.2}$$

$$\frac{L_t}{L_a} = e^{-k_1 t} = 10^{-kt} \quad \text{Ec.2.3}$$

$$LS = L_a - L_t = L_a (1 - 10^{-kt}) \quad \text{Ec.2.4}$$

en la que:

$$t = \text{ tiempo (días)}$$

$$L_a = \text{ DBO}_u \text{ o DBO inicial (mg / m }^3 \text{)}$$

$$K_1 = \text{ Constante de reacción, log e (descomposición/día)}$$

$$K = \text{ Constante de reacción, log}_{10} \text{ (descomposición/día)}$$

La cinética de la reacción de la DBO por razones de tipo práctico, se formula de acuerdo con una reacción de primer orden como se expreso en la ecuación 2.2. Sin embargo, se puede calcular para un orden n (Metcalf, 1977).

Cuando no se conoce el mecanismo de reacción, a veces se intenta ajustar los datos con una ecuación cinética de orden n , de la forma:

$$-\frac{dL_t}{dt} = K_1 L_t^n \quad \text{Ec.2.5}$$

y después de separar variables e integrar se transforma en :

$$L_t^{1-n} - L_a^{1-n} = (n-1)K_1 t, \text{ donde } n \neq 1 \quad \text{Ec.2.6}$$

El orden n no puede calcularse directamente a partir de la Ec.2.6 sino que ha de efectuarse su cálculo por tanteo. Para un valor supuesto de n se calcula k para todos y cada uno de los puntos; el valor de n que de mínima variación en k es el valor buscado.

Las reacciones con orden $n > 1$ no pueden nunca completarse en un tiempo finito. Por otra parte, para ordenes $n < 1$ esta expresión cinética predice que la DBO aún no satisfecha (remanente) se hará cero en un tiempo finito calculado a partir de la Ec. 2.6 y después será negativa (Levenspiel, 1981).

2.2.7 DETERMINACION DE LA DEMANDA QUIMICA DE OXIGENO

La determinación de la demanda química de oxígeno, tal como se efectúa actualmente, proporciona una medida de la cantidad de oxígeno que corresponde a la porción de materia orgánica de una muestra que es susceptible de oxidarse por un producto químico de alto poder oxidante. Si las aguas residuales llevan sustancias tóxicas, esta técnica puede constituir el único método práctico para la determinación de la carga orgánica.

2.2.8 NITRIFICACION

En algunos tratamientos de aguas residuales se exige la eliminación del nitrógeno de los efluentes tratados con objeto de reducir al mínimo la descarga de nutrientes de las plantas en las aguas receptoras, ya que la formación de tales nutrientes en un curso de agua o en un embalse acelera el proceso de envejecimiento conocido con el nombre de eutroficación. Hasta el momento no se ha utilizado con éxito los estanques de estabilización para la desnitrificación (Gloyne, 1973).

2.3 CLASIFICACION DE LAS LAGUNAS

Los estanques o lagunas de estabilización suelen clasificarse según la naturaleza de la actividad biológica que tenga lugar. Otras clasificaciones son según el tipo de afluente (sin tratar, agua residual sedimentada o afluentes de fangos activados, etc.). Respecto a su actividad biológica, los estanques de estabilización se clasifican en aerobios, aerobios-anaerobios y anaerobios. Se han utilizado sistemas diferentes de estanques y aplicados para satisfacer algunos objetivos de determinados tratamientos.

Estanques Aerobios

Un estanque aerobio de estabilización contiene algas y bacterias en suspensión y en su profundidad prevalecen las condiciones aerobias. Básicamente existen dos tipos de estanques aerobios. En el primero el objetivo es elevar al máximo la producción de algas. La profundidad de estos estanques suele limitarse de 15 a 45 cm aproximadamente. En el segundo de estos tipos, el objeto es elevar al máximo la cantidad de oxígeno producido, y en este caso, se utilizan estanques de hasta 1.5 m de profundidad. En ambos tipos, al oxígeno producido por las algas ha de agregarse el transferido al líquido a través de la difusión atmosférica (Metcalf, 1977).

A fin de lograr los mejores resultados posibles en los estanques aerobios, su contenido debe mezclarse periódicamente mediante bombas o aireadores de superficie. La eficiencia de la conversión de DBO en estanques aerobios es elevada, hasta un 95%.

Estanques aerobio-anaerobios

En un estanque aerobio-anaerobio existen tres zonas: 1) una zona superficial donde las algas y bacterias aerobias existen en una relación simbiótica 2) una zona anaerobia inferior en la que los sólidos acumulados son activamente descompuestos por las bacterias anaerobias, y 3) una zona intermedia, en parte aerobia y en parte anaerobia, donde la descomposición de los residuos orgánicos es llevada a cabo por las bacterias facultativas. Por esta razón, a estos estanques se les llaman "facultativos".

En estos estanques se deja que los sólidos en suspensión en el agua residual se depositen al fondo. Como resultado de ello no es preciso que haya algas. El mantenimiento de la zona aerobia sirve para minimizar los problemas de olores, ya que muchos productos de la descomposición anaerobia de los gases y líquidos son llevados a la superficie por las corrientes de mezclado, donde son utilizados por los organismos aerobios. La ventaja de utilizar aireadores superficiales es que pueden aplicarse mayores cargas orgánicas.

Estanques anaerobios

Estos estanques son anaerobios en toda su profundidad excepto en una capa extremadamente delgada en la superficie. Con objeto de conservar la energía térmica y

mantener las condiciones anaerobias, estos estanques se construyen con profundidades de hasta 6 m. La estabilización se consigue mediante una combinación de precipitación y conversión anaerobia de los residuos orgánicos. Eficiencias de conversión de la DBO hasta de un 70% se obtienen de forma rutinaria y, en condiciones óptimas de funcionamiento, es posible obtener eficiencias de eliminación de hasta un 85%.

2.4 CONSIDERACIONES SOBRE EL DISEÑO DEL PROCESO

Antes de hacer un proyecto de instalación concreto es necesario obtener toda la información de base posible acerca del problema. Algunos de los datos requeridos que pueden ser útiles están: clasificación de aguas residuales (domésticas, industriales, agrícolas, etc); características de las aguas residuales (volumen mínimo, concentración de materia orgánica, toxicidad, etc); hidrología y meteorología, luz, topografía, utilización del efluente, etc. (Gloyna, 1973).

La tabla 2.1 presenta datos típicos sobre los parámetros para los distintos tipos de estanques previamente tratados (Rivas Mijares, 1978).

Para precisar conceptos dentro de los parámetros de diseño citados , se presentan las siguientes definiciones:

a) Carga Orgánica (CO) :

$$CO = DBO \text{ (a ser removida en la laguna) } \cdot \text{caudal} \quad \text{Ec.2.7}$$

b) Volumen (V) :

$$V = \text{días} \cdot \text{caudal} \quad \text{Ec.2.8}$$

c) Período de detención (t):

$$t = \text{volumen} / \text{caudal} \quad \text{Ec.2.9}$$

d) Superficie (S):

$$S = \text{volumen} / \text{profundidad} \quad \text{Ec.2.10}$$

e) Carga Hidráulica (CH) :

$$CH = \text{profundidad} / \text{período de retención} \quad \text{Ec.2.11}$$

TABLA 2.1 LAGUNAS DE ESTABILIZACION-CRITERIOS DE DISEÑO
(RIVAS MIJARES, 1978)

Tipo de Laguna	Carga Orgánica (Kg DBO / día)	Período de Retención (días)	Carga Hidráulica (cm / día)	Profundidad (cm)
Aerobias	225 o menor	2 a 4	5 a 12	30 a 60
Anaerobia	225 o mayor	10 a 50	6 o más	250 a 300
Facultativa	56 a 168 (70 tratando líquido crudo; 190 tratando efluente primario)	15 a 100 (5 a 7 para efluentes de tratamiento primario; y de 50 o más para líquido cloacal crudo; 80 o más sin efluente)	2.5 (de 2.5 para líquido crudo; de 7 a 15 para efluente primario)	90 a 150 (90 para efluente primario, 125 para líquido crudo)

La profundidad de la penetración del oxígeno en relación con la carga superficial ha sido , expresada como DBO kg / (ha)(d). Cuanto mayor sea la carga menor será la profundidad de la penetración de oxígeno ya que la demanda de este elemento es superior. La Figura 2.3 muestra una correlación para la penetración de oxígeno en las lagunas de estabilización. La Figura 2.3 se utilizará en el presente proyecto para poder estimar la profundidad a la cual prevalecen las condiciones aerobias y así poder seleccionar la laguna de estabilización más conveniente dependiendo de la carga de DBO (Ramalho, 1991).

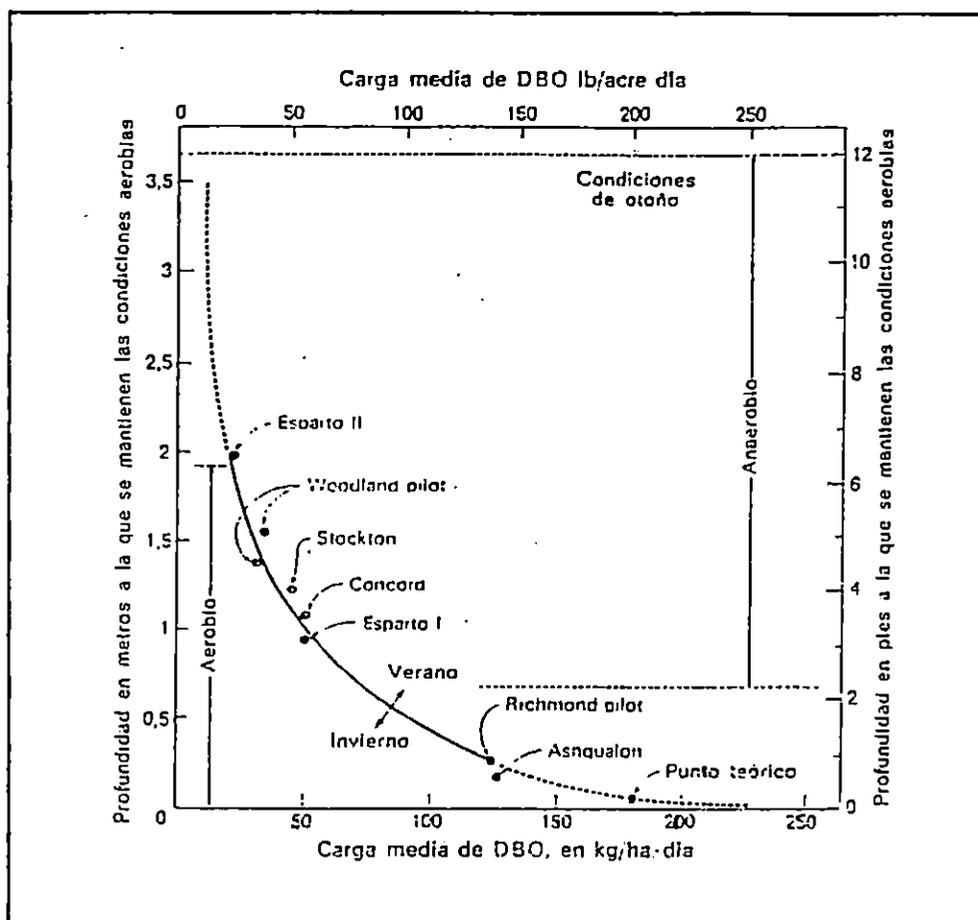


FIGURA 2.3 RELACIÓN ENTRE LA CARGA DE DBO Y LA PROFUNDIDAD (METCALF, 1977).

2.4.1 LAGUNAS AEROBIAS

Entre los factores que deben considerarse en el diseño de estas lagunas es la eliminación de la DBO, así como las necesidades de oxígeno.

La principal fuente de oxígeno en un estanque de estabilización aerobio es la fotosíntesis, que se rige por la energía solar. En base a resultados de numerosos estudios, se ha encontrado que la producción de oxígeno puede calcularse de acuerdo a la siguiente ecuación 2.12 :

$$O = 0.25 \cdot F \cdot I_1$$

Ec.2.12

donde

O : producción de oxígeno, en Kg O₂ ha día

F : factor de oxigenación

I_l : radiación solar, en cal / cm día

El factor de rendimiento de conversión de luz ó factor de oxigenación, se puede encontrar a través de un método empírico en función de la DBO (Oswald,1964; Gloyna,1973).

$$F = 1.3 (\ln CO - 3.5) \quad \text{Ec.2.13}$$

La radiación solar I_l para nuestro país El Salvador, se encontrará en la tabla 2.3, donde cada estación ubicada en diferente lugar del país, proporciona la radiación promedio mensual en $\text{cal/cm}^2\text{-día}$. La tabla 2.2 muestra las abreviaturas de las estaciones y su latitud Norte y longitud oeste. El anexo 1 muestra un mapa de El Salvador, donde se localizan las diferentes estaciones climatológicas.

Experimentalmente se ha encontrado que la producción de células de algas en relación con el producto de la eficiencia de conversión de la luz por radiación solar es:

$$A = 0.15 F \cdot I_l \quad \text{Ec.2.14}$$

en la que:

A : producción de células de algas (Kg de algas / ha por día)

I_l y F : se calcularán igual que en la Ec. 2.12

En estas lagunas se utiliza la energía de luz visible absorbida por cultivos de algas, para el tratamiento de los despojos líquidos residuales.

La presencia de algas en una laguna no garantiza que el oxígeno estará también presente. Sólo cuando el alga lleva a cabo la fotosíntesis (crecimiento activo y producción de material celular), es cuando se produce un exceso de oxígeno molecular. Sino ocurre un crecimiento adecuado, la producción de oxígeno puede aún ser menor a los requerimientos exigidos por las propias algas.

TABLA 2.2 . ABREVIATURAS DE LAS ESTACIONES , SUS ELEVACIONES Y COORDENADAS (ALMANAQUE,CENTRO DE METERELOGIA,1993)

ABREVIATURA	ESTACION	ELEVACION	LATITUD N.	LONGITUD W.
Mo	Montecristo	2230	14°21.2'	89°21.0'
Cv	Cerro Verde	2030	13°49.6'	89°37.4'
PI	Las Pilas	1960	14°21.9'	89°05.4'
LG	La Galera	1900	14°02.6'	88°05.2'
LA	Finca los Andes	1770	13°52.5'	89°38.7'
LN	Los Naranjos	1450	13°52.5'	89°40.5'
AP	Apaneca	1300	13°51.7'	89°46.9'
LP	La Palma	1000	14°17.5'	89°09.7'
ST	Santa Tecla	965	13°41.2'	89°17.3'
SG	Santiago de Ma	920	13°29.1'	88°28.3'
CJ	Cojutepeque	880	13°42.8'	88°55.8'
CO	Corinto	820	13°48.3'	87°58.1'
SN	Santa Ana	725	13°58.6'	89°34.2'
AH	Ahuachapán	725	13°56.6'	89°51.6'
SS	San Salvador	710	13°43.3'	89°12.4'
CP	Candelaria de la F	700	14°07.2'	89°39.1'
IL	Aerop. Ilopango	615	13°41.9'	89°07.1'
AQ	Apastepeque	579	13°42.0'	88°44.8'
GU	Guyja	485	14°13.7'	89°28.7'
SA	San Andrés	460	13°48.5'	89°24.4'
IZ	Izalco	390	13°45.7'	89°42.3'
NC	Nueva Concep.	320	14°07.5"	89°17.4'
FG	San Fco. Gotera	250	13°41.8'	88°06.4'
CH	Chorrera delGuay.	190	13°59.8'	88°45.4'
SM	San Miguel	140	13°26.6'	88°07.4'
IN	Intípica	110	13°12.0'	88°03.4'
UN	La Union	95	13°20.1'	87°52.3'
CA	La Carrera	75	13°19.6'	88°31.5'
AS	Astpria	40	13°27.4'	89°02.7'
SC	Sta.Cruz Pomilo	30	13°26.4'	88°48.2'
SD	San Diego	20	13°28.2'	89°15.2'
AC	Acajutla	15	13°34.4'	89°50.0'

El uso de la tabla anterior (2.2), sirve para obtener la abreviatura de la estación metereológica que este más cercana al lugar donde se pretende construir la laguna. Luego, la tabla 2.3 posee abreviaturas de algunas de las estaciones y dependiendo del mes en que trabajará la laguna se obtendrá el dato de radiación que servirá para encontrar la producción de oxígeno (Ec.2.12) y la producción de algas (Ec.2.14). Esto es para el caso de El Salvador.

Si la laguna se construye en un país determinado existen tablas en la que se ubica la latitud y el mes, obteniendo así el valor de energía solar máximo y mínimo para calcular el valor medio.

TABLA 2.3 PROMEDIOS MENSUALES Y ANUALES DE RADIACIÓN GLOBAL ,CAL X CM² X DÍA 1980. (ALMANAQUE, CENTRO DE METEREOLÓGÍA, 1993)

PROMEDIOS MENSUALES DE RADIACIÓN

	ENE	FEB	MAR	ABR	MAY	JUN
AH	493.0	538.2	561.5	551.2	461.6	444.0
SS	418.4	458.4	479.5	473.0	473.9	421.4
NC	422.0	467.3	489.2	487.4	446.0	455.4
UN	422.3	465.1	489.8	472.0	433.0	442.6
AQ	395.3	445.7	457.3	439.0	414.4	406.5
CA	419.4	460.4	476.2	447.3	414.6	413.1
MO	398.4	427.3	414.3	370.2	309.1	331.3
LG	375.0	420.9	440.4	389.6	389.6	367.3
PI	341.2	296.5	392.3	357.7	332.3	342.5

	JUL	AGO	SEP	OCT	NOV	DIC
AH	487.6	465.8	408.2	423.8	432.2	429.6
SS	473.7	442.4	390.7	403.3	411.5	409.0
NC	472.4	472.3	442.0	438.6	422.0	402.5
UN	465.0	449.3	427.2	427.7	404.1	405.8
AQ	443.0	447.2	392.8	367.0	391.8	376.9
CA	462.4	466.3	414.6	413.5	409.9	413.2
MO	405.7	351.9	294.6	336.6	366.3	369.7
LG	393.7	365.6	314.4	326.5	320.8	328.8
PI	370.5	386.1	298.2	232.2	303.6	308.4

PROMEDIO ANUAL DE RADIACIÓN

AH	SS	NC	UN	AQ	CA	MO	LG	PI
11Años	12Años	11Años	11Años	9 Años	8 Años	5 Años	5 Años	4Años
474.7	432.9	451.4	442.0	416.4	434.2	464.6	366.9	326.8

Para el diseño de lagunas aerobias se necesitan datos como DBO₅, DBO_u, temperatura, caudal, profundidad y otros para poder calcular parámetros de diseño como los de las ecuaciones 2.7- 2.11.

Otros parámetros a calcular son:

$$Q = N \cdot q$$

Ec.2.15

donde:

Q : caudal (m³/ día)

N : número de personas que aportan agua

q : producción de agua residual por persona (m³/ día)

$$CS = CO / S \quad \text{Ec.2.16}$$

donde:

CS : Carga Superficial (Kg / día)

CO: Carga Orgánica (Kg / día)

S : superficie (m²)

La temperatura a la que se determina la DBO de una muestra residual suele ser de 20°C; y el efecto de la temperatura sobre la velocidad de reacción de un proceso biológico se determina por una ecuación aproximada derivada de Arrhenius (Metcalf, 1977) :

$$K_T = K_{20} \cdot \theta^{(T-20)} \quad \text{Ec.2.17}$$

donde:

K_T : velocidad de reacción a T°C

K₂₀ : velocidad de reacción de 20°C

θ : Coeficiente actividad-temperatura

T : temperatura en °C

Los valores de θ para ciertos procesos biológicos, se muestran en la tabla 2.4 (Metcalf,1977).

TABLA 2.4 COEFICIENTES ACTIVIDAD-TEMPERATURA PARA DISTINTOS PROCESOS BIOLÓGICOS (METCALF, 1977)

PROCESOS	θ
Fango activado	1.0 - 1.03
Filtro percoladores	1.02 - 1.04
lagunas de aireación	1.06 - 1.09

Si se tiene un porcentaje de DBO afluente remanente y admitiendo que en la práctica ni el reactor de flujo en pistón ni el de mezcla completa funcionan como se ha supuesto, Wehner y Wilhem presentaron la siguiente ecuación para un reactor con dispersión axial, cinética de primer orden y condiciones arbitrarias de entrada y salida:

$$L_t = \frac{4ae^{(1/2p)}}{(1+a)^2 e^{(a/2p)} - (1-a)^2 e^{(-a/2p)}} \quad \text{EC.2.18}$$

donde:

L_t : DBO remanente (%)

$$a : \sqrt{1 + 4ktp}$$

p : factor de dispersión axial = D / ul

D : coeficiente de dispersión axial, $m^2 / \text{día}$

u : velocidad del fluido, m / h

l : longitud característica m

k : constante de reacción de primer orden

t : tiempo de detención, días

A fin de facilitar el uso de la Ec. 2.18, Thirumurthi, de acuerdo con su trabajo de estanques de estabilización, desarrolló la Figura 2.4, en la que kt se relaciona gráficamente con L_t para factores de dispersión que varían desde cero en el caso de un reactor con flujo en pistón ideal, hasta infinito para un reactor de mezcla completa (Metcalf, 1977).

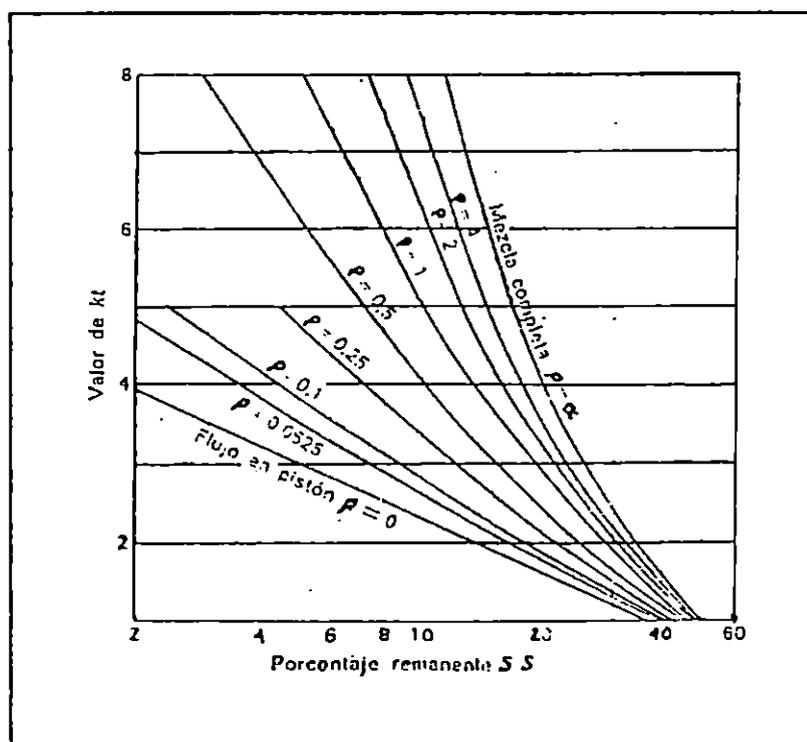


FIGURA 2.4 RELACIÓN ENTRE KT Y PORCENTAJE DE SUBSTRATO REMANENTE PARA VARIOS FACTORES DE DISPERSIÓN (METCALF, 1977)

Lagunas Aireadas

Son lagunas con profundidades de 1 a 4 metros en las que la oxigenación de las aguas residuales se realiza mediante unidades de aireación bien sea superficiales, turbinas o difusores (Ramalho, 1991).

Los aireadores de superficie pueden fijarse en determinada posición o utilizarse como unidades flotantes. Generalmente, un 10-20% del oxígeno total requerido se obtiene por transferencia a través de la interfase superficial y el resto por mezcla y por atrapamiento de aire. Las unidades de aireación mecánica de superficie pueden suministrar 1-1,8 kg de oxígeno/CVh (caballo de vapor-hora). Cuanto más grande es el estanque, menor es el suministro.

El oxígeno suministrado por aireación mecánica se determina usualmente como sigue:

$$O_s = O_m \frac{C_{sw} - C}{C_s} \cdot \alpha \cdot \theta^{(T-20)} F_a \quad \text{Ec.2.19}$$

en donde:

- O_s : oxígeno suministrado por aireación mecánica (Kg de O_2 / Cvh)
- O_m : rendimiento del aireador dado por el fabricante (usualmente 1,8-2,1 kg de oxígeno / Cvh)
- C_{sw} : nivel de saturación de O_2 en el estanque a la temperatura T (Kg / m^3)
- C : nivel de oxígeno disuelto en el estanque (Kg / m^3)
- C_s : nivel de saturación de O_2 en agua destilada a 20°C (Kg / m^3)
- α : coeficiente de transferencia total del agua residual sobre el agua de grifo
(típicamente, los valores oscilan entre 0,6 y 1,1)
- θ : coeficiente de reacción de la temperatura = 1,02
- F_a : factor de corrección para altitudes superiores a 1200 m (utilizar relaciones normalizadas de densidad del aire)

La ecuación 2.20 puede usarse para calcular las necesidades de oxígeno en un sistema aerobio con mezcla completa. Estos estanques necesitan oxígeno para la DBO causada por los microorganismos que están degradando el agua residual que entra y para degradar también parte de los sólidos biológicos volátiles acumulados.

$$O_n = a'(L_o - L_p) + b' X_t \quad \text{Ec.2.20}$$

donde:

- O_n : oxígeno requerido (Kg / m^3)
- a' : DBO reducida y utilizada para proporcionar energía de crecimiento, expresada como fracción de la DBO_u o de la DQO (varía de 1/2 a 1/1)
- b' : coeficiente de respiración endógena por día, expresado como fracción de la DBO_u o de la DQO (9/100 a 10/100)
- X_t : sólidos volátiles suspendidos en el líquido mezclado (Kg / m^3)

En la mayor parte de los estanques donde únicamente se requiere un pretratamiento o una aireación suplementaria, la cantidad de sólidos biológicos puede mantenerse a un nivel bajo. En consecuencia, la ecuación 2.20 puede simplificarse como sigue:

$$O_d = a'' L_r \quad \text{Ec.2.21}$$

en donde:

O_d : Oxígeno requerido

a'' : coeficiente (varía de 0,7 a 1,4)

L_r : DBO reducida (kg / día)

2.4.2 LAGUNAS FACULTATIVAS

En la actualidad la mayor parte de los estanques de estabilización que tratan aguas residuales crudas son de tipo facultativo. Los sólidos sedimentables forman en el fondo del estanque una capa de fangos y de organismos animales y vegetales.

Los estanques facultativos se oxigenan principalmente por la actividad fotosintética de las algas bajo la influencia de la radiación solar, aunque en los estanques grandes la aireación superficial por la acción del viento también aporta una importante proporción de oxígeno.

El viento representa usualmente la principal fuente de energía en la mezcla del agua en los estanques facultativos.

La temperatura es de gran importancia porque afecta a la velocidad de degradación bioquímica. El anexo 2, muestra un diagrama de Lagunas Facultativas.

Hermann y Gloyna (1958) presentaron por primera vez una ecuación basada en la cinética de la degradación de los residuos en un estanque facultativo; la idea fué desarrollada posteriormente por Marais y Shaw (1961), Marais (1966) Huang y Gloyna (1968) y Gloyna-Espino (1969).

Si se supone que toda la DBO del agua que entra en el estanque queda estabilizada por organismos facultativos, que se produce una mezcla completa y que la descomposición tiene lugar de acuerdo con una reacción de primer orden, las concentraciones de la DBO en el efluente y en el agua afluyente pueden expresarse mediante la ecuación 2.22 :

$$L_p = \frac{L_o}{K_T R_T + 1} \quad \text{Ec.2.22}$$

en la que

L_p : DBO₅ del estanque y del agua efluente (Kg / m³)

L_o : DBO₅ del agua afluente (Kg / m³)

K_T : velocidad de descomposición a la temperatura T

R_T : tiempo de detención a la temperatura T

La velocidad de descomposición K_T depende de la temperatura en la siguiente forma:

$$\frac{K_{35}}{K_T} = \theta_2^{(35-T)} \quad \text{Ec.2.23}$$

en la que

T : temperatura a que funciona el estanque (°C)

θ_2 : coeficiente de reacción en función de la temperatura = 1.085

K_{35} : velocidad de descomposición a 35 °C

Para un porcentaje fijo de reducción de la DBO, la simetría de K_T y R_T en la ecuación 2.22 permite que la ecuación 2.23 pueda escribirse como sigue:

$$\frac{K_{35}}{K_T} = \theta_2^{(35-T)} = \frac{R_T}{R_{35}} \quad \text{Ec.2.24}$$

en la que:

R_{35} : tiempo de detención a 35 °C

Los datos obtenidos por Suwannakarn y Gloyna (1964) en una serie de estanques experimentales que trataban un agua residual sintética no sedimentable a diferentes temperaturas han sido analizados por Marais (1966), que obtuvo valores de 1.2 y 1.085 para K_{35} y θ , respectivamente (Gloyna, 1973).

A partir de la ecuación 2.22, la eficacia de la reducción de la DBO puede calcularse así, en función de diversos valores de R_T y K_T :

$$\text{Reduccion}(\%) = \frac{L_o - L_p}{L_o} 100 = \left(1 - \frac{1}{k_T R_T + 1} \right) 100 = \left(\frac{R_T}{\frac{1}{K_T} + R_T} \right) 100 \quad \text{Ec.2.25}$$

En la práctica, el proyecto de un estanque facultativo depende de una gran variedad de condiciones locales: sin embargo, a pesar de ello, se han establecido algunos procedimientos útiles y racionales para ese fin. Entre algunos procedimientos están: 1) carga por unidad de superficie; 2) el procedimiento empírico; 3) el sistema empleado en Sudáfrica, 4) Procedimiento CIPHERI de la India.

En el presente proyecto se desarrollará el procedimiento empírico, ya que después de estudiar los resultados de numerosos estanques experimentales, plantas piloto de mayor tamaño y más de 200 estanques en explotación; Gloyna y sus colaboradores han desarrollado varias fórmulas que permiten establecer una relación entre el volúmen del estanque y la temperatura, la DBO, el volúmen de agua residual y la toxicidad.

La intensidad de la luz, el viento, el total de sólidos disueltos y otros factores pueden influir en el volumen requerido de un estanque facultativo (Gloyna, 1973).

Con una temperatura elegida puede obtenerse el volumen requerido (en m³) de una laguna facultativa Ec. 2.26

$$V = (35) N q La \theta^{(35-T_m)} \quad \text{Ec.2.26}$$

En la que:

V : volúmen del estanque (m³)

N : número de personas que aportan agua residual

q : producción de agua residual por persona (m³ / día)

θ : coeficiente de temperatura de reacción = 1.085

T_m : temperatura media del agua en el mes más frío (°C)

La : Demanda Bioquímica de Oxígeno total

2.4.3 LAGUNAS ANAEROBIAS

El pretratamiento anaerobio de aguas residuales domésticas crudas afecta el comportamiento del estanque facultativo primario. Reduce la carga de DBO en el estanque facultativo y cambia la naturaleza de los sólidos sedimentables en la capa de fango. En un

estanque facultativo que esta precedido de otro anaerobio, el fango tiene un potencial de fermentación reducido, de modo que durante la estación cálida el fango que asciende es escaso.

La principal desventaja del tratamiento anaerobio es la posibilidad de que se produzcan olores que no se pueden evitar. En consecuencia la proximidad de zonas habitadas es un factor que debe tenerse siempre en cuenta en el proyecto de un estanque anaerobio.

La descomposición anaerobia en tanques sépticos, letrinas de pozo anegado y estanques anaerobios parece ser idéntica. La reducción de la DBO es función del tiempo de retención de los sólidos (es decir, en relación con el tiempo de detención de la fracción soluble).

El establecimiento de condiciones estables de fermentación en una unidad anaerobia puede exigir mucho tiempo, según la temperatura reinante.

Los resultados de experimentos realizados en tanques sépticos y letrinas en los Estados Unidos (1949) y en Zambia (1963) indican que en las regiones tropicales y subtropicales la reducción de la DBO puede calcularse así:

$$L_p = \frac{L_o}{K_n \left(\frac{L_p}{L_o} \right)^m R + 1} \quad \text{Ec. 2.27}$$

en la que:

R : tiempo de detención para un sistema de mezcla completa (días)

m : exponente, que se determinará por experimentación
(para Zambia, $m= 4.8$)

K_n : coeficiente de proyecto (6 a 22°C)

Van Eck (1965) observó que no se produce ninguna reducción de la DBO en los estanques anaerobios durante los meses de invierno con temperaturas inferiores a 10°C, mientras las reducciones llagan al 40-60% a 20°C y pasa del 80% a temperaturas de 25°C o mayores.

Otros parámetros de diseño se calculan igual que en las lagunas de Estabilización anteriores.

2.5 APLICACION DE ESTANQUES DE ESTABILIZACION

Los estanques de estabilización se aplican, solos o en varias combinaciones, al tratamiento de residuos industriales y domésticos. En la tabla 2.5 y 2.5.1, se muestran sus presentaciones típicas.

Los estanques aerobios se usan primordialmente para el tratamiento de residuos orgánicos solubles y efluentes de plantas de tratamiento de agua residual.

Los estanques aerobio-anaerobios son el tipo más utilizado y se aplican al tratamiento de agua residual así como a una amplia variedad de residuos industriales. Los estanques anaerobios se usan en serie con estanques aerobios-anaerobios que proporcionen un tratamiento completo .

Operación en serie o paralelo.

Los estanques de estabilización pueden emplearse dispuestos en serie o paralelo con fin de lograr los objetivos. El funcionamiento en serie resulta beneficioso en aquellos casos en que el alto nivel de DBO o eliminación de coliformes sea importante . El efluente de los estanques aerobio-anaerobios en serie tienen una concentración de algas mucho menor que la obtenida en paralelo , con una disminución resultante del calor y turbidez .

Se han diseñado muchas instalaciones de unidades múltiples y operadas en serie que aportan un tratamiento completo con una retención total de agua residual, evaporándose el líquido a la atmósfera o filtrándose por el suelo .

Las unidades en paralelo proporcionan una mejor distribución de los sólidos depositados (Metcalf, 1977).

TABLA 2.5 APLICACIÓN DE DIVERSOS TIPOS DE ESTANQUES DE ESTABILIZACIÓN Y SISTEMAS DE ESTANQUES (METCALF, 1977)

Tipo de Estanques o Sistema de Estanque	Aplicación
Aerobio (15-45 cm de profundidad)	Eliminación de nutrientes, tratamiento de residuos orgánicos solubles, producción de tejido celular de algas
Aerobio (hasta 1.5 m de profundidad)	Tratamiento de residuos orgánicos solubles y efluentes secundarios
Aerobio-anaerobio (fuente oxígeno : algas)	Tratamiento de aguas residuales crudas previamente sometidas a desbaste o sedimentación primaria y de aguas residuales industriales
Aerobio-anaerobio Con y sin recirculación de efluente (fuente oxígeno : aireadores de superficie)	Tratamiento de aguas residuales crudas previamente sometidas a desbaste o sedimentación primaria y de aguas residuales Industriales
Anaerobio	Tratamiento de residuos Industriales y domésticos
Anaerobio + aerobio-anaerobio con recirculación de aerobio-anaerobio a anaerobio	Tratamiento completo de residuos industriales y agua residual
Anaerobio + aerobio-anaerobio + sistema de estanque aerobio con recirculación de aerobio a anaerobio	Tratamiento completo de agua residual y residuos industriales con alta eliminación de bacterias

TABLA 2.5.1 CARACTERÍSTICAS DEL EFLUENTE DE DIVERSOS TIPOS DE ESTANQUES DE ESTABILIZACIÓN (METCALF, 1977)

Tipo de Estanques o Sistema de estanque	Sólidos suspendidos, mg / l			DBO ₅ mg / l	
	Micro- algas	organismos	otros	Soluble	Suspendida
Aerobio (15-45 cm de profundidad)	0.5-1.2	0.2-0.5	Bajo	0.02-0.1	0.3-1.0
Aerobio (hasta 1.15 m de profundidad)	0.4-1.0	0.2-0.5	Bajo	0.02-0.1	0.3-1.0
Aerobio-Anaerobio (fuente oxígeno : algas)	0.2-0.8	0.2-0.5	0.1-0.4	0.02-0.1	0.3-1.0
Aerobio-Anaerobio Con y sin recirculación de efluente (fuente de oxígeno: aireadores de superficie)	0.02-0.1	0.2-0.5	0.1-0.4	0.02-0.1	0.3-0.8
Anaerobio	-	0.1-0.3	0.3-0.5	0.05-0.2	0.3-0.8
Anaerobio + aerobio- anaerobio con recirculación de aerobio-anaerobio a anaerobio	-	0.2-0.5	0.05-0.15	0.05-0.1	0.3-0.8
Anaerobio + aerobio- anaerobio + sistema de estanque aerobio con recirculación de aerobio a anaerobio	0.05-0.1	0.02-0.05	0.03-0.1	0.02-0.1	0.3-1.0

Los sólidos suspendidos del efluente están compuestos de algas y otros microorganismos, que se estiman en función de la DBO_5 del afluente y una fracción de los sólidos suspendidos del afluente.

La DBO_5 efluente se compone de una fracción de la DBO_5 afluente soluble más una contribución de los sólidos suspendidos de efluente (Metcalf, 1977).

2.6 EJEMPLOS DE CALCULO DE LAGUNAS DE ESTABILIZACION

LAGUNA AEROBIA

Diseñese un estanque de estabilización aerobia para tratar un caudal residual de 3785 m^3 / día con una DBO_L soluble de 100 mg / l. Supongase que son de aplicación las siguientes condiciones :

1. Los sólidos suspendidos del afluente son despreciables
2. Eliminación (conversión) de $DBO_L = 90\%$
3. Constante de la eliminación de la DBO_L soluble de primer orden = 0.25 a 20°C y 40 °N
4. Coeficiente de temperatura $\theta = 1.06$ a 20°C
5. Latitud 40°N
6. Temperatura del estanque en verano= 32°C
7. Temperatura del estanque en invierno =10°C
8. Area máxima de un estanque individual=4ha
9. Profundidad máxima del estanque= 0.9m
10. Factor de dispersión del estanque = 1

Solución:

A partir de ecuación de WEHNER y WILHELM , (Ec. 2.18) se encontrará el tiempo de detención de la laguna aerobia, para luego encontrar otros parámetros de diseño.

Se calculará primeramente la velocidad de reacción a T °C (Ec. 2.17) para las dos condiciones :

Para condiciones invernales :

$$K_T = K_{20} \theta^{(T-20)}$$

Sustituyendo datos

$$K_{10} = 0.25 (1.06)^{10-20} = 0.14$$

Lo mismo se hará para las condiciones de verano

$$K_{32} = 0.25(1.06)^{32-20} = 0.5$$

Sustituyendo la velocidad de reacción K_T en $a = \sqrt{1 + 4k_r tp}$; queda como incógnita el tiempo, que será un dato que se irá proporcionando en la (Ec.2.18), hasta lograr la igualación del porcentaje de sustrato remanente (L_t).

$$L_t = \frac{4ae^{(1/2p)}}{(1+a)^2 e^{(a/2p)} - (1-a)^2 e^{(-a/2p)}}$$

Para condiciones invernales el tiempo de detención (t) es 35.7 días y para condiciones de verano el tiempo es de 10 días.

Para calcular el área superficial se utilizará la ecuación 2.10 :

$$S = \text{volumen} / \text{profundidad}$$

donde

$$\text{Volumen} = (\text{Caudal}) (\text{tiempo})$$

Sustituyendo el volumen en la Ec.2.10 ,se transforma en:

$$S = ((\text{Caudal}) (\text{tiempo})) / \text{profundidad}$$

Sustituyendo datos para condiciones invernales :

$$S = ((3785\text{m}^3 / \text{día})(35.7 \text{ día})) / 0.9\text{m}$$

$$S = 150,138 \text{ m}^2 \quad (15 \text{ ha})$$

Por lo tanto se utilizarán cuatro estanques ó lagunas de 3.75 ha.

Para condiciones de verano :

$$S = ((3785\text{m}^3 / \text{día})(10 \text{ días})) / 0.9$$

$$S = 42,055.556 \text{ m}^2 \quad (4.2 \text{ ha})$$

Solamente se utilizará una laguna.

LAGUNA AIREADA

Para proyectar una unidad aireada de pretratamiento los datos disponibles son:

Caudal (Q) : 7570 m³ / día

DBO₅ (Lo) : 0.250 Kg / m³

DBO_u (La) : 0.305 Kg / m³

Nivel de oxígeno disuelto en el estanque (C) : 0.0015 Kg / m³

Nivel de saturación de O₂ en agua destilada a 20°C (Cs) : 0.0092 Kg / m³

Nivel de saturación de O₂ en el estanque a temperatura T (C_{sw}) : 10.2 Kg / m³

Temperatura (T) : 15 ° C

α' : 0.7

Rendimiento del aireador dado por el fabricante : 1.6 Kg de O₂ / Cvh a 20° C

α : 0.90

Velocidad de reacción (K_T) : 0.35 por día

Calcular el tamaño y el número de aireadores mecánicos necesarios para reducir la DBO del agua residual afluyente en un 50%.

Solución :

a) Carga orgánica (CO) , usando la Ec.2.7:

$$CO : DBO_5 \cdot Q$$

$$CO : 0.25 \frac{Kg}{m^3} \cdot 7570 \frac{m^3}{dia}$$

$$CO : 1892.5 \frac{Kg}{dia}$$

b) DBO a reducir (L_r) :

$$L_r = CO \cdot \%$$

$$L_r = 1892.5 \cdot 0.5$$

$$L_r = 946.2 \frac{Kg}{dia}$$

c) El tiempo de detención (R_T), se encuentra con la Ec.2.22:

$$R_T = \frac{L_o - L_p}{K_T L_p}$$

$$R_T = \frac{0.25 - 0.125}{0.35 \times 0.125}$$

$$R_T = 2.86 \text{ días}$$

d) Volumen (V), se encuentra con la Ec.2.8 :

$$V = R_T \cdot Q$$

$$V = 2.86 \cdot 7570$$

$$V = 2,628.53$$

e) Superficie (S), se encuentra con la Ec.2.9

$$S = \frac{V}{d}$$

$$S = \frac{21628}{3}$$

$$S = 7,209.52 \text{ m}^2$$

f) Aireación suplementaria (O_d), con la Ec. 2.21:

$$O_d = a'' L_r$$

$$O_d = 0.7 \cdot 946.25$$

$$O_d = 662.375 \frac{\text{Kg}(O_2)}{\text{dia}} = 27.7 \frac{\text{Kg}(O_2)}{\text{hora}}$$

g) Aireación mecánica (O_s), con la Ec.2.19:

$$O_s = Om \frac{C_{sw} - C}{C_s} * \alpha * \theta^{(T-20)} Fa$$

$$O_s = 1.6 \frac{0.0102 - 0.0015}{0.0092} * 0.9 * 1.02^{(15-20)} * 1$$

$$O_s = 1.23 \text{ Kg de } \frac{O_2}{\text{CVh}}$$

h) Necesidades de energía (E):

$$E = \frac{O_d}{O_s}$$

$$E = \frac{27.7}{1.23}$$

$$E = 22.5 \text{ CV}$$

Por consiguiente, se requiere una unidad de aireación de 25 CV (1 CV = 0.736 kw)

LAGUNA FACULTATIVA

Calcúlese los siguientes datos: a) carga orgánica ; b) volumen ; c) superficie; d) tiempo de detención ; e) carga de superficie

Datos disponibles:

Localidad donde las aguas residuales son principalmente de origen doméstico.

DBO_5 del agua residual a $20^\circ\text{C} = 0.250 \text{ Kg} / \text{m}^3$

DBO_u del agua residual (L_a) = $0.300 \text{ Kg} / \text{m}^3$

Habitantes que producen residuos (N) = 20,000

Volumen del agua residual por habitante (q) = $0.15 \text{ m}^3 / \text{día}$

Caudal (Nq) = $3,000 \text{ m}^3 / \text{día}$

Temperatura media del agua en el mes más frío (T_m) = 10°C

Coefficiente de temperatura (θ) = 1.085

Pérdidas por filtración = insignificante

Evaporación = lluvia caída

Sulfatos = menos de $0.500 \text{ Kg} / \text{m}^3$

Toxicidad industrial = despreciable

Profundidad = 1.75 m

Solución:

a) Carga orgánica a partir de la ecuación 2.7

$$\text{CO} = \text{DBO} \text{ (a ser removida en la laguna). caudal}$$

Sustituyendo datos :

$$\text{CO} = (0.3 \text{ Kg} / \text{m}^3) (3000 \text{ m}^3 / \text{día})$$

$$\text{CO} = 900 \text{ Kg DBO}_u / \text{día}$$

b) Volumen a partir de ecuación 2.26

$$V = (35) Nq L_a \theta^{(35-T_m)}$$

Sustituyendo datos:

$$V = (35)(3000 \text{ m}^3 / \text{día})(0.3 \text{ Kg} / \text{m}^3)(1.085^{35-10})$$

$$V = 2.4 \times 10^5 \text{ m}^3$$

c) Superficie , se encuentra a partir de la ecuación 2.10

$$S = \text{volumen} / \text{profundidad}$$

Sustituyendo datos:

$$S = (2.4 \times 10^5) / 1.75$$

$$S = 1.4 \times 10^5 \text{ m}^2$$

d) Tiempo de detención se encuentra a partir de la ecuación 2.9

$$t = \text{volumen} / \text{caudal}$$

Sustituyendo datos:

$$t = 2.4 \times 10^5 / 3000$$

$$t = 80 \text{ días}$$

e) Carga superficial , se usa la ecuación 2.16

$$CS = CO / S.$$

Sustituyendo datos:

$$CS = 900 / 14$$

$$CS = 65 \text{ Kg DBO}_u / \text{ha}$$

LAGUNA ANAEROBIA

Los datos para el proceso de cálculo de un proyecto de estanque anaerobio de zona subtropical son :

Población=8000 habitantes

DBO₅ del agua afluente (L_0) = 0.400 Kg / m³

DBO₅ del agua efluente (L_p) = 0.240 kg / m³

$K_n = 6.0$

$m = 4.8$

Caudal de agua residual = 0.07 m³ por habitante / día

Profundidad= 2 m

Calcular: a) el tiempo de detención, b) Volumen del estanque, c) la superficie del estanque

Solución :

a) El tiempo de detención , R se basa en la ecuación 2.27

$$L_p = \frac{L_0}{K_n \left(\frac{L_p}{L_0} \right)^m R + 1}$$

Despejando R se tiene :

$$R = \left(\frac{L_0}{L_p} - 1 \right) \cdot \left[\frac{1}{K_n \left(\frac{L_p}{L_0} \right)^m} \right]$$

Sustituyendo datos:

$$R = \left(\frac{0.400}{0.240} - 1 \right) \cdot \left[\frac{1}{6 * \left(\frac{0.240}{0.400} \right)^{4.8}} \right]$$

$$R = 1.3 \text{ días}$$

b) El volumen se calculará mediante la ecuación 2.8

$$V = \text{días} \cdot \text{caudal}$$

Sustituyendo datos:

$$\text{caudal (Q)} = N \cdot q$$

$$V = 1.3 (8000) 0.07$$

$$V = 722.4 \text{ m}^3$$

c) La superficie se calculará mediante la ecuación 2.10

$$S = \text{volumen} / \text{profundidad}$$

Sustituyendo datos:

$$S = 722.4 / 2$$

$$S = 361 \text{ m}^2$$

2.7 PLANTEAMIENTO DE LO QUE APORTARA LA CODIFICACION DEL CASO

Según la naturaleza de actividad biológica, los estanques ó lagunas de estabilización se han clasificado en aerobias, aerobias-anaerobias (facultativas) y anaerobias, como ya se ha mencionado.

Antes de proceder al diseño de un sistema de lagunas de estabilización, se tomarán criterios de diseño que servirán de parámetros para las diferentes lagunas. (Tabla 2.1, Rivas Mijares, 1978).

LAGUNAS AEROBIAS

En el diseño de lagunas aerobias los procesos que se considerarán son:

- a) Producción de Oxígeno
- b) Producción de células de algas
- c) Penetración de oxígeno
- d) Parámetros de diseño:
 - carga orgánica
 - volumen
 - período de detención
 - superficie
 - carga hidráulica
 - caudal
 - carga superficial
 - velocidad de reacción
- e) Aireación mecánica

LAGUNAS AEROBIAS-ANAEROBIAS

La mayoría de estanques de estabilización son de tipo facultativo; entre los parámetros de diseño a considerar están :

- a) DBO₅ del agua efluente
- b) Velocidad de descomposición a 35°C
- c) Volumen
- d) Superficie
- e) Tiempo de detención
- f) Carga superficial

LAGUNAS ANAEROBIAS

Son usadas para el tratamiento de desechos domésticos con altas concentraciones y desechos industriales. Este tipo de unidades soporta altas cargas volumétricas.

Los parámetros de diseño que se considerarán son:

- a) DBO₅ del agua afluente
- b) Volumen
- c) Superficie
- d) Tiempo de detención
- e) Carga superficial

2.7.1 DEFINICION DE VARIABLES

- A = producción de células de algas (Kg-algas / ha-día)
- $a = \text{constante que se calcula por } \sqrt{1+4ktp}$
- $a' = \text{DBO reducida y utilizada para proporcionar energía de crecimiento, expresada como fracción de la DBU}_u \text{ o DQO (1/2 a 1/1)}$
- $a'' = \text{coeficiente (0.7 a 1.4)}$
- $b' = \text{coeficiente de respiración endógena por día expresada como fracción de la DBO}_u \text{ ó DQO (9/100 a 10/100)}$
- C = nivel de oxígeno disuelto en el estanque(Kg / m³)
- CH = carga hidráulica (m/día)
- CO = carga orgánica(Kg DBO /día)
- Cs = nivel de saturación de O₂ en agua destilada a 20°C (Kg / m³)
- Csw = nivel de saturación de O₂ en el estanque a temperatura T.(Kg / m³)
- CS = carga superficial (kg /día)
- d = profundidad (m)
- D = coeficiente de dispersión axial (m²/ día)
- E = necesidades de energía (CV)
- f = factor adimensional

- F = factor de oxigenación ó eficacia de utilización de la energía luminosa (%)
 F_a = factor de corrección para altitudes superiores a 1200 m
 I_l = radiación solar (cal/ cm-día)
 K = constante de reacción, \log_{10} (descomposición / día)
 K_n = coeficiente de proyecto (6 a 22°C)
 K_T = velocidad de reacción a T °C
 K_1 = constante de reacción, \log_e (descomposición / día)
 K_{20} = velocidad de reacción de 20°C
 K_{35} = velocidad de descomposición a 35 °C
 l = longitud característica (m)
 L = longitud de la laguna (m)
 L_a = demanda bioquímica de O₂ total, DBO_u(en el agua afluente) (Kg / m³)
 L_o = DBO₅ del agua afluente (Kg /m³)
 L_p = DBO₅ del estanque y del agua efluente (Kg / m³)
 L_r = demanda bioquímica de oxígeno reducida.(Kg / día)
 L_s = demanda bioquímica de oxígeno, satisfecha al cabo de t días (Kg / m³l)
 L_t = demanda bioquímica de oxígeno remanente al cabo de t días (Kg / m³)
 m = exponente determinado por experimentación, en Zambia para regiones tropicales y subtropicales (4.8)
 n = orden de reacción
 N = número de personas que aportan agua
 O = producción de oxígeno (Kg-O₂/ ha-día)
 O_d = oxígeno requerido como aireación suplementaria (kg / h)
 O_n = oxígeno requerido en un sistema aerobio con mezcla completa (Kg / m³l)
 O_m = rendimiento del aireador dado por el fabricante
 O_s = oxígeno suministrado por aireación mecánica (Kg de o₂/ Cvh)
 p = factor de dispersión axial
 Q = caudal (m³ / día)
 q = producción de agua residual por persona (m³ / día)
 R = tiempo de detención para un sistema de mezcla completa (días)

- R_T = tiempo de detención a la temperatura T (días)
 R_{35} = tiempo de detención a 35°C
 S = superficie de la laguna (m^2)
 S_e = concentración del substrato en el efluente
 S_o = concentración del substrato en el afluente
 T = temperatura en °C
 T_m = temperatura media del agua en el mes más frío(°C)
 t = tiempo ó período de detención (días)
 u = velocidad del fluido (m / h)
 V = volumen del estanque (m^3)
 W = ancho de la laguna (m)
 x_t = sólidos volátiles suspendidos en el líquido mezclado (Kg / m^3)
 Z = profundidad de la laguna (m)
 α = coeficiente que relaciona la transferencia total del agua residual con la transferencia total del agua de grifo
 θ = coeficiente de actividad -temperatura
 = coeficiente de reacción de la temperatura (aireación = 1.02)
 = coeficiente de reacción en función de la temperatura para lagunas facultativas (1.085)
 $\%$ = porcentaje a reducir la DBO

2.7.2 SOLUCION DE MODELOS

Muchos investigadores han tratado de correlacionar los diferentes factores que afectan el comportamiento de las lagunas de estabilización con el propósito de llegar a obtener fórmulas matemáticas en función de esos factores o parámetros que permitan hacer un diseño.

A continuación se presentan algunos criterios de diseño para las lagunas ya mencionadas anteriormente.

2.7.3 LAGUNAS AEROBIAS

Producción de Oxígeno en Lagunas de Estabilización

Las lagunas de estabilización aerobias dependen de las algas para obtener el oxígeno necesario que satisfaga la demanda que representa la DBO (Ramalho, 1991).

La cantidad de oxígeno que se produce por las algas puede estimarse por la Ec. 2.12:

$$O = 0.25 F I_l$$

Ahora, para evaluar el factor de oxigenación ó % de energía luminosa (F) de la Ec.2.12, se realiza a través de un método empírico en función de la DBO, Ec.2.13:

$$F = 1.3 (\ln CO - 3.5)$$

Para calcular la carga orgánica (CO) de la Ec.2.13 se hace a partir de la Ec.2.7:

$$CO = DBO \text{ (a ser removida) } \cdot \text{Caudal}$$

La DBO será un dato conocido y el caudal (Q) se encuentra a partir de la Ec.2.15:

$$Q = N \cdot q$$

N y q son datos que se conocerán para cada caso, el número de personas que aportan agua y la producción de agua residual por persona.

Debido a que este oxígeno se produce mediante fotosíntesis se necesita también de la luz solar (I_l). El valor de la luz solar se obtendrá a partir de la tabla 2.3 como un promedio mensual para un determinado lugar en el país.

Como datos principales se pueden dar la elevación, latitud norte, latitud oeste y buscar la estación metereológica que se encuentra a determinadas condiciones según la tabla 2.2. Una vez ubicada la estación metereológica a partir de su abreviación, se encontrará el promedio mensual de radiación en $\text{cal/cm}^2\text{-día}$ en la tabla 2.3.

Los datos de las tablas 2.2 y 2.3 estarán incluidos en el programa.

Producción de Células de Algas

La producción de algas tiene relación con el factor de conversión de la luz (F) por lo tanto se calcula a partir de la Ec.2.14:

$$A = 0.15 F I_l$$

La radiación solar (I_l) y el factor de oxigenación (F) se calcula de igual forma que en la producción de oxígeno.

El Diagrama 2.A, presenta un flujograma para la producción de oxígeno y de células de algas.

Penetración de Oxígeno

La penetración de oxígeno en lagunas de estabilización se puede calcular dependiendo de la carga y de las profundidades para condiciones aerobias. También, para determinar si habrá condiciones aerobias en los estanques facultativos.

La ecuación que correlaciona la penetración de oxígeno con la carga DBO es la siguiente:

$$d = a + b(\text{DBO}) + c(\text{DBO})^2 + e/(\text{DBO}) \quad \text{Ec.2.28}$$

Esta ecuación (Ec.2.28), se obtuvo de la correlación de datos de la figura (Fig.2.3), utilizando el programa Tbl Curve. Los datos de los coeficientes obtenidos fueron:

$$a = 0.90132612, \quad b = -0.0095455207, \quad c = 2.2472713 \times 10^{-5}, \quad e = 26.485764$$

Parámetros de Diseño

Las lagunas de estabilización de aguas residuales son factibles cuando se dispone de grandes superficies de terreno a bajo costo y cuando la calidad exigida del efluente no es muy estricto.

Los modelos utilizados para los cálculos de diseño de lagunas aerobias son:

A través de la Ec.2.8 se calcula el Volumen:

$$V = t \cdot N \cdot q$$

Al multiplicar $N \cdot q$ nos resulta el caudal (Q) que es la ecuación (Ec.2.15). La variable t es el tiempo de detención y este es un dato proporcionado en el proyecto, para lagunas aeradas. En caso se proporcionaran las otras variables de la Ec. 2.8, entonces se podrá despejar t de la siguiente manera:

$$t = \frac{V}{N \cdot q} \quad \text{Ec.2.29}$$

Superficie:

La superficie se evaluará dividiendo el volumen entre la profundidad de la laguna, que para lagunas aerobias la profundidad suele ser entre 30 a 60 cms. según la tabla 2.1 (Rivas Mijares, 1978).

Por lo tanto la ecuación para la superficie es la 2.10:

$$S = \frac{V}{d}$$

Carga Hidráulica:

Para encontrar la carga hidráulica se hará la división entre la profundidad y el período de detención; así como en la ecuación 2.11:

$$CH = \frac{d}{t}$$

La Carga Superficial se encontrará como en la ecuación 2.16 :

$$CS = \frac{CO}{S}$$

La carga orgánica CO y S se analizaron anteriormente.

Para una muestra residual, la temperatura suele ser de 20°C y para determinar la velocidad de reacción a una temperatura T se hace a través de la ecuación 2.17:

$$K_T = K_{20} \theta^{(T-20)}$$

El coeficiente de actividad- temperatura (θ) puede ser elegido según el proceso biológico, Tabla 2.4 (Metcalf, 1977); para lagunas de aireación se tomará 1.06. Un valor conocido será el de la velocidad de reacción a 20°C y T es la temperatura a que estará la laguna dependiendo de sus condiciones estacionarias., (invierno-verano).

Con base a resultados experimentales, se sabe hoy en día que las lagunas de estabilización son de flujo disperso, y que el grado de dispersión depende de la geometría de las lagunas.

Siendo el paralelepípedo rectángulo la figura geométrica más simple, se han llegado a producir modelos matemáticos que facilitan el dimensionamiento de la laguna en función de longitud (L), ancho (W) y profundidad (Z).

Por lo tanto si el volumen del paralelepípedo es :

$$V = L \cdot W \cdot Z \quad \text{Ec.2.30}$$

el tiempo de detención será:

$$t = V / Q = (L \cdot W \cdot Z) / Q \quad \text{Ec. 2.31}$$

La relación del sustrato del efluente sobre el sustrato del afluente (S_e / S_o) es el porcentaje de sustrato remanente (L_t); y se calculará a partir del modelo de flujo disperso, cuya aplicación ha sido propuesta por muchos investigadores para la reducción bacteriana. La ecuación para este cálculo es la Ec. 2.18:

$$L_t = \frac{4ae^{(1/2p)}}{(1+a)^2 e^{(a/2p)} - (1-a)^2 e^{(-a/2p)}}$$

donde $a = (1 + 4ktp)^{1/2}$; el valor de la constante de velocidad de reacción (k) se calculará con la Ec. 2.17. Si se conoce el porcentaje a eliminar de DBO, se obtendrá el remanente (L_t). Una forma de encontrar el factor de dispersión adimensional (p):

$$p = \frac{1.158 [t(W + 2Z)]^{0.489} (W)^{1.511}}{(T + 42.5)^{0.734} (LZ)^{1.489}} \quad \text{Ec.2.32}$$

L, W y Z son las dimensiones de la laguna y t es el período de detención.

El factor de dispersión también puede ser determinado según la forma de la laguna (Yanez Cossío, 1993) por medio de la tabla siguiente:

TABLA 2.6 FACTOR DE DISPERSIÓN PARA UNA LAGUNA DE ESTABILIZACIÓN (YANEZ COSSIO, 1993)

Relación Largo / ancho	Factor de dispersión
1	1.0
2	0.5
4	0.25
8	0.12

El modelo de la Ec. (2.18), es utilizado para estimar el período de detención, por lo que se realizará una prueba y error para encontrar t. Se sustituye el valor de k obtenido de la Ec.(2.17).

Los Diagramas 2.B y 2.C, muestran los diagramas de flujo de los procesos para las lagunas aerobias.

Aireación Mecánica

La cantidad de aireación mecánica esta dada por la Ec. 2.19:

$$O_s = Om \frac{C_{sw} - C}{C_s} \alpha \cdot \theta_1^{(T-20)} F a$$

Las unidades de aireación mecánica de superficie pueden suministrar 1-1.8 Kg de oxígeno/CVh.

El rendimiento del aireador dado por el fabricante(Om) usualmente es de 1.8- 2.1 Kg de oxígeno/CVh.

Las variables C_{sw} , C , C_s serán datos proporcionados; así como las características de transferencia de oxígeno entre el agua residual y el agua limpia (α); típicamente, los valores oscilan entre 0.6 y 1.1 (Gloyna, 1973).

El coeficiente de temperatura (θ^{-1}) será 1.02 y el factor de corrección se asumirá 1.

Para una aireación suplementaria y para mantener una cantidad de sólidos a un nivel bajo, las necesidades de oxígeno se obtendrán por la ecuación 2.21:

$$O_d = a'' L_r$$

El coeficiente a'' varía de 0.7 a 1.4 y la DBO reducida (L_r) se encontrará multiplicando la CO por el porcentaje de DBO a que se quiere reducir (%).

Las necesidades de energía (E) se encontrarán así:

$$E = \frac{O_d}{O_s} \quad \text{Ec. 2.33}$$

A partir de E se podrá conocer el número de aireadores necesarios, dependiendo de los CV resultantes.

El Diagrama 2.D, es el diagrama de flujo para la aireación.

2.7.4 LAGÚNAS AEROBIAS-ANAEROBIAS

Basada en la degradación de los residuos en un estanque facultativo, las concentraciones de la DBO en el agua efluente y en el agua afluente pueden expresarse por la ecuación 2.22:

$$L_p = \frac{L_o}{K_T R_T + 1}$$

L_p y L_o son la DBO₅ del agua efluente y afluente (kg / m^3) respectivamente. El tiempo de retención R_T se puede calcular despejando de la ecuación 2.22 ó puede ser un dato proporcionado.

La velocidad de descomposición K_T depende de la temperatura en la siguiente forma :

$$\frac{K_{35}}{K_T} = \theta_2^{(35-T)} \quad \text{Ec. 2.23}$$

El coeficiente de temperatura θ_2 será 1.085, la temperatura (T) a que funciona el estanque será una variable proporcionada para encontrar la velocidad de reacción a la temperatura T y K_{35} será 1.2.

K_T y R_T se relacionan de la siguiente forma, Ec.2.24:

$$\frac{K_{35}}{K_T} = \theta_2^{(35-T)} = \frac{R_T}{R_{35}}$$

Volumen del Estanque :

El volumen de una laguna facultativa se puede calcular según la ecuación 2.26 :

$$V = (35) Nq L a . \theta_2^{(35-T_m)}$$

Los valores de N, q, L, a, θ_2 y T_m serán proporcionados ó si se tiene alguno de ellos se despejará de la ecuación anterior.

Superficie

Si se ha obtenido el volumen y teniendo la profundidad de la laguna facultativa, se conocerá la superficie según la ecuación 2.10:

$$S = \frac{V}{d}$$

Otros parámetros como tiempo de detención, carga superficial, se calcularán igual que en la sección 2.9.1 (Lagunas Aerobias).

Los Diagramas 2.E y 2.F, son los diagramas de flujo para los procesos en estas lagunas.

2.9.3 LAGUNAS ANAEROBIAS

Tomando como base el procedimiento africano, debido a que los resultados experimentales para la reducción de DBO se asemejan al de las regiones tropicales y subtropicales, se calculará por medio de la ecuación 2.27:

$$Lp = \frac{Lo}{K_n \left(\frac{L_p}{L_o}\right)^m R + 1}$$

Asumiendo que la temperatura del agua de entrada y del estanque es 20°C y que m es 4.8 por experimentación y K_n es un coeficiente de proyecto de 6, entonces a partir de ésta ecuación 2.27 se despejará R que es el tiempo de detención. Despejando, queda la ecuación 2.34:

$$R = \left(\frac{L_o}{L_p} - 1\right) \left[\frac{1}{K_n \left(\frac{L_p}{L_o}\right)^m} \right] \quad \text{Ec. 2.34}$$

Con el tiempo de detención R para lagunas anaerobias, se puede encontrar los demás parámetros de diseño de igual forma que en las lagunas aerobias.

El Diagrama 2.G, representa el flujograma de procesos .

**DIAGRAMAS DE FLUJO
PARA
LAGUNAS DE ESTABILIZACION**

- 2.A DETERMINACION DE PRODUCCION DE OXIGENO Y CELULAS DE ALGAS EN LAGUNAS AEROBIAS
- 2.B DETERMINACION DEL PORCENTAJE DE SUSTRATO REMANENTE EN LAGUNAS AEROBIAS
- 2.C PARAMETROS DE DISEÑO PARA LAGUNAS AEROBIAS
- 2.D DETERMINACION DE AIREACION
- 2.E DETERMINACION DE LA DBO DEL AGUA EN UNA LAGUNA FACULTATIVA
- 2.F PARAMETROS DE DISEÑO PARA LAGUNAS FACULTATIVAS
- 2.G PARAMETROS DE DISEÑO PARA LAGUNAS ANAEROBIAS

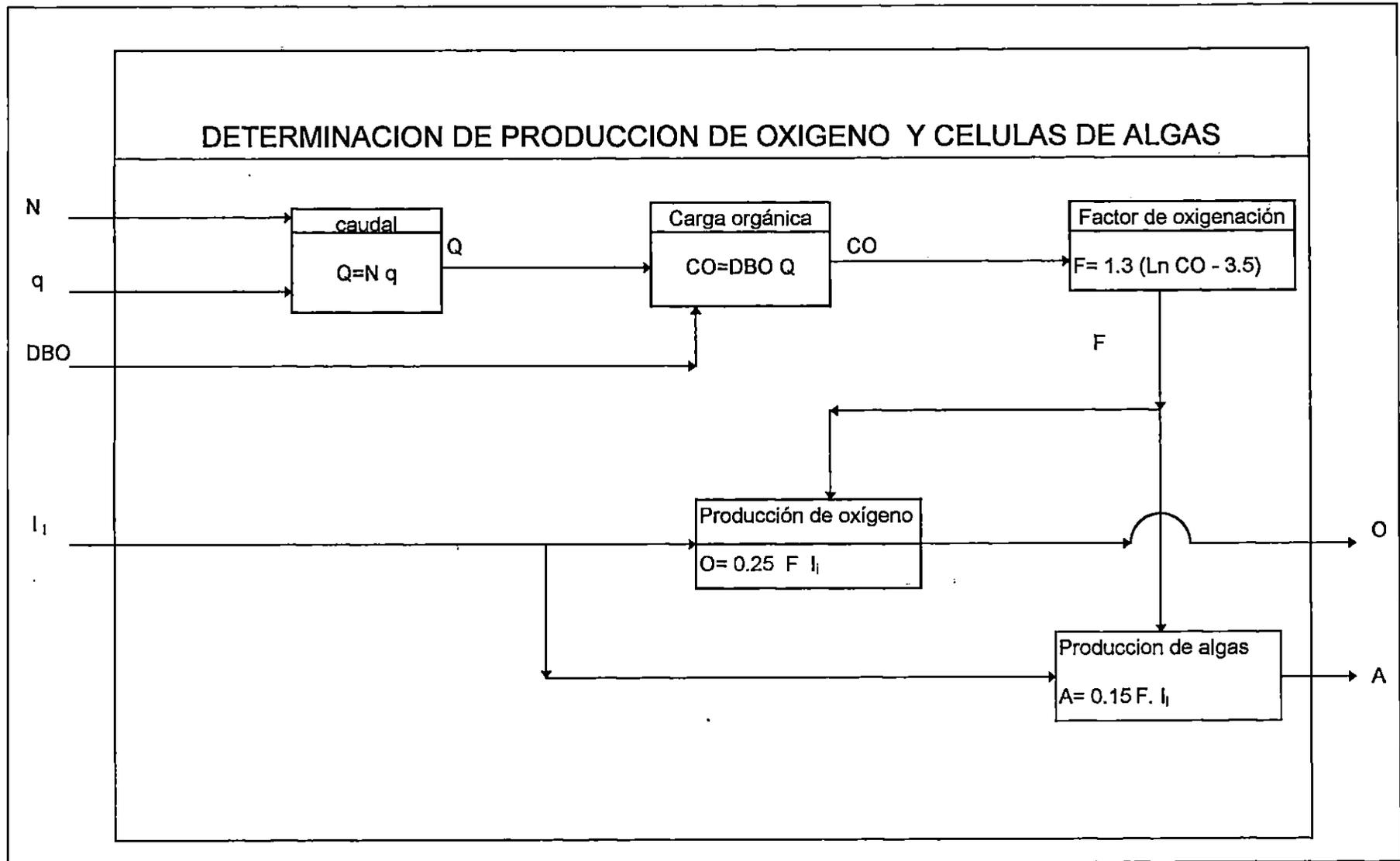


DIAGRAMA 2.A DETERMINACION DE PRODUCCION DE OXIGENO Y CELULAS DE ALGAS

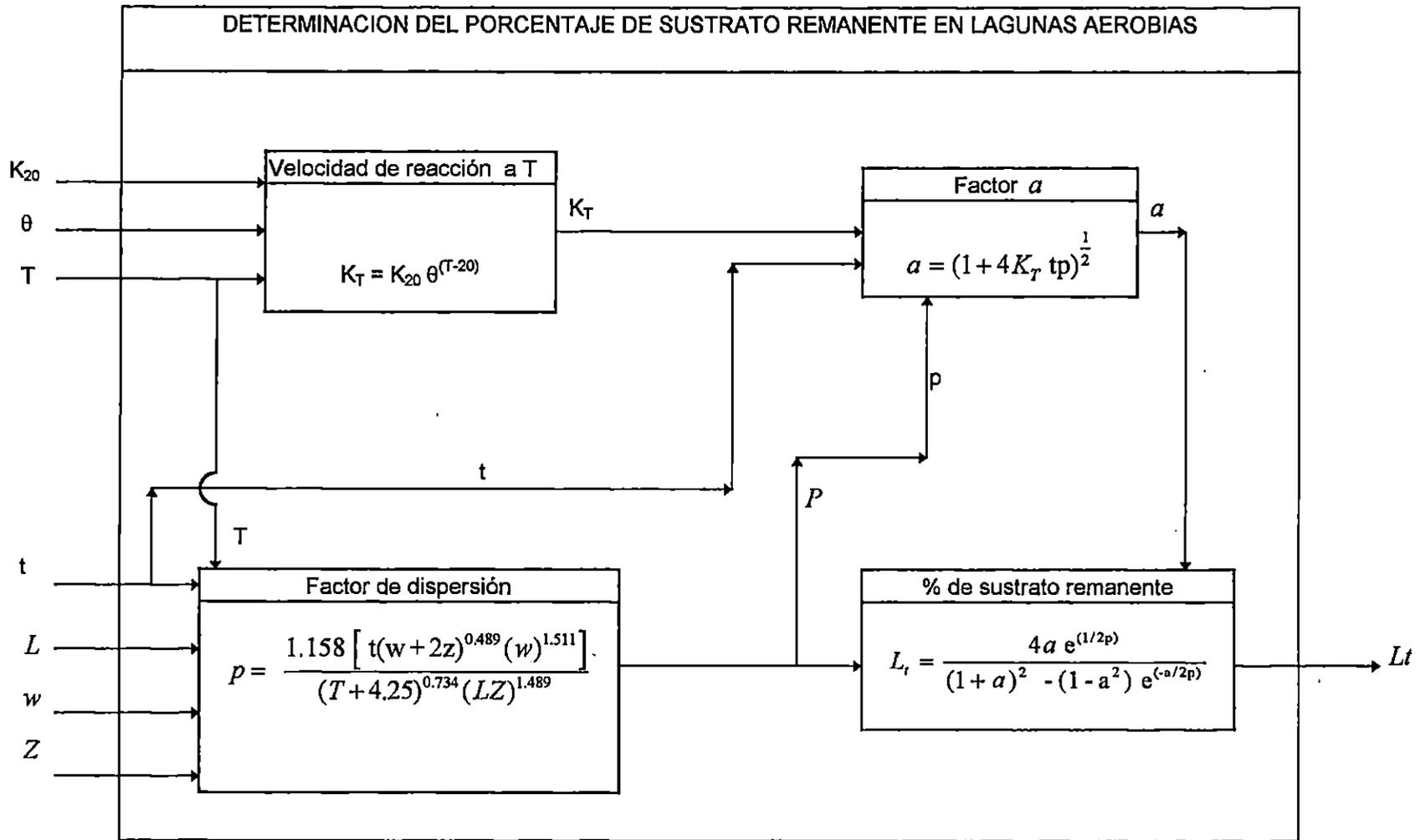


DIAGRAMA 2.B DETERMINACION DEL PORCENTAJE DE SUSTRATO REMANENTE

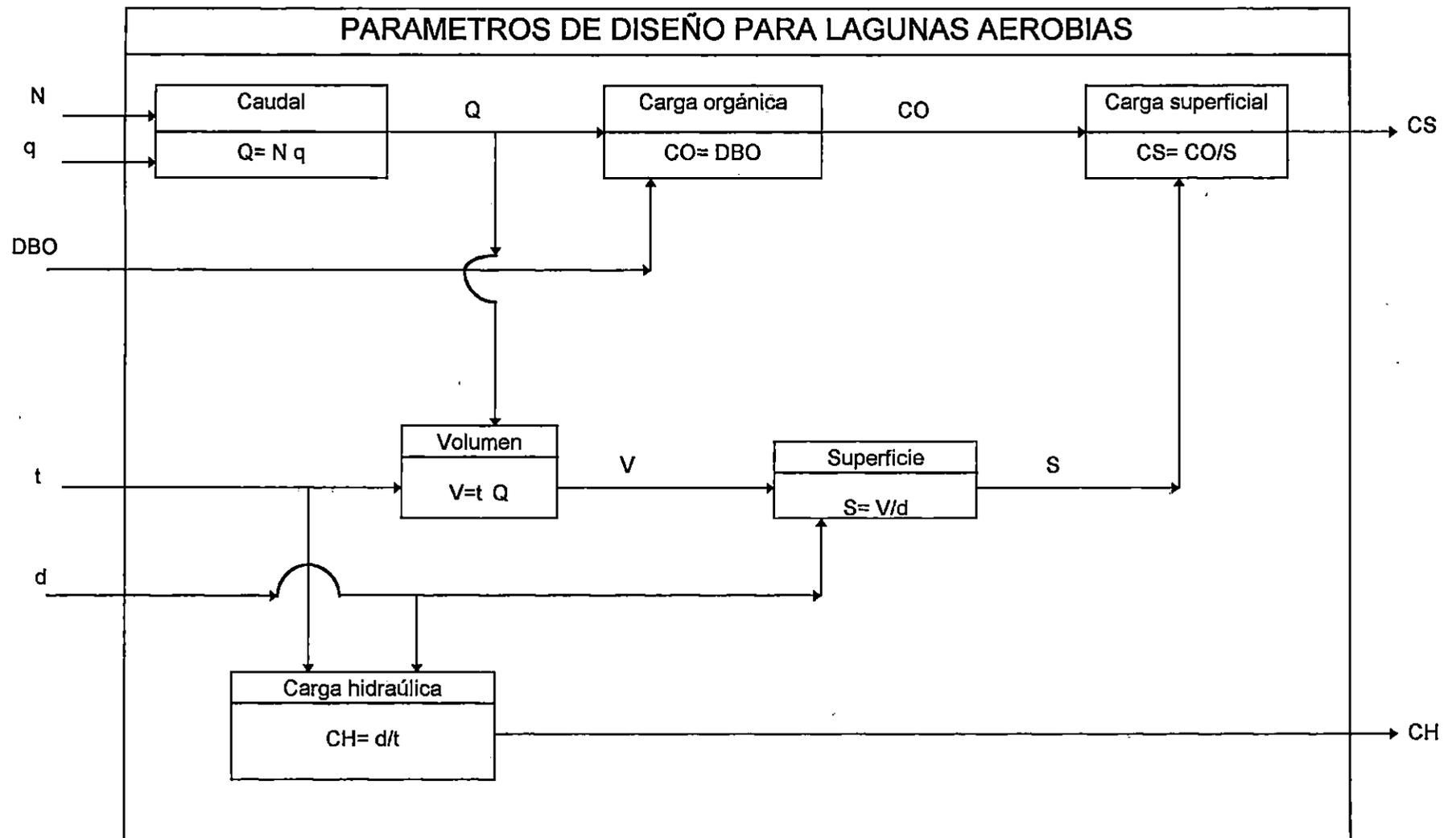


DIAGRAMA 2.C PARAMETROS DE DISEÑO PARA LAGUNAS AEROBIAS

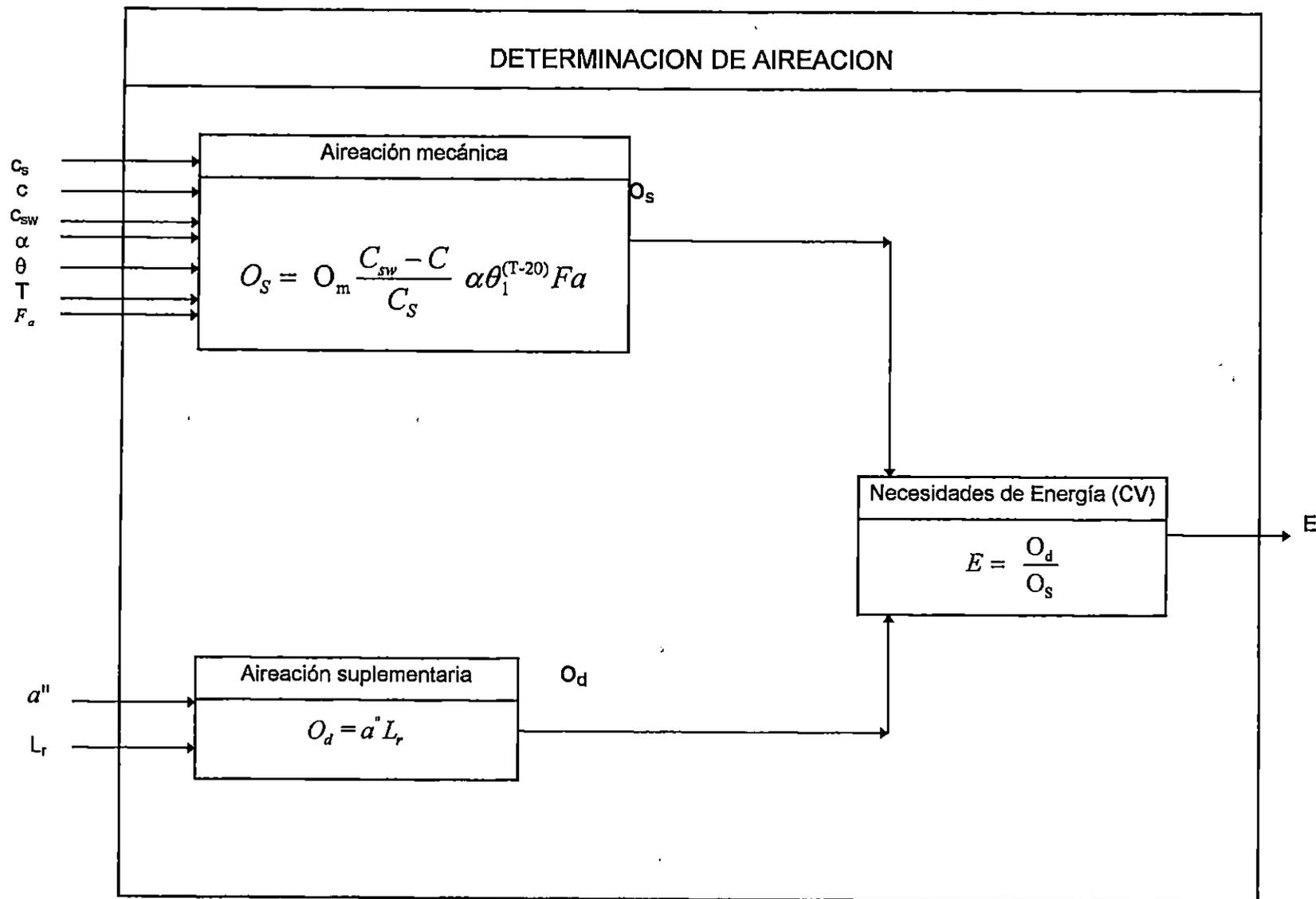


DIAGRAMA 2.D DETERMINACION DE AIREACION

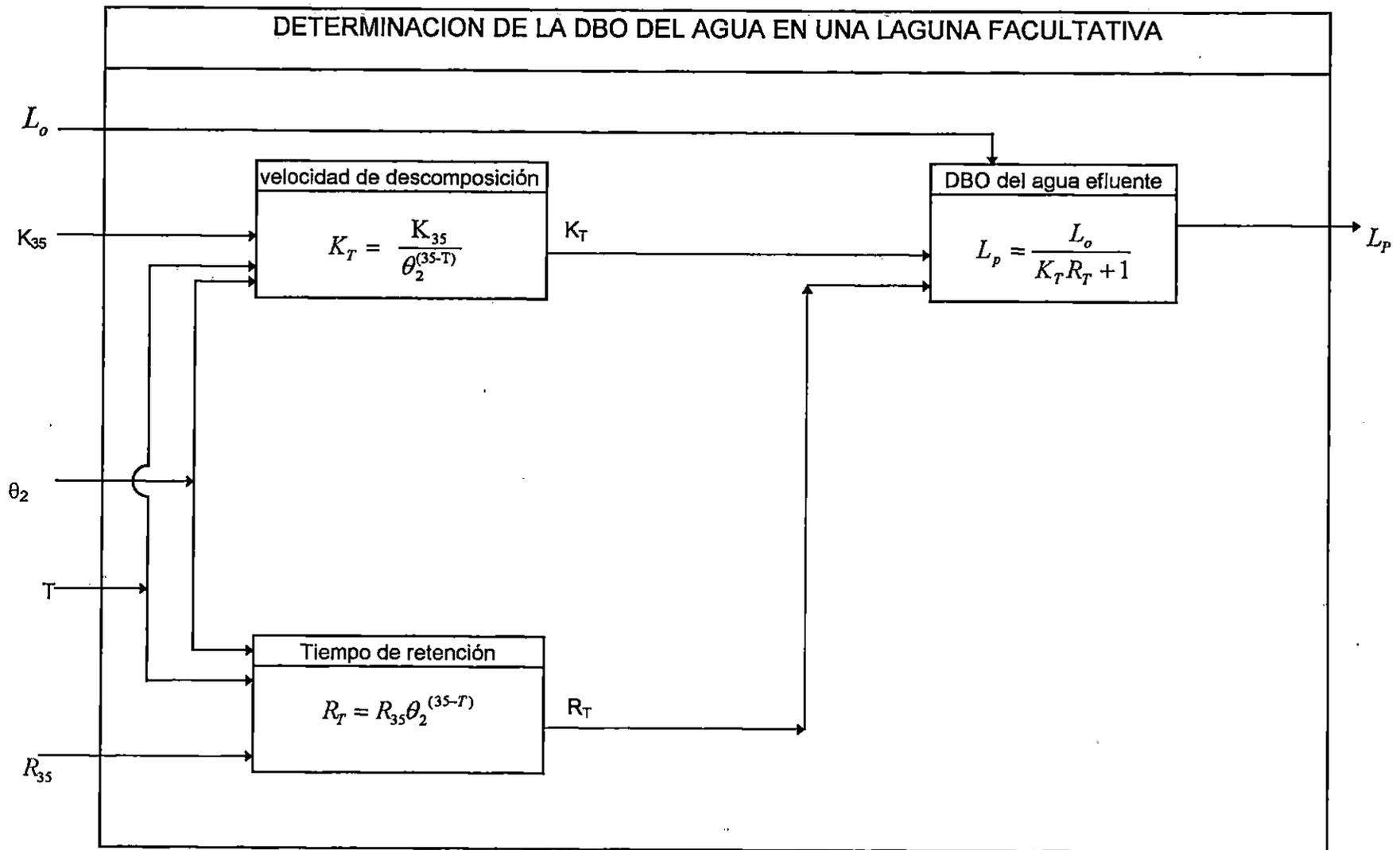


DIAGRAMA 2.E DETERMINACION DE LA DBO DEL AGUA EN UNA LAGUNA FACULTATIVA

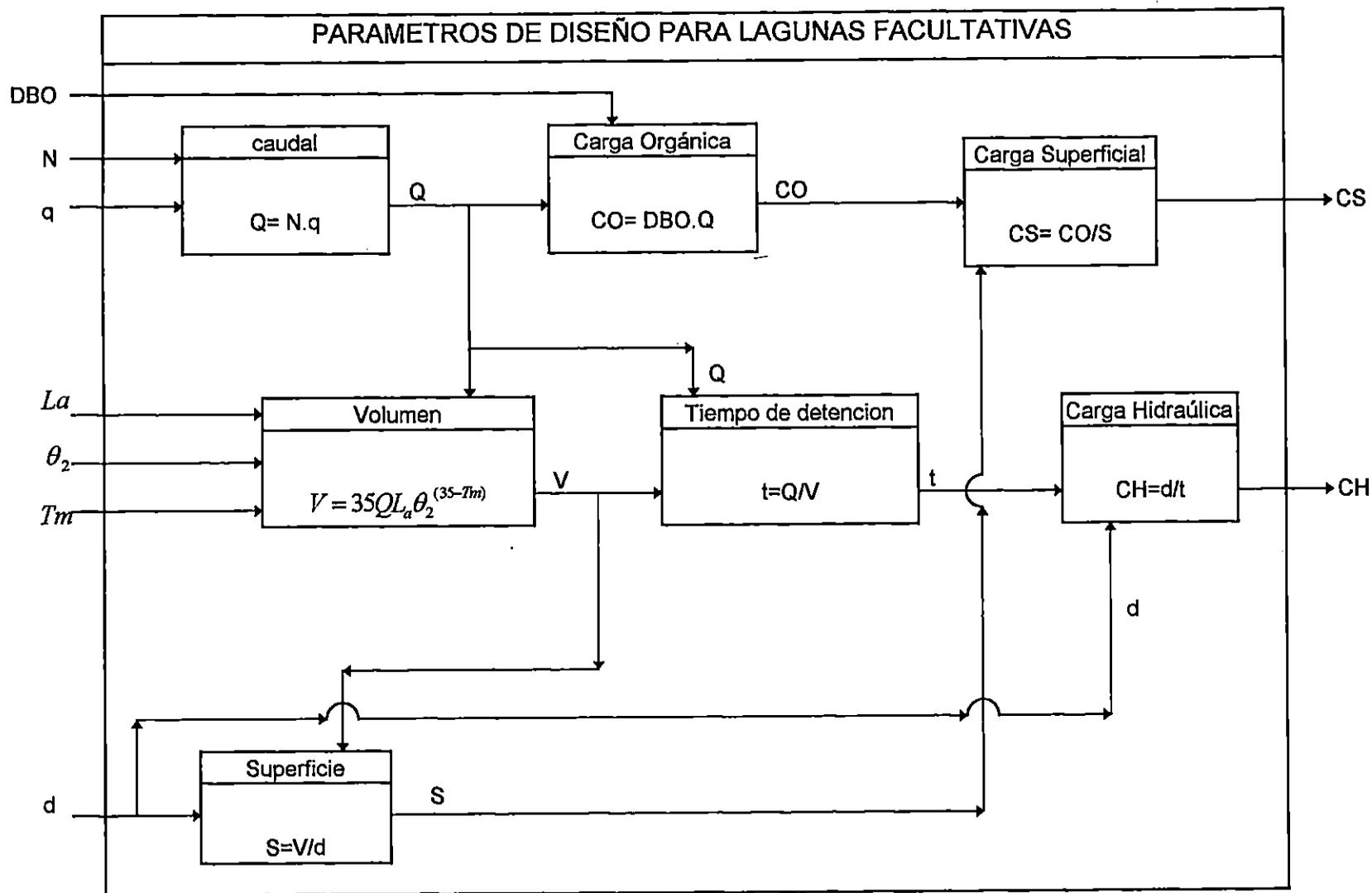


DIAGRAMA 2.F PARAMETROS DE DISEÑO PARA LAGUNAS FACULTATIVAS

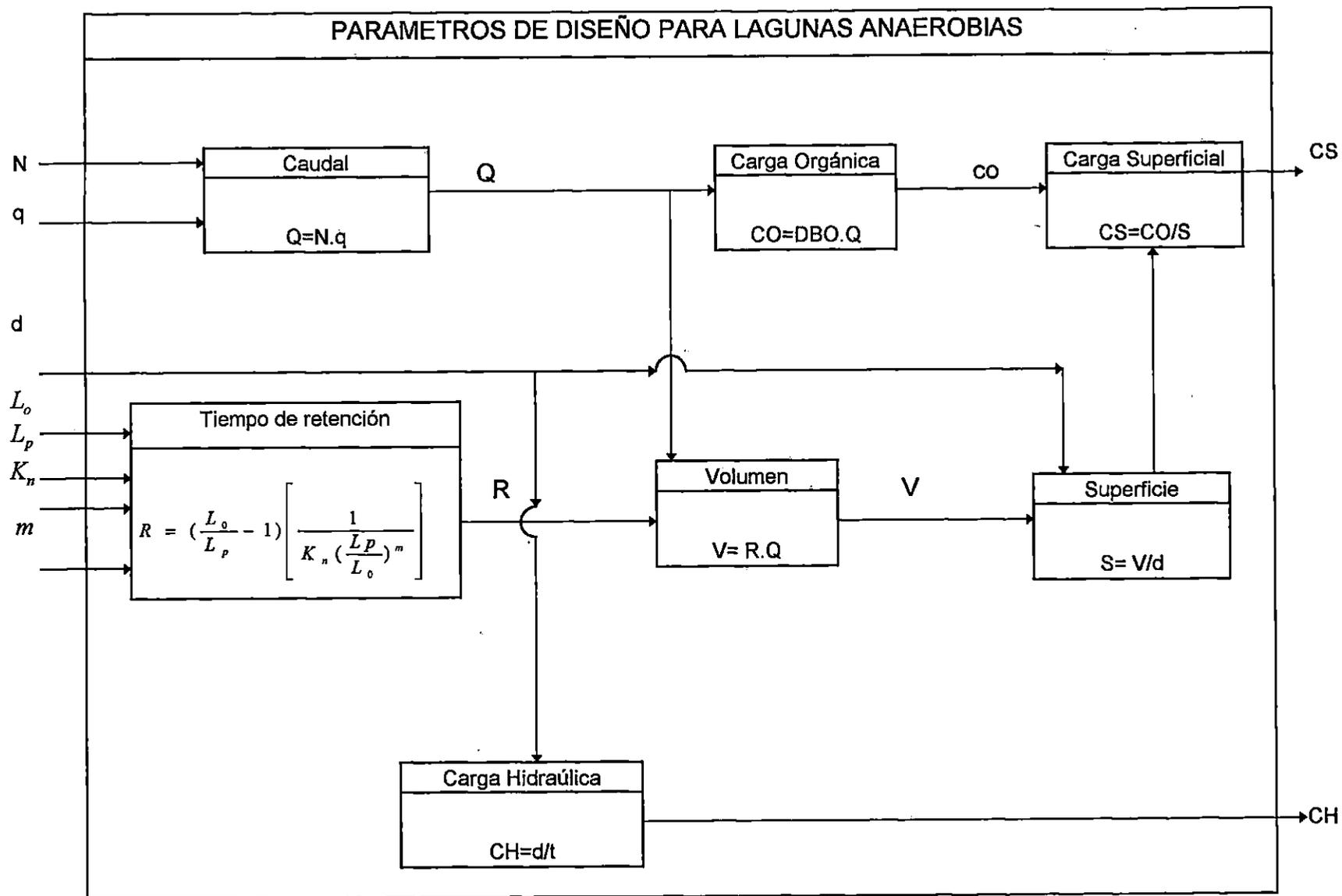


DIAGRAMA 2.G PARAMETROS DE DISEÑO PARA LAGUNAS ANAEROBIAS

2.8 PROGRAMA PARA LAGUNAS DE ESTABILIZACION EN QUICK C

```

#include "conio.h"
#include "stdio.h"
#include "stdlib.h"
#include "dos.h"
#include "stdarg.h"
#include "math.h"

/* ----- FUNCIONES COMUNES ----- */
int CreaVentanaMensaje(int MensajeUnico, int NumeroPalabras,...);
void Cursor(int row, int col);
void BorraBloque(int top, int left, int btm, int right, int attr);
void DespliegaCadena(char *str, int attr, int row, int col);
void pca(char c, int attr, int row, int col);
void Ventana(int trow, int lcol, int brow, int rcol, int attr, int border,
             char *title);
int getattr(void);
void cls(int attr);
int getvmode(int *iscolor);

#define IZQUIERDA      75
#define DERECHA       77
#define ARRIBA        72
#define ABAJO         80
#define F3             61
#define F1             59
#define ENTER         13
#define ESCAPE        27
#define WHITE         15
#define BLACK          0
#define BLUE           1

/* ----- Define's propios ----- */
#define OPCION_131 "Lagunas Aerobias"
#define OPCION_132 "Lagunas Facultativas"
#define OPCION_133 "Lagunas Anaerobias"
#define OPCION_134 "Aireaci3n Mec nica"
#define MENSAJE_131 "Lagunas Aerobias."
#define MENSAJE_132 "Lagunas Facultativas."
#define MENSAJE_133 "Lagunas Anaerobias."
#define MENSAJE_134 "Aireaci3n Mec nica."
#define MENSAJE_ERR01 "ERR01!: El Dato debe ser >= 0 y <= 100. Presione Cualquier Tecla."
#define MENSAJE_ERR02 "ERR02!: El Dato debe ser > 0. Presione Cualquier Tecla."
#define MENSAJE_ERR03 "ERR03!: El Mes debe estar entre [1,12]. Presione Cualquier Tecla."
#define MENSAJE_ERR04 "ERR04!: Memoria Insuficiente para esta Opci3n. Presione ESCAPE."
#define MAGENTA      5
#define CONST_A      0.90132612
#define CONST_B      -0.0095455207
#define CONST_C      0.000022472713
#define CONST_D      26.485764
#define EXP          2.718
#define DIF          0.1

/* ----- */

int i, PosicionMenu;
int attr0, attr1, attr2, attr3, iscolor;
char temp[80], Imprimir[1];
FILE *pfile5;

/* ----- Variables propias ----- */
char Respuesta[1];
int SalirOpcion_13, NumeroCorrecto, Reconstruye, attr4, NivelRegeneracion;
int NumeroPersonas, MesLaguna, ZonaCorrecta, NumeroIntentos;
double ProduccionResidual, Caudal, DemandaBO, EnergiaLuminosa, CargaOrganica;
char NumeroEnCadena[80], ZonaLaguna[2], temp1[80];
double MatrizSolar[9][12], LuzSolar, ProduccionOxigeno, ProduccionAlgas;
double PenetracionOxigeno, Longitud, Ancho, Profundidad, Volumen;
double ProfundidadLaguna, Superficie, CargaHidraulica, CargaSuperficial;
double TemperaturaEstanque, VelocidadReaccion, PctSustrato, PSC, PSC1, PSC2, PSC3;
double DifSustrato, FactDispersion, ConstA, TiempoDetencion;

```

```

double Rendimiento,Saturacion,ODisuelto,TODisuelto,CoefARAG;
double CoefTemperatura, TemperaturaAgua,AireacionMecanica;
double CoefAAS,PctDBO,DBOReducida,AireacionSuplementaria,DBOTotal;
double NecesidadEnergia, VelocidadDescomposicion, TiempoDetencion35C;
double DBOEfluente,DBOAfluente, TemperaturaMedia, CoefProyecto, Exponente;

```

```

/* ----- Funciones propias ----- */
int LagunasAerobias(void);
int LagunasFacultativas(void);
int LagunasFacultativas(void);
double BuscaLuzSolar(int mes);
int InicializaMatrizSolar(void);
int HallaZona(void);
int AMecanica(void);
void VentanaNoBorra(int trow, int lcol, int brow, int rcol, int attr, int border,
char *title);

```

```

/* ----- */
void Cursor(int row, int col)
/* ----- */
{
/* env/a el Cursor a la Posici3n (row=0,col=0 Origin) */
union REGS rin,rout;
rin.h.ah=2;
rin.h.bh=0;
rin.h.dh=row;
rin.h.dl=col;
int86(0x10,&rin,&rout);
}

```

```

/* ----- */
void DespliegaCadena(char *str, int attr, int row, int col)
/* ----- */
{
/* Escribe una Cadena con Atributo y coordenadas */
while (*str)
pca(*str++,attr,row,col++);
}

```

```

/* ----- */
void BorraBloque(int top, int left, int btm, int right, int attr)
/* ----- */
{
/* Clears Screen Block w/Attribute, Cursor Inside */
union REGS rin,rout;
rin.h.ah=6;
rin.h.al=0;
rin.h.bh=attr;
rin.h.ch=top; /* top row */
rin.h.cl=left; /* top column */
rin.h.dh=btm; /* bottom row */
rin.h.dl=right; /* bottom column */
int86(0x10,&rin,&rout);
Cursor(top+1,left+1);
}

```

```

/* ----- */
int getattr(void)
/* ----- */
{
union REGS rin,rout;
rin.h.ah=8; /* get attribute at Cursor */
rin.h.bh=0;
int86(0x10,&rin,&rout);
return((int)rout.h.ah);
}

```

```

/* ----- */
void pca(char c, int attr, int row, int col)
/* ----- */
{
union REGS rin,rout;
Cursor(row,col); /* set cursor */
}

```

```

rin.h.ah=9;          /* print char with attribute */
rin.h.al=c;
rin.h.bh=0;
rin.h.bl=attr;
rin.x.cx=1;
int86(0x10,&rin,&rout);
}

/*-----*/
void Ventana(int trow, int lcol, int brow, int rcol, int attr,
int border, char *title)
/*-----*/
{
    /* Set Simple Window and Title */
    int tl,tr,bl,br,lr,tb,i,length;
    BorraBloque(trow,lcol,brow,rcol,attr);
    if(border==1){          /* Single Border */
        tl=218; tr=191; bl=192; br=217; lr=196; tb=179;
    }
    else{                  /* Double Border */
        tl=201; tr=187; bl=200; br=188; lr=205; tb=186;
    }
    pca(tl,attr,trow,lcol);
    pca(bl,attr,brow,lcol);
    pca(tr,attr,trow,rcol);
    pca(br,attr,brow,rcol);
    for(i=lcol+1; i<rcol; i++){
        pca(lr,attr,trow,i);
        pca(lr,attr,brow,i);
    }
    for(i=trow+1; i<brow; i++){
        pca(tb,attr,i,lcol);
        pca(tb,attr,i,rcol);
    }
    i=0;
    while(title[i++]!='\0');
    length=i-1;
    DespliegaCadena(title,attr,trow,(rcol+lcol)/2-length/2);
}

/*-----*/
void cls(int attr)
/*-----*/
{
    /* Clears Screen w/Attribute, Cursor to Origin */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.al=0;
    rin.h.bh=attr;
    rin.h.ch=0; /* top row */
    rin.h.ci=0; /* top column */
    rin.h.dh=24; /* bottom row */
    rin.h.dl=79; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(0,0);
}

/*-----*/
int getvmode(int *iscolor)
/*-----*/
{
    union REGS rin,rout;
    int mode;
    rin.h.ah=15;          /* get display mode */
    int86(0x10,&rin,&rout);
    mode=(int)rout.h.al;
    *iscolor=1;
    if(mode==0 || mode==2 || mode==7)
        *iscolor=0;
    return(mode);        /* sets color flag, returns video mode */
}

```

```

/*-----*/
int CreaVentanaMensaje(int Unico, int NumeroPalabras, ...)
/*-----*/
{
    int Bandera = 1;
    int Longitud;
    char *Rotulo;

    va_list parameters;
    va_start(parameters, NumeroPalabras);

    strcpy(temp, "");
    Ventana(21,0,23,79,attr1,1,temp);
    BorraBloque(22,1,22,78,attr2);
    if (Unico)
    {
        Rotulo = va_arg(parameters, char *);
        DespliegaCadena(Rotulo,attr1,22,3);
        return(0);
    }
    Longitud = 3;
    for (i=0; i<NumeroPalabras; i++)
    {
        Rotulo = va_arg(parameters, char *);
        if (Bandera)
        {
            DespliegaCadena(Rotulo,attr3,22,3+Longitud);
            Bandera = 0;
        }
        else
        {
            DespliegaCadena(Rotulo,attr1,22,3+Longitud);
            Bandera = 1;
        }
        Longitud = Longitud + strlen(Rotulo) + 2;
    }
    return(0);
}

/*-----*/
int CapturaNumero(char *fid, int row, int col, int maxden, int attr)
/*-----*/
/* Simple Numeric Input Routine
 * fid storage for number as char string
 * row,col starting cursor position for field
 * maxden field length
 * attr entry field attribute
 * returns entry length(x), -entry length(y), 0(ESC hit) */
{
    int i,j,yflag,expflag,pass;
    char c;

    for(j=0;j<maxden;j++) pca(32,attr,row,col+j);
    Cursor(row,col);
    i=j=0;
    while(j<maxden)
    {
        c=getch();
        pass=0;
        if((c>='0' && c<='9') || c=='.' || pass)
        {
            pca(c,attr,row,col+i); /* print character */
            Cursor(row,col+i+1); /* advance cursor */
            i++;
            if(!pass) fid[j++]=c;
        }
        else
            if (c==27) return(-2);
            else
                {

```

```

if((c==10 || c==13 || i==maxden) && i!=0) break;
else
  if(c=='\b' && i>0)
  {
    i--;
    pca(32,attr,row,col+i);
    if(j!=0)
    {
      j--;
      fld[j]=0;
    }
  }
  else if(c==27) return(0);
}
}
fld[j]='\0';
return(i);
}

/*-----*/
double RetomaDbI(int NumeroEval,int fil,int col,int cifras)
/*-----*/
{
  double NumeroDbI;
  NumeroCorrecto = -1;
  while (NumeroCorrecto < 0)
  {
    NumeroCorrecto = CapturaNumero(NumeroEnCadena,fil,col,cifras,attr0);
    if (NumeroCorrecto >= 0) NumeroDbI = atof(NumeroEnCadena);
    else return(0);
    switch (NumeroEval)
    {
      case 1:
      {
        if (NumeroDbI < 0)
        {
          CreaVentanaMensaje(1,1,MENSAJE_ERR02);
          while (!kbhit());
          CreaVentanaMensaje(0,2,"ESC","Salir");
          NumeroCorrecto = -1;
        }
        break;
      }
      case 2:
      {
        if ((NumeroDbI < 0) || (NumeroDbI > 100))
        {
          CreaVentanaMensaje(1,1,MENSAJE_ERR01);
          while (!kbhit());
          CreaVentanaMensaje(0,2,"ESC","Salir");
          NumeroCorrecto = -1;
        }
        break;
      }
      case 3:
      {
        if ((NumeroDbI < 1) || (NumeroDbI > 12))
        {
          CreaVentanaMensaje(1,1,MENSAJE_ERR03);
          while (!kbhit());
          CreaVentanaMensaje(0,2,"ESC","Salir");
          NumeroCorrecto = -1;
        }
      }
    }
  }
  return(NumeroDbI);
}

/*----- funciones propias -----*/

```

```

/*-----*/
void VentanaNoBorra(int trow, int lcol, int brow, int rcol, int attr,
int border, char *title)
/*-----*/
{
    /* Set Simple Window and Title */
    int ti,tr,bl,br,lr,tb,i,length;
    if(border==1){ /* Single Border */
        ti=218; tr=191; bl=192; br=217; lr=196; tb=179;
    }
    else{ /* Double Border */
        ti=201; tr=187; bl=200; br=188; lr=205; tb=186;
    }
    pca(ti,attr,trow,lcol);
    pca(bl,attr,brow,lcol);
    pca(tr,attr,trow,rcol);
    pca(br,attr,brow,rcol);
    for(i=lcol+1; i<rcol; i++){
        pca(lr,attr,trow,i);
        pca(lr,attr,brow,i);
    }
    for(i=trow+1; i<brow; i++){
        pca(tb,attr,i,lcol);
        pca(tb,attr,i,rcol);
    }
    i=0;
    while(title[i++]!='\0');
    length=i-1;
    DespliegaCadena(title,attr,trow,(rcol+lcol)/2-length/2);
}

/*-----*/
int InicializaMatrizSolar(void)
/*-----*/
{
    int i,j;

    /* Reservando en Memoria */
    if (malloc(108 * sizeof(double)) == NULL) return (-1);

    MatrizSolar[0][0] = 493.0;
    MatrizSolar[0][1] = 538.2;
    MatrizSolar[0][2] = 561.5;
    MatrizSolar[0][3] = 551.2;
    MatrizSolar[0][4] = 461.6;
    MatrizSolar[0][5] = 444.0;
    MatrizSolar[0][6] = 487.6;
    MatrizSolar[0][7] = 465.8;
    MatrizSolar[0][8] = 408.2;
    MatrizSolar[0][9] = 423.8;
    MatrizSolar[0][10] = 432.2;
    MatrizSolar[0][11] = 429.6;

    MatrizSolar[1][0] = 418.4;
    MatrizSolar[1][1] = 458.4;
    MatrizSolar[1][2] = 479.5;
    MatrizSolar[1][3] = 473.0;
    MatrizSolar[1][4] = 473.9;
    MatrizSolar[1][5] = 421.4;
    MatrizSolar[1][6] = 473.7;
    MatrizSolar[1][7] = 442.4;
    MatrizSolar[1][8] = 390.7;
    MatrizSolar[1][9] = 403.3;
    MatrizSolar[1][10] = 411.5;
    MatrizSolar[1][11] = 409.0;

    MatrizSolar[2][0] = 422.0;
    MatrizSolar[2][1] = 467.3;
    MatrizSolar[2][2] = 489.2;
    MatrizSolar[2][3] = 487.4;
    MatrizSolar[2][4] = 446.0;
}

```

MatrizSolar[2][5] = 455.4;
MatrizSolar[2][6] = 472.4;
MatrizSolar[2][7] = 472.3;
MatrizSolar[2][8] = 442.0;
MatrizSolar[2][9] = 438.6;
MatrizSolar[2][10] = 422.0;
MatrizSolar[2][11] = 402.5;

MatrizSolar[3][0] = 422.3;
MatrizSolar[3][1] = 465.1;
MatrizSolar[3][2] = 489.8;
MatrizSolar[3][3] = 472.0;
MatrizSolar[3][4] = 433.0;
MatrizSolar[3][5] = 442.6;
MatrizSolar[3][6] = 465.0;
MatrizSolar[3][7] = 449.3;
MatrizSolar[3][8] = 427.2;
MatrizSolar[3][9] = 427.7;
MatrizSolar[3][10] = 404.1;
MatrizSolar[3][11] = 405.8;

MatrizSolar[4][0] = 395.3;
MatrizSolar[4][1] = 445.7;
MatrizSolar[4][2] = 457.3;
MatrizSolar[4][3] = 439.0;
MatrizSolar[4][4] = 414.4;
MatrizSolar[4][5] = 406.5;
MatrizSolar[4][6] = 443.0;
MatrizSolar[4][7] = 447.2;
MatrizSolar[4][8] = 392.8;
MatrizSolar[4][9] = 387.0;
MatrizSolar[4][10] = 391.8;
MatrizSolar[4][11] = 376.9;

MatrizSolar[5][0] = 419.4;
MatrizSolar[5][1] = 460.4;
MatrizSolar[5][2] = 476.2;
MatrizSolar[5][3] = 447.3;
MatrizSolar[5][4] = 414.6;
MatrizSolar[5][5] = 413.1;
MatrizSolar[5][6] = 462.4;
MatrizSolar[5][7] = 466.3;
MatrizSolar[5][8] = 414.6;
MatrizSolar[5][9] = 413.5;
MatrizSolar[5][10] = 409.9;
MatrizSolar[5][11] = 413.2;

MatrizSolar[6][0] = 398.4;
MatrizSolar[6][1] = 427.3;
MatrizSolar[6][2] = 414.3;
MatrizSolar[6][3] = 370.2;
MatrizSolar[6][4] = 309.1;
MatrizSolar[6][5] = 331.3;
MatrizSolar[6][6] = 405.7;
MatrizSolar[6][7] = 351.9;
MatrizSolar[6][8] = 294.6;
MatrizSolar[6][9] = 336.6;
MatrizSolar[6][10] = 366.3;
MatrizSolar[6][11] = 369.7;

MatrizSolar[7][0] = 375.0;
MatrizSolar[7][1] = 420.9;
MatrizSolar[7][2] = 440.4;
MatrizSolar[7][3] = 389.6;
MatrizSolar[7][4] = 389.6;
MatrizSolar[7][5] = 367.3;
MatrizSolar[7][6] = 393.7;
MatrizSolar[7][7] = 365.6;
MatrizSolar[7][8] = 314.4;
MatrizSolar[7][9] = 326.5;

```

MatrizSolar[7][10] = 320.8;
MatrizSolar[7][11] = 328.8;

MatrizSolar[8][0] = 341.2;
MatrizSolar[8][1] = 296.5;
MatrizSolar[8][2] = 392.3;
MatrizSolar[8][3] = 357.7;
MatrizSolar[8][4] = 332.3;
MatrizSolar[8][5] = 342.5;
MatrizSolar[8][6] = 370.5;
MatrizSolar[8][7] = 366.1;
MatrizSolar[8][8] = 298.2;
MatrizSolar[8][9] = 232.1;
MatrizSolar[8][10] = 303.6;
MatrizSolar[8][11] = 308.4;

return 0;
}

/*-----*/
int HallaZona(void)
/*-----*/
{
int Indice;
if (strcmp(ZonaLaguna,"AH") == 0)
{
Indice = 0;
return(Indice);
}
if (strcmp(ZonaLaguna,"SS") == 0)
{
Indice = 1;
return(Indice);
}
if (strcmp(ZonaLaguna,"NC") == 0)
{
Indice = 2;
return(Indice);
}
if (strcmp(ZonaLaguna,"UN") == 0)
{
Indice = 3;
return(Indice);
}
if (strcmp(ZonaLaguna,"AQ") == 0)
{
Indice = 4;
return(Indice);
}
if (strcmp(ZonaLaguna,"CA") == 0)
{
Indice = 5;
return(Indice);
}
if (strcmp(ZonaLaguna,"MO") == 0)
{
Indice = 6;
return(Indice);
}
if (strcmp(ZonaLaguna,"LG") == 0)
{
Indice = 7;
return(Indice);
}
if (strcmp(ZonaLaguna,"PI") == 0)
{
Indice = 8;
return(Indice);
}
return(-1);
}

```

```

}
/*-----*/
double BuscaLuzSolar(int mes)
/*-----*/
{
    int Fallo;
    double Valor;

    mes = mes - 1;
    Fallo = InicializaMatrizSolar();
    if (!Fallo)
    {
        Valor = MatrizSolar[ZonaCorrecta][mes];
        return(Valor);
    }
    else return(0);
}

/*-----*/
int LagunasAerobias(void)
/*-----*/
{
    BorraBloque(0,0,20,79,attr2);
    strcpy(temp,"Producción de Oxígeno");
    strcpy(temp1,"");
    Ventana(0,0,5,79,attr2,2,temp);

    Ventana(18,0,20,79,attr2,1,temp1);
    strcpy(Respuesta[0],"Q");
    while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
    {
        DespliegaCadena("Desea Calcular el Caudal (S/N)?",attr1,19,3);
        Cursor(19,35);
        Respuesta[0] = getch();
        if (Respuesta[0] == 27) return(-1);
    }
    if (Respuesta[0] == 'S')
    {
        NumeroPersonas = 0;
        BorraBloque(19,3,19,70,attr2);
        DespliegaCadena("Número de Personas que aportan agua:",attr1,19,3);
        NumeroCorrecto = -1;
        while (NumeroCorrecto < 0)
        {
            NumeroCorrecto = CapturaNumero(NumeroEnCadena,19,40,7,attr0);
            if (NumeroCorrecto>0) NumeroPersonas = atoi(NumeroEnCadena);
            else
            {
                if (NumeroCorrecto==2) return(-1);
                CreaVentanaMensaje(1,1,MENSAJE_ERR02);
                while (!kbhit());
                CreaVentanaMensaje(0,2,"ESC","Salir");
                NumeroCorrecto = -1;
            }
        }
        BorraBloque(19,3,19,70,attr2);
        DespliegaCadena("Producción Residual por Persona:",attr1,19,3);
        ProduccionResidual = RetomaDb(1,19,36,9);
        if (NumeroCorrecto==2) return(-1); /* presión ESC */

        Caudal = NumeroPersonas * ProduccionResidual;
        printf(NumeroEnCadena,"Caudal: %5.3f (m3/dia)",Caudal);
        DespliegaCadena(NumeroEnCadena,attr1,1,3);
        BorraBloque(18,0,20,79,attr2);
    }
    else
    {
        BorraBloque(18,0,20,79,attr2);
        DespliegaCadena("Caudal (m3/dia):",attr1,1,3);
    }
}

```

```

Caudal = RetomaDb(1,1,19,9);
if (NumeroCorrecto==2) return(-1);
}
DespliegaCadena("Demanda Bioquímica de Oxígeno:",attr1,2,3);
DemandaBO = RetomaDb(1,2,34,9);
if (NumeroCorrecto==2) return(-1);
CargaOrganica = DemandaBO * Caudal;
EnergiaLuminosa = 1.3 * (log(CargaOrganica) - 3.5);
sprintf(NumeroEnCadena,"Energía Luminosa (F): %5.3f",EnergiaLuminosa);
DespliegaCadena(NumeroEnCadena,attr1,3,3);

Ventana(18,0,20,79,attr2,1,temp1);
Respuesta[0] = '';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
    DespliegaCadena("Desea Calcular el Valor de Luz Solar (S/N)?",attr1,19,3);
    Cursor(19,47);
    Respuesta[0] = getch();
    if (Respuesta[0]==27) return(-1);
}
if (Respuesta[0] == 'S')
{
    strcpy(temp,"Zonas del País");
    Ventana(7,15,17,45,attr1,1,temp);
    DespliegaCadena("AH Ahuachapan",attr1,8,16);
    DespliegaCadena("SS San Salvador",attr1,9,16);
    DespliegaCadena("NC Nva. Concepción, Chalate.",attr1,10,16);
    DespliegaCadena("UN La Unión",attr1,11,16);
    DespliegaCadena("AQ Apastepeque",attr1,12,16);
    DespliegaCadena("CA La Carrera",attr1,13,16);
    DespliegaCadena("MO Montecristo",attr1,14,16);
    DespliegaCadena("LG La Galera",attr1,15,16);
    DespliegaCadena("PI Las Pilas",attr1,16,16);
    BorraBloque(22,3,22,70,attr2);
    ZonaCorrecta = -1;
    strcpy(ZonaLaguna, ' ');
    BorraBloque(19,3,19,70,attr2);
    DespliegaCadena("Zona donde se diseñar la laguna:",attr1,19,3);
    while(ZonaCorrecta == -1)
    {
        CreaVentanaMensaje(1,1,"Introduzca el código de la Zona Geográfica.");
        Cursor(19,37);
        gets(ZonaLaguna);
        ZonaCorrecta = HallaZona();
    }
    BorraBloque(7,15,17,45,attr2);
    BorraBloque(19,3,19,70,attr2);
    CreaVentanaMensaje(1,1,"Introduzca el código del 1 al 12");
    DespliegaCadena("Mes en que se trabajar la laguna:",attr1,19,3);
    MesLaguna = RetomaDb(3,19,38,3);
    if (Respuesta[0] == 27) return(-1);
    LuzSolar = BuscaLuzSolar(MesLaguna);
    sprintf(NumeroEnCadena,"Luz Solar (l): %5.3f (cal x m2 x día)",LuzSolar);
    DespliegaCadena(NumeroEnCadena,attr1,4,3);
    BorraBloque(18,0,20,79,attr2);
}
else
{
    BorraBloque(18,0,20,79,attr2);
    DespliegaCadena("Luz Solar (cal x m2 x día):",attr1,4,3);
    LuzSolar = RetomaDb(1,4,31,9);
    if (NumeroCorrecto==2) return(-1);
}

ProduccionOxigeno = 0.25 * EnergiaLuminosa * LuzSolar;
Ventana(7,0,11,79,attr4,2,temp);
sprintf(NumeroEnCadena,"Producción de Oxígeno: %5.3f (Kg-O2/Ha-Día)",ProduccionOxigeno);
DespliegaCadena(NumeroEnCadena,attr4,8,3);
ProduccionAlgas = 0.15 * EnergiaLuminosa * LuzSolar;
sprintf(NumeroEnCadena,"Producción de Células de Algas: %5.3f (Kg-algas/Ha-Día)",ProduccionAlgas);

```

```

DespliegaCadena(NumeroEnCadena,attr4,9,3);
PenetracionOxigeno = CONST_A + (CONST_B * DemandaBO);
PenetracionOxigeno = PenetracionOxigeno + (CONST_C * DemandaBO * DemandaBO);
PenetracionOxigeno = PenetracionOxigeno + (CONST_D / DemandaBO);
sprintf(NumeroEnCadena,"Penetraci3n de Oxigeno: %5.3f (KgDBO/Ha-Dia)",PenetracionOxigeno);
DespliegaCadena(NumeroEnCadena,attr4,10,3);
CreaVentanaMensaje(1,1,"Digite Cualquier Tecla para continuar.");
while(!kbhit());

BorraBloque(0,0,20,79,attr2);
CreaVentanaMensaje(1,1,"Definici3n de Par metros de Dise1o.");
strcpy(temp,"Par metros de Dise1o");
Ventana(0,0,6,79,attr1,1,temp);

BorraBloque(18,0,20,79,attr2);
Ventana(18,0,20,79,attr1,1,temp1);
Respuesta[0] = '';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
    DespliegaCadena("Desea Calcular el Tiempo de Detenci3n (S/N)?",attr1,19,3);
    Cursor(19,48);
    Respuesta[0] = getche();
    if (Respuesta[0]==27) return(-1);
}
if (Respuesta[0] == 'S')
{
    Respuesta[0] = '';
    while (Respuesta[0] != 'V' && Respuesta[0] != 'S')
    {
        DespliegaCadena("Por el Volumen o por el Sustrato Remanente (V/S)?",attr1,19,3);
        Cursor(19,53);
        Respuesta[0] = getche();
        if (Respuesta[0]==27) return(-1);
    }

    if (Respuesta[0] == 'V')
    {
        BorraBloque(19,3,19,70,attr2);
        Respuesta[0] = '';
        while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
        {
            DespliegaCadena("Desea Calcular el Volumen (S/N)?",attr1,19,3);
            Cursor(19,36);
            Respuesta[0] = getche();
            if (Respuesta[0]==27) return(-1);
        }
        BorraBloque(19,3,19,70,attr2);
        if (Respuesta[0] == 'S')
        {
            DespliegaCadena("Longitud: ",attr1,19,3);
            Longitud = RetomaDbf(1,19,13,9);
            if (NumeroCorrecto==2) return(-1);
            DespliegaCadena("Anchura: ",attr1,19,23);
            Ancho = RetomaDbf(1,19,32,9);
            if (NumeroCorrecto==2) return(-1);
            DespliegaCadena("Profundidad: ",attr1,19,42);
            Profundidad = RetomaDbf(1,19,55,9);
            if (NumeroCorrecto==2) return(-1);
            Volumen = Longitud * Ancho * Profundidad;
        }
        else
        {
            DespliegaCadena("Volumen: ",attr1,19,3);
            Volumen = RetomaDbf(1,19,12,9);
            if (NumeroCorrecto == -2) return(-1);
        }
    }
    TiempoDetencion = Volumen / Caudal;
    sprintf(NumeroEnCadena,"Tiempo de Detenci3n: %5.3f (d1as)",TiempoDetencion);
    DespliegaCadena(NumeroEnCadena,attr1,1,3);
    BorraBloque(18,0,20,79,attr2);

```

```

}
else
{
    BorraBloque(19,3,19,70,attr2);
    DespliegaCadena("Temperatura del Estanque (°C): ",attr1,19,3);
    TemperaturaEstanque = RetomaDb(1,19,34,9);
    if (NumeroCorrecto==2) return(-1);

    VelocidadReaccion = 0.25 * pow(1.06,(TemperaturaEstanque - 20));
    BorraBloque(19,3,19,70,attr2);
    DespliegaCadena("% de Sustrato Permanente: ",attr1,19,3);
    PctSustrato = RetomaDb(2,19,29,9);
    if (NumeroCorrecto==2) return(-1);
    BorraBloque(19,3,19,70,attr2);
    DespliegaCadena("Factor de Dispersion: ",attr1,19,3);
    FactDispersion = RetomaDb(1,19,25,9);
    if (NumeroCorrecto==2) return(-1);
    PSC = 0;
    DifSustrato = 1;
    NumeroIntentos = 0;
    TiempoDetencion = 0;
    while(DifSustrato > DIF)
    {
        TiempoDetencion = TiempoDetencion + 1;
        ConstA = sqrt((1 + (4 * VelocidadReaccion * TiempoDetencion * FactDispersion)));
        PSC1 = 4 * ConstA * pow(EXP,(0.5 * FactDispersion));
        PSC2 = pow((1 + ConstA),2) * pow(EXP,(ConstA / (2 * FactDispersion)));
        PSC3 = pow((1 - ConstA),2) * pow(EXP,((-1 * ConstA) / (2 * FactDispersion)));
        PSC = (PSC1 / (PSC2 - PSC3)) * 100;
        DifSustrato = PSC - PctSustrato;
    }
    BorraBloque(18,0,20,79,attr2);
    if (TiempoDetencion == 0) TiempoDetencion = 1;
    sprintf(NumeroEnCadena,"Tiempo de Detención: %5.3f (dias)",TiempoDetencion);
    DespliegaCadena(NumeroEnCadena,attr1,1,3);
    Volumen = TiempoDetencion * Caudal;
}
}
else
{
    BorraBloque(18,0,20,79,attr2);
    DespliegaCadena("Tiempo de Detención:",attr1,1,3);
    NumeroCorrecto = -1;
    while (NumeroCorrecto < 0)
    {
        NumeroCorrecto = CapturaNumero(NumeroEnCadena,1,24,3,attr0);
        if (NumeroCorrecto>0) TiempoDetencion = atoi(NumeroEnCadena);
        else
        {
            if (NumeroCorrecto==2) return(-1);
            CreaVentanaMensaje(1,1,MENSAJE_ERR02);
            while (!kbhit());
            CreaVentanaMensaje(0,2,"ESC","Salir");
            NumeroCorrecto = -1;
        }
    }
    Volumen = TiempoDetencion * Caudal;
}

DespliegaCadena("Profundidad de la Laguna: ",attr1,2,3);
ProfundidadLaguna = RetomaDb(1,2,29,9);
if (NumeroCorrecto==2) return(-1);

Superficie = Volumen / ProfundidadLaguna;
sprintf(NumeroEnCadena,"Superficie: %5.3f m²",Superficie);
DespliegaCadena(NumeroEnCadena,attr1,3,3);

CargaHidraulica = ProfundidadLaguna / TiempoDetencion;
sprintf(NumeroEnCadena,"Carga Hidráulica: %5.3f (m/Día)",CargaHidraulica);
DespliegaCadena(NumeroEnCadena,attr1,4,3);

```

```

CargaSuperficial = CargaOrganica / Superficie;
sprintf(NumeroEnCadena,"Carga Superficial: %5.3f (KgDBO/Ha-Dia)",CargaSuperficial);
DespliegaCadena(NumeroEnCadena,attr1,5,3);

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
Ventana(21,0,23,79,attr2,1,temp);
DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
Cursor(22,25);
Imprimir[0] = getche();
if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
// impresion de encabezados en el impresor
if ((pfile5=fopen("LPT1","w"))==NULL)
{
BorraBloque(19,3,19,70,attr2);
DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
while(!kbhit());
return(0);
}
fprintf(pfile5,"\n\n***** Lagunas Aerobias *****\n");
fprintf(pfile5,"Energia Luminosa : %5.3f F ln",EnergiaLuminosa);
fprintf(pfile5,"Produccion de Oxigeno : %5.3f Kg-O2/Ha-Dia ln",ProduccionOxigeno);
fprintf(pfile5,"Produccion de Celulas de Algas : %5.3f Kg-algas/Ha-Dia ln",ProduccionAlgas);
fprintf(pfile5,"Penetracion de Oxigeno : %5.3f KgDBO/Ha-Dia ln",PenetracionOxigeno);
fprintf(pfile5,"Superficie : %5.3f m2 ln",Superficie);
fprintf(pfile5,"Carga Hidraulica : %5.3f m/Dia ln",CargaHidraulica);
fprintf(pfile5,"Carga Superficial : %5.3f KgDBO/Ha-Dia ln",CargaSuperficial);
fclose(pfile5);
}
CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
return(0);
}

/*-----*/
int AMecanica(void)
/*-----*/
{
BorraBloque(0,0,20,79,attr2);
strcpy(temp,"Aireaci3n Mec nica");
Ventana(0,0,17,79,attr2,1,temp);
CreaVentanaMensaje(1,1,"Titulo de la Aireaci3n Mec nica.");
DespliegaCadena("Rendimiento del Aireador s/Fabricante: ",attr1,1,3);
Rendimiento = RetornaDb(1,1,42,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Nivel de Saturaci3n de O2 en Agua Destilada a 20°C(Kg/m3): ",attr1,2,3);
Saturacion = RetornaDb(1,2,62,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Nivel de Oxigeno disuelto en el estanque (Kg/m3): ",attr1,3,3);
ODisuelto = RetornaDb(1,3,52,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Nivel de Saturaci3n de O2 a Temperatura T3C (Kg/m3):",attr1,4,3);
TODisuelto = RetornaDb(1,4,56,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Coef. Transf. Agua Residual/Transf. Agua grifo: ",attr1,5,3);
CoefARAG = 0;
NumeroCorrecto = -1;
while (NumeroCorrecto < 0)
{
CoefARAG = RetornaDb(1,5,51,9);
if (CoefARAG < 0.6 || CoefARAG > 1.1)
{
if (NumeroCorrecto==2) return(-1);
CreaVentanaMensaje(1,1,"N3mero debe ser > 0.6 y < 1.1");
}
}
}

```

```

        while (!kbhit());
        CreaVentanaMensaje(1,1,"Cliculo de la Aireaci3n Mec nica.");
        NumeroCorrecto = -1;
    }
}
CoefTemperatura = 1.02;
DespliegaCadena("Temperatura del Agua a tratar (3C): ",attr1,6,3);
TemperaturaAgua = RetomaDb(1,6,38,9);
if (NumeroCorrecto===-2) return(-1);
DespliegaCadena("Demanda Bioquimica de Oxigeno del Afluente(Kg/m3):",attr1,7,3);
DBOAfluente = RetomaDb(1,7,54,9);
if (NumeroCorrecto===-2) return(-1);
DespliegaCadena("Demanda Bioquimica de Oxigeno del Efluente(Kg/m3):",attr1,8,3);
DBOEfluente = RetomaDb(1,8,54,9);
if (NumeroCorrecto===-2) return(-1);
DespliegaCadena("Coeficiente a": ",attr1,9,3);
CoefAAS = 0;
NumeroCorrecto = -1;
while (NumeroCorrecto < 0)
{
    CoefAAS = RetomaDb(1,9,20,9);
    if (CoefAAS < 0.7 || CoefAAS > 1.4)
    {
        if (NumeroCorrecto===-2) return(-1);
        CreaVentanaMensaje(1,1,"N3mero debe ser >0.7 y < 1.4.");
        while (!kbhit());
        CreaVentanaMensaje(1,1,"Cliculo de la Aireaci3n Suplementaria.");
        NumeroCorrecto = -1;
    }
}
DespliegaCadena("Caudal(m3/dia):",attr1,10,3);
Caudal = RetomaDb(1,10,19,9);
if (NumeroCorrecto===-2) return(-1);
CargaOrganica = DBOAfluente * Caudal;
sprintf(NumeroEnCadena,"Carga Org nica(KgDBO/dia): %5.3f",CargaOrganica);
DespliegaCadena(NumeroEnCadena,attr1,10,35);
Ventana(18,0,20,79,attr2,2,temp1);
BorraBloque(19,3,19,70,attr2);
DespliegaCadena("% de DBO a reducir: ",attr1,19,3);
PctDBO = RetomaDb(2,19,23,9);
if (NumeroCorrecto===-2) return(-1);
PctDBO = PctDBO / 100;
DBOReducida = CargaOrganica * PctDBO;
BorraBloque(18,0,20,79,attr2);
sprintf(NumeroEnCadena,"DBO reducida: %5.3f (Kg/m3)",DBOReducida);
DespliegaCadena(NumeroEnCadena,attr1,11,3);
CoefTemperatura = 1.085;
VelocidadReaccion = 1.2;
Ventana(18,0,20,79,attr2,2,temp1);
Respuesta[0] = ' ';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
    DespliegaCadena("Conoce la Velocidad de Descomposici3n (S/N)? ",attr1,19,3);
    Cursor(19,49);
    Respuesta[0] = getch();
    if (Respuesta[0]==27) return(-1);
}
BorraBloque(18,0,20,79,attr2);
if (Respuesta[0] == 'S')
{
    DespliegaCadena("Velocidad de Descomposici3n: ",attr1,12,3);
    VelocidadDescomposicion = RetomaDb(1,12,32,9);
    if (NumeroCorrecto===-2) return(-1);
}
}
else
{
    VelocidadDescomposicion = VelocidadReaccion / pow(CoefTemperatura,(35 - TemperaturaAgua));
    sprintf(NumeroEnCadena,"Velocidad de Descomposici3n: %5.3f (Desc./dias)",VelocidadDescomposicion);
    DespliegaCadena(NumeroEnCadena,attr1,12,3);
}
}

```

```

TiempoDetencion = (DBOAfluente - DBOAfluente * PctDBO) / (VelocidadDescomposicion * DBOAfluente * PctDBO);
sprintf(NumeroEnCadena,"Tiempo de Detención %f dias",TiempoDetencion);
DespliegaCadena(NumeroEnCadena,attr4,13,3);
Volumen = TiempoDetencion * Caudal;
sprintf(NumeroEnCadena,"Volumen %5.3f m3",Volumen);
DespliegaCadena(NumeroEnCadena,attr4,14,3);
DespliegaCadena("Profundidad (m):",attr1,14,30);
Profundidad = RetomaDbl(1,14,47,9);
Superficie = Volumen / Profundidad;
sprintf(NumeroEnCadena,"Superficie %5.3f m²",Superficie);
DespliegaCadena(NumeroEnCadena,attr4,15,3);
AireacionMecanica = Rendimiento * ((TODisuelto - ODisuelto) / Saturacion) * CoefARAG;
AireacionMecanica = AireacionMecanica * pow(1.02, TemperaturaAgua-20);
sprintf(NumeroEnCadena,"Aireación Mecánica: %5.3f (Kg-O2/CV-h)",AireacionMecanica);
DespliegaCadena(NumeroEnCadena,attr4,15,35);
AireacionSuplementaria = CoefAAS * DBOReducida / 24;
sprintf(NumeroEnCadena,"Aireación Suplementaria %5.3f Kg/h",AireacionSuplementaria);
DespliegaCadena(NumeroEnCadena,attr4,16,3);
NecesidadEnergia = AireacionSuplementaria / AireacionMecanica;
sprintf(NumeroEnCadena,"Necesidad de Energía: %5.3f CV",NecesidadEnergia);
DespliegaCadena(NumeroEnCadena,attr4,16,45);

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,attr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getch();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(19,3,19,70,attr2);
        DespliegaCadena("ERROR! No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5, "\n\n***** Aireacion Mecanica *****\n");
    fprintf(pfile5,"Carga Organica      : %5.3f KgDBO/Dia \n",CargaOrganica);
    fprintf(pfile5,"DBO reducida      : %5.3f Kg/m3 \n",DBOReducida);
    fprintf(pfile5,"Tiempo de Detencion : %f dias \n",TiempoDetencion);
    fprintf(pfile5,"Volumen           : %5.3f m3 \n",Volumen);
    fprintf(pfile5,"Superficie        : %5.3f m2 \n",Superficie);
    fprintf(pfile5,"Aireacion Mecanica : %5.3f Kg-O2/CV-h \n",AireacionMecanica);
    fprintf(pfile5,"Aireacion Suplementaria : %5.3f Kg/h \n",AireacionSuplementaria);
    fprintf(pfile5,"Necesidad de Energia : %5.3f CV \n",NecesidadEnergia);
    fclose(pfile5);
}
CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
return(0);
}

/*-----*/
int LagunasFacultativas(void)
/*-----*/
{
    BorraBloque(0,0,20,79,attr2);
    CoefTemperatura = 1.085;
    strcpy(temp,"Lagunas Facultativas");
    CreaVentanaMensaje(1,1,"Cálculo de Lagunas Facultativas.");
    Ventana(0,0,16,79,attr2,1,temp);
    DespliegaCadena("Temperatura del Estanque (°C): ",attr1,1,3);
    TemperaturaEstanque = RetomaDbl(1,1,33,9);
    if (NumeroCorrecto==2) return(-1);
}

```

```

VelocidadReaccion = 1.2;
Ventana(18,0,20,79,attr2,2,temp1);
Respuesta[0] = ' ';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
  DespliegaCadena("Conoce la Velocidad de Descomposici n (S/N)? ",attr1,19,3);
  Cursor(19,49);
  Respuesta[0] = getche();
  if (Respuesta[0]==27) return(-1);
}
BorraBloque(18,0,20,79,attr2);
if (Respuesta[0] == 'S')
{
  DespliegaCadena("Velocidad de Descomposici n a T C: ",attr1,2,3);
  VelocidadDescomposicion = RetornaDbI(1,2,37,9);
  if (NumeroCorrecto==2) return(-1);
}
else
{
  VelocidadDescomposicion = VelocidadReaccion / pow(CoefTemperatura,(35 - TemperaturaEstanque));
  sprintf(NumeroEnCadena,"Velocidad de Descomposici n a T C: %5.3f",VelocidadDescomposicion);
  DespliegaCadena(NumeroEnCadena,attr1,2,3);
}

TiempoDetencion35C = 0;
while(TiempoDetencion35C < 6 || TiempoDetencion35C > 40)
{
  DespliegaCadena("Tiempo de Detencion(35 C): ",attr1,3,3);
  TiempoDetencion35C = RetornaDbI(1,3,30,9);
  if (NumeroCorrecto==2) return(-1);
  if (TiempoDetencion35C < 6 || TiempoDetencion35C > 40)
  {
    CreaVentanaMensaje(1,1,"Tiempo de Detencion a 35 C debe estar [6-40] dias.");
    while(!kbhit());
    CreaVentanaMensaje(1,1,"C lculo de Lagunas Facultativas.");
  }
}
TiempoDetencion = TiempoDetencion35C * pow(CoefTemperatura,(35 - TemperaturaEstanque));

DespliegaCadena("Demanda Bioqu mica de O5 inicial del Afluente (Kg/m3):",attr1,4,3);
DBOAfluente = RetornaDbI(1,4,58,9);
if (NumeroCorrecto==2) return(-1);
Ventana(18,0,20,79,attr2,2,temp1);
Respuesta[0] = ' ';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
  DespliegaCadena("Conoce la Demanda Bioqu mica de O5 del Efluente (S/N)? ",attr1,19,3);
  Cursor(19,59);
  Respuesta[0] = getche();
  if (Respuesta[0]==27) return(-1);
}
BorraBloque(18,0,20,79,attr2);
if (Respuesta[0] == 'S')
{
  DespliegaCadena("Demanda Bioqu mica de Ox geno del Efluente (Kg/m3):",attr1,5,3);
  DBOEfluente = RetornaDbI(1,5,55,9);
  if (NumeroCorrecto==2) return(-1);
}
else
{
  DBOEfluente = DBOAfluente / ((VelocidadDescomposicion * TiempoDetencion) + 1);
  sprintf(NumeroEnCadena,"Demanda Bioqu mica de Ox geno del Efluente (Kg/m3): %5.3f",DBOEfluente);
  DespliegaCadena(NumeroEnCadena,attr1,5,3);
}
DespliegaCadena("Caudal (m3/dia):",attr1,6,3);
Caudal = RetornaDbI(1,6,20,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("N mero de Personas que aportan agua:",attr1,7,3);
NumeroPersonas = RetornaDbI(1,7,40,9);
if (NumeroCorrecto==2) return(-1);

```

```

Ventana(18,0,20,79,attr2,2,temp1);
Respuesta[0] = '';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
    DespliegaCadena("Conoce la Producción de Agua Residual (m3/día) (S/N)? ",attr1,19,3);
    Cursor(19,58);
    Respuesta[0] = getche();
    if (Respuesta[0]==27) return(-1);
}
if (Respuesta[0] == 'S')
{
    BorraBloque(18,0,20,79,attr2);
    DespliegaCadena("Producción de Agua Residual (m3/día):",attr1,8,3);
    ProduccionResidual = RetornaDbf(1,8,41,9);
    if (NumeroCorrecto==2) return(-1);
}
else
{
    BorraBloque(18,0,20,79,attr2);
    ProduccionResidual = Caudal / NumeroPersonas;
    sprintf(NumeroEnCadena,"Producción de Agua Residual (m3/día): %5.3f",ProduccionResidual);
    DespliegaCadena(NumeroEnCadena,attr1,8,3);
}

```

```

BorraBloque(16,0,16,79,attr2);
VentanaNoBorra(0,0,18,79,attr2,1,temp);
DespliegaCadena("Temperatura media del agua en el mes m s frío (°C):",attr1,9,3);
TemperaturaMedia = RetornaDbf(1,9,56,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Demanda Bioquímica Total (Kg/m3):",attr1,10,3);
DBOTotal = RetornaDbf(1,10,37,9);
if (NumeroCorrecto==2) return(-1);
Volumen = 35 * Caudal * DBOTotal * pow(CoefTemperatura,(35 - TemperaturaMedia));
sprintf(NumeroEnCadena,"Volumen de la Laguna %5.3f m3",Volumen);
DespliegaCadena(NumeroEnCadena,attr4,11,3);
DespliegaCadena("Profundidad de la Laguna (m):",attr1,12,3);
ProfundidadLaguna = RetornaDbf(1,12,33,9);
if (NumeroCorrecto==2) return(-1);
Superficie = Volumen / ProfundidadLaguna;
sprintf(NumeroEnCadena,"Superficie de la Laguna %5.3f m²",Superficie);
DespliegaCadena(NumeroEnCadena,attr4,13,3);
CargaOrganica = DBOTotal * Caudal;
sprintf(NumeroEnCadena,"Carga Orgánica(KgDBO/día): %5.3f",CargaOrganica);
DespliegaCadena(NumeroEnCadena,attr1,14,3);
CargaSuperficial = CargaOrganica / Superficie;
sprintf(NumeroEnCadena,"Carga Superficial(KgDBO/Ha-día): %5.3f",CargaSuperficial);
DespliegaCadena(NumeroEnCadena,attr1,15,3);
CargaHidraulica = ProfundidadLaguna / TiempoDetencion;
sprintf(NumeroEnCadena,"Carga Hidráulica (KgDBO/Día): %5.3f",CargaHidraulica);
DespliegaCadena(NumeroEnCadena,attr4,16,3);
TiempoDetencion = Volumen / Caudal;
sprintf(NumeroEnCadena,"Tiempo de Detención: %5.3f días",TiempoDetencion);
DespliegaCadena(NumeroEnCadena,attr4,17,3);

```

```

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,attr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getche();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(19,3,19,70,attr2);
    }
}

```

```

DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",atr1,19,3);
while(!kbhit());
return(0);
}
fprintf(pfile5, "\n\n***** Lagunas Facultativas *****\n");
fprintf(pfile5, "Volumen de la Laguna : %5.3f m3 \n", Volumen);
fprintf(pfile5, "Superficie de la Laguna : %5.3f m2 \n", Superficie);
fprintf(pfile5, "Carga Organica : %5.3f KgDBO/Dia \n", CargaOrganica);
fprintf(pfile5, "Carga Superficial : %5.3f KgDBO/Ha-Dia \n", CargaSuperficial);
fprintf(pfile5, "Carga Hidraulica : %5.3f KgDBO/Dia \n", CargaHidraulica);
fprintf(pfile5, "Tiempo de Detencion : %5.3f Dias \n", TiempoDetencion);
fclose(pfile5);
}
CreaVentanaMensaje(0,2,"ESC", "Salir");
while(getch() != ESCAPE);

return(0);
}

/*-----*/
int LagunasAnaerobias(void)
/*-----*/
{
BorraBloque(0,0,20,79,atr2);
strcpy(temp, "Lagunas Anaerobias");
CreaVentanaMensaje(1,1,"C lculo de Lagunas Anaerobias.");
Ventana(0,0,9,79,atr2,1,temp);
DespliegaCadena("Demanda Bioquimica de Oxigeno Inicial (Kg/m3):",atr1,1,3);
DBOAfluente = RetomaDb(1,1,49,9);
if (NumeroCorrecto==2) return(-1);
DespliegaCadena("Demanda Bioquimica de Oxigeno del Efluente (Kg/m3):",atr1,2,3);
DBOEfluente = RetomaDb(1,2,55,9);
if (NumeroCorrecto==2) return(-1);
CoefProyecto = 6;
Exponente = 4.8;
TiempoDetencion = ((DBOAfluente / DBOEfluente) - 1) *
(1 / (CoefProyecto * pow((DBOEfluente / DBOAfluente), Exponente)));
Ventana(18,0,20,79,atr2,2,temp1);
Respuesta[0] = ' ';
while (Respuesta[0] != 'S' && Respuesta[0] != 'N')
{
DespliegaCadena("Conoce el Caudal (S/N)? : ",atr1,19,3);
Cursor(19,28);
Respuesta[0] = getche();
if (Respuesta[0]==27) return(-1);
}
if (Respuesta[0] == 'S')
{
DespliegaCadena("Caudal(m3/dia):",atr1,3,3);
Caudal = RetomaDb(1,3,18,9);
if (NumeroCorrecto==2) return(-1);
BorraBloque(18,0,20,79,atr2);
}
else
{
BorraBloque(19,3,19,70,atr2);
DespliegaCadena("Numero de Personas que aportan agua:",atr1,19,3);
NumeroPersonas = RetomaDb(1,19,41,9);
if (NumeroCorrecto==2) return(-1);
BorraBloque(19,3,19,70,atr2);
DespliegaCadena("Produccion de Agua Residual por persona (m3/dia):",atr1,19,3);
ProduccionResidual = RetomaDb(1,19,54,9);
if (NumeroCorrecto==2) return(-1);
BorraBloque(18,0,20,79,atr2);
Caudal = NumeroPersonas * ProduccionResidual;
sprintf(NumeroEnCadena, "Caudal(m3/dia): %5.3f", Caudal);
DespliegaCadena(NumeroEnCadena,atr1,3,3);
}
}

Volumen = TiempoDetencion * Caudal;

```

```

sprintf(NumeroEnCadena,"Volumen de la Laguna(m3): %5.3f",Volumen);
DespliegaCadena(NumeroEnCadena,attr4,4,3);
DespliegaCadena("Profundidad de la Laguna(m):",attr1,5,3);
ProfundidadLaguna = RetornaDb(1,5,32,9);
if (NumeroCorrecto==2) return(-1);
Superficie = Volumen / ProfundidadLaguna;
sprintf(NumeroEnCadena,"Superficie de la Laguna(m2): %5.3f",Superficie);
DespliegaCadena(NumeroEnCadena,attr4,6,3);
CargaHidraulica = ProfundidadLaguna / TiempoDetencion;
sprintf(NumeroEnCadena,"Carga Hidraulica(KgDBO/dia): %5.3f",CargaHidraulica);
DespliegaCadena(NumeroEnCadena,attr4,7,3);
sprintf(NumeroEnCadena,"Tiempo de Detención (dias): %5.3f",TiempoDetencion);
DespliegaCadena(NumeroEnCadena,attr4,8,3);

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
Ventana(21,0,23,79,attr2,1,temp);
DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
Cursor(22,25);
Imprimir[0] = getch();
if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
// impresion de encabezados en el impresor
if ((pfile5=fopen("LPT1","w"))==NULL)
{
BorraBloque(19,3,19,70,attr2);
DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
while(!kbhit());
return(0);
}
fprintf(pfile5,"\n\n***** Lagunas Anaerobias *****\n");
fprintf(pfile5,"Volumen de la Laguna : %5.3f m3 \n",Volumen);
fprintf(pfile5,"Superficie de la Laguna : %5.3f m2 \n",Superficie);
fprintf(pfile5,"Carga Hidraulica : %5.3f KgDBO/Dia \n",CargaHidraulica);
fprintf(pfile5,"Tiempo de Detencion : %5.3f dias \n",TiempoDetencion);
fclose(pfile5);
}
CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);

return(0);
}

/*-----*/
void Lagunas(void)
/*-----*/
{
/*— Definición temporal de atributos — */
attr3=WHITE(5<<4); /* Atributos del Mensaje abajo en la Pantalla */
attr2=WHITE(BLACK<<4); /* Atributos del Marco de Ventana Blanco/Negro */
attr1=WHITE(BLACK<<4); /* Atributos del Texto Desactivado Blanco/Negro */
attr0=WHITE(BLUE<<4); /* Atributos del Texto Activado Blanco/Azul */
BorraBloque(0,0,23,79,attr1);
/* —Este atributo es propio — */
attr4=WHITE(MAGENTA<<4); /* Atributos del Texto para respuestas Blanco/Cyan */
/*-----*/

Reconstruye = 1;
SalirOpcion_13 = 0;
while(!SalirOpcion_13)
{
if (Reconstruye)
{
BorraBloque(0,0,20,79,attr2);
SalirOpcion_13 = 0;
PosicionMenu = 131;
strcpy(temp,"Lagunas de Estabilizacion");

```

```

CreaVentanaMensaje(1,1,MENSAJE_131);
Ventana(8,21,13,56,attr2,2,temp);
DespliegaCadena(OPCION_131,attr0,9,22);
DespliegaCadena(OPCION_132,attr1,10,22);
DespliegaCadena(OPCION_133,attr1,11,22);
DespliegaCadena(OPCION_134,attr1,12,22);
strcpy(temp,"");
Cursor(9,22);
Reconstruye = 0;
}
switch(getch())
{
  case 72:
  {
    switch(PosicionMenu)
    {
      case 131:
      {
        DespliegaCadena(OPCION_131,attr1,9,22);
        CreaVentanaMensaje(1,1,MENSAJE_134);
        DespliegaCadena(OPCION_134,attr0,12,22);
        PosicionMenu = 134;
        break;
      }
      case 132:
      {
        DespliegaCadena(OPCION_132,attr1,10,22);
        CreaVentanaMensaje(1,1,MENSAJE_131);
        DespliegaCadena(OPCION_131,attr0,9,22);
        PosicionMenu = 131;
        break;
      }
      case 133:
      {
        DespliegaCadena(OPCION_133,attr1,11,22);
        CreaVentanaMensaje(1,1,MENSAJE_132);
        DespliegaCadena(OPCION_132,attr0,10,22);
        PosicionMenu = 132;
        break;
      }
      case 134:
      {
        DespliegaCadena(OPCION_134,attr1,12,22);
        CreaVentanaMensaje(1,1,MENSAJE_133);
        DespliegaCadena(OPCION_133,attr0,11,22);
        PosicionMenu = 133;
        break;
      }
    }
  }
  break;
}
case 80:
{
  switch(PosicionMenu)
  {
    case 131:
    {
      DespliegaCadena(OPCION_131,attr1,9,22);
      CreaVentanaMensaje(1,1,MENSAJE_132);
      DespliegaCadena(OPCION_132,attr0,10,22);
      PosicionMenu = 132;
      break;
    }
    case 132:
    {
      DespliegaCadena(OPCION_132,attr1,10,22);
      CreaVentanaMensaje(1,1,MENSAJE_133);
      DespliegaCadena(OPCION_133,attr0,11,22);
      PosicionMenu = 133;
      break;
    }
  }
}

```

```

    }
    case 133:
    {
        DespliegaCadena(OPCION_133,attr1,11,22);
        CreaVentanaMensaje(1,1,MENSAJE_134);
        DespliegaCadena(OPCION_134,attr0,12,22);
        PosicionMenu = 134;
        break;
    }
    case 134:
    {
        DespliegaCadena(OPCION_134,attr1,12,22);
        CreaVentanaMensaje(1,1,MENSAJE_131);
        DespliegaCadena(OPCION_131,attr0,9,22);
        PosicionMenu = 131;
        break;
    }
}
break;
}
case ENTER:
{
    switch(PosicionMenu)
    {
        case 131:
        {
            LagunasAerobias();
            BorraBloque(0,0,20,79,attr2);
            Reconstruye = 1;
            break;
        }
        case 132:
        {
            LagunasFacultativas();
            BorraBloque(0,0,20,79,attr2);
            Reconstruye = 1;
            break;
        }
        case 133:
        {
            LagunasAnaerobias();
            BorraBloque(0,0,20,79,attr2);
            Reconstruye = 1;
            break;
        }
        case 134:
        {
            AMecanica();
            break;
        }
    }
    BorraBloque(0,0,20,79,attr2);
    Reconstruye = 1;
    break;
}
case ESCAPE:
{
    BorraBloque(3,10,7,30,attr2);
    SalirOpcion_13 = 1;
}
}
}
BorraBloque(0,0,23,79,attr2);
}
}
void main(void)
{
    Lagunas();
}

```

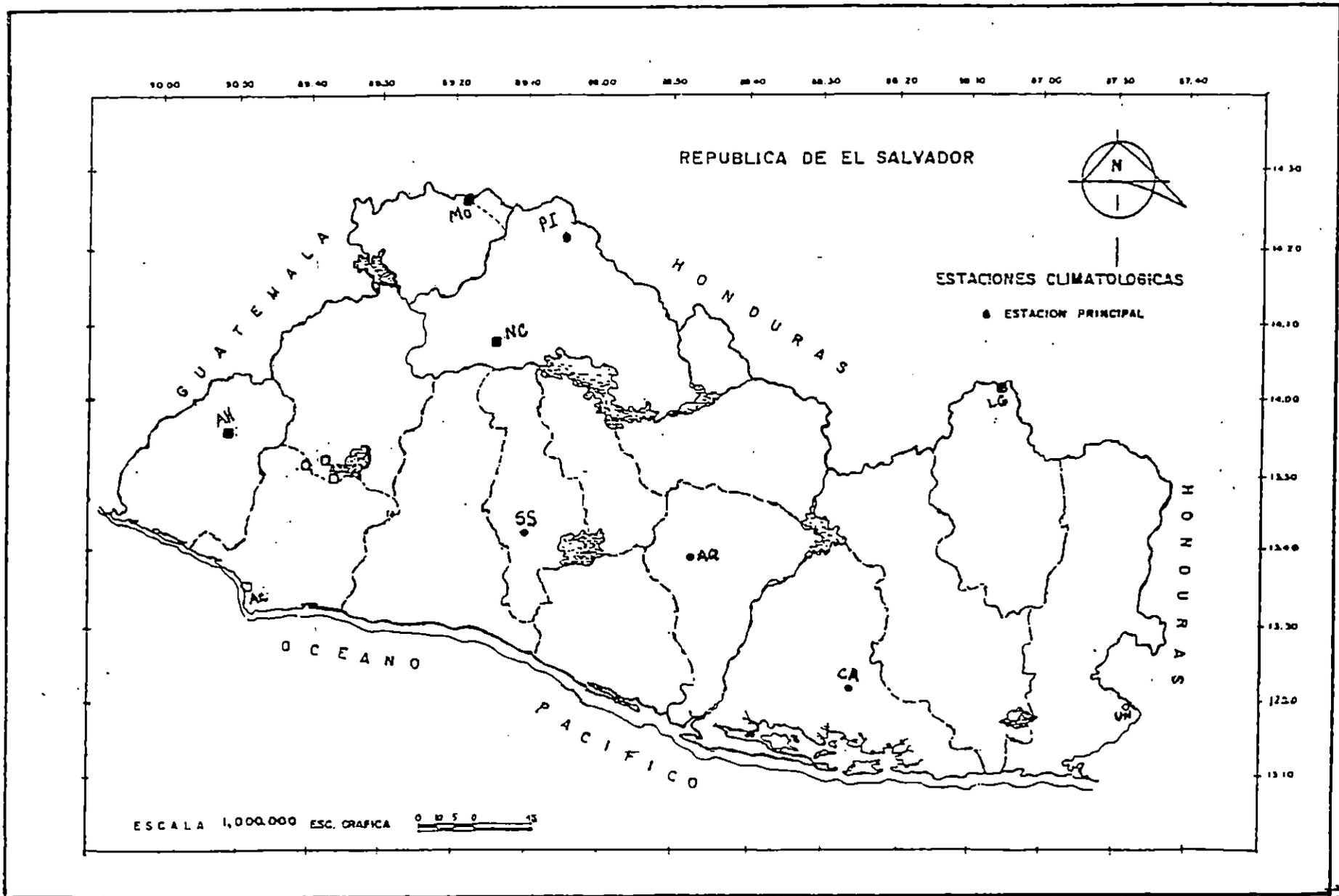
**ANEXOS PARA
LAGUNAS DE ESTABILIZACION**

ANEXO 1

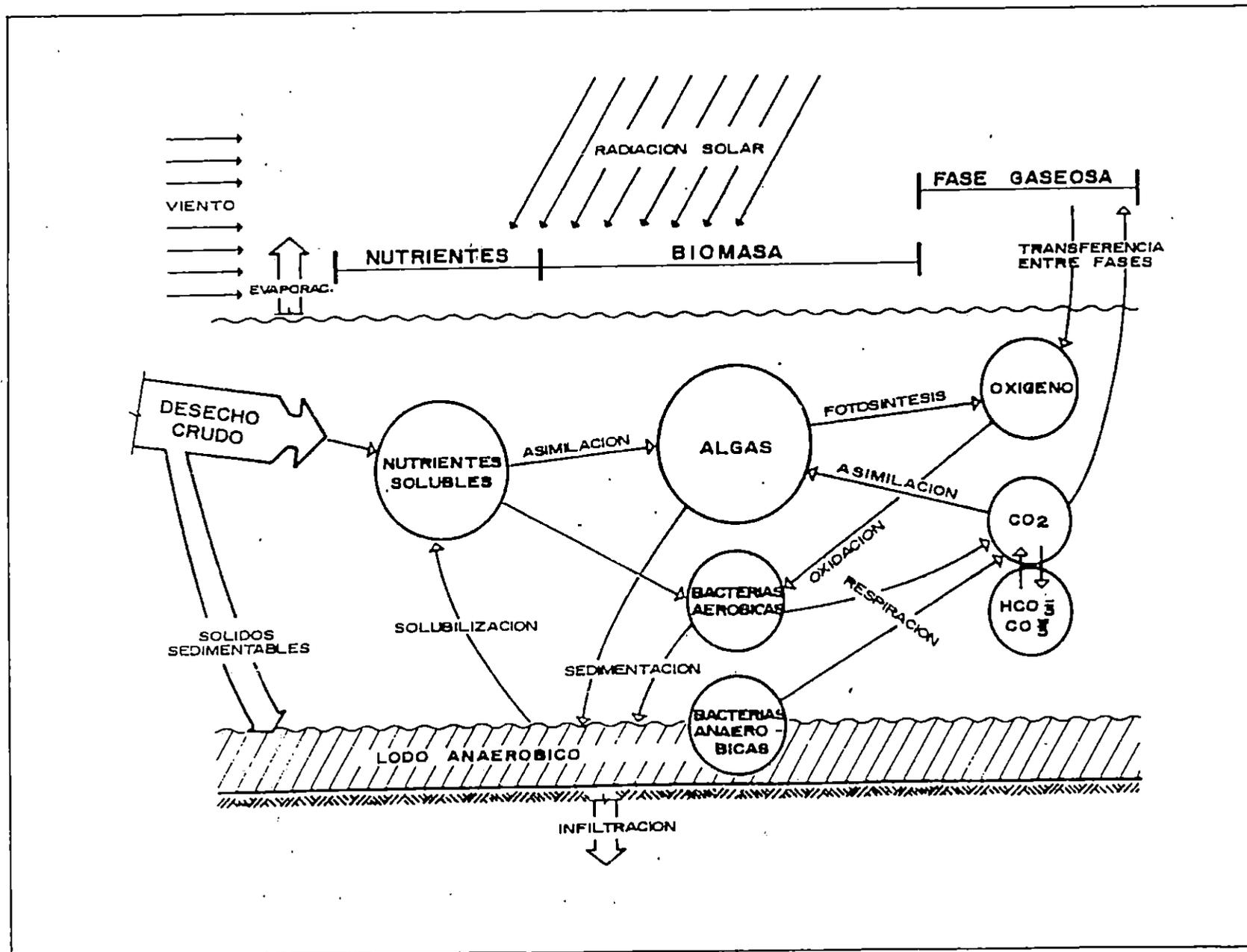
ESTACIONES CLIMATOLOGICAS

ANEXO 2

MODELO DINAMICO DE LAGUNAS FACULTATIVAS



ANEXO 1 ESTACIONES CLIMATOLÓGICAS (ALMANAQUE SALVADOREÑO, 1993)



ANEXO 2 MODELO DINAMICO DE LAGUNAS FACULTATIVAS (CEPIS, 1986)

CAPITULO III

ADSORCION EN CARBON ACTIVADO

3.1 GENERALIDADES

Los efluentes procedentes de un tratamiento secundario pueden encontrarse aún insuficientemente depurados, siendo por ello necesario efectuar una serie de procesos que liberen a los vertidos de sus contaminantes. Estos procesos constituyen los tratamientos terciarios, los cuales eliminan fundamentalmente la materia orgánica que aún persiste en el agua residual, los sólidos en suspensión generados por procesos biológicos y las sales inorgánicas disueltas.

Uno de los principales tratamientos terciarios es la adsorción, que se emplea para eliminar restos de materia orgánica que existen en disolución y algunos contaminantes específicos. En la actualidad, el adsorbente más utilizado es el carbón activado.

El carbón activado, cobró importancia por primera vez debido a su aplicación como adsorbente en las máscaras antigas usadas durante la Primera Guerra Mundial.

La primera aplicación comercial se realizó en 1794, cuando se emplearon filtros de carbón de leña en una refinería británica de azúcar (Austin, 1994).

En el pasado, también se ha utilizado para la eliminación de compuestos orgánicos causantes de sabor y olor en abastecimientos de agua potable (Metcalf, 1977).

El carbón activado se prepara haciendo, en primer lugar, carbón. Un material, como madera o carbón, se lleva al rojo vivo en una retorta para expulsar los hidrocarburos, pero con una aportación insuficiente de aire para mantener la combustión. La partícula de carbón se activa seguidamente exponiéndola a un gas oxidante a elevada temperatura. Este gas desarrolla una estructura porosa en el carbón natural y crea así una gran superficie interna. Las propiedades de superficie resultantes son función del material inicial usado y del procedimiento exacto de preparación.

Tras la activación, el carbón puede separarse, o prepararse, en diferentes tamaños, los dos más corrientes son granular y en polvo. El diámetro del carbón en polvo es generalmente inferior que la malla 200, mientras que el carbón granular tiene un diámetro superior a 0.1 mm (Metcalf, 1977).

Para la adsorción de gases y vapores, se emplea material granulado y para la purificación de líquidos es necesario material en polvo (Austin, 1994). El polvo se ha aplicado también con éxito a los tanques de almacenamiento y tanques de sedimentación. Ahí, actúa como adsorbente y como un medio para obstruir el paso de la luz, impidiendo así el crecimiento de algas (Fair, 1993).

El carbón activado es una forma de carbón vegetal. Entre la materia prima para la manufactura se puede mencionar: madera, estopas de coco, algas marinas, hulla, bagazo, huesos, etc.(Tabla 3.1). Estos materiales tienen aplicaciones especiales. El carbón activado difiere, dependiendo de la materia prima usada en los procesos de activación, ya que esta consiste en un cambio físico por el cual la superficie del carbón es tremendamente aumentada por la eliminación de hidrocarburos. Existen varios métodos idóneos para la activación. Esencialmente, todos los carbones activados que se producen en Estados Unidos están hechos por el método de activación con vapor. Muchos de los carbones activados usados en tratamiento de aguas son sustancias vendidas bajo nombres de marca como Darco, Minchar y Nuchar.

Algunos adsorbentes que pueden usarse en el tratamiento de aguas son:

Tierras de Fuller, son arcillas naturales; las variedades norteamericanas proceden generalmente de Florida y Georgia. Principalmente son silicatos de aluminio y magnesio, bajo la forma de atapulguita y montmorillonita. La arcilla se calienta y se seca, y durante esta operación desarrolla una estructura porosa, es molida y cernida. Los tamaños que se consiguen comercialmente van desde grandes gránulos hasta polvos finos. Las arcillas son particularmente útiles para decolorar, neutralizar y secar productos del petróleo como aceites lubricantes, aceites de transformador, kerosenos y gasolinas, lo mismo que aceites vegetales y animales. Lavando y quemando la materia orgánica adsorbida sobre la arcilla durante su utilización, el adsorbente puede utilizarse muchas veces (Treybal, 1988).

Carbones decolorantes, Se preparan de modos distintos. a) se mezcla materia vegetal con sustancias inorgánicas como cloruro de calcio; se carboniza y elimina por lixiviación la materia inorgánica. b) Mezclando materia orgánica, como aserrín, con sustancias porosas como piedra pómez; luego se emplea calentamiento y carbonización para depositar la materia carbónica en todas las partículas porosas. c) Se carboniza madera, aserrín y similares; luego se emplea activación con aire o vapor caliente. La lignita y el carbón bituminoso también son materias primas. Se utilizan para gran cantidad de fines, incluso para la decoloración de soluciones de azúcar, sustancias químicas industriales, drogas y líquidos de limpieza en seco, purificación de agua, refinamiento de aceites vegetales y animales y para la recuperación de oro y plata a partir de sus soluciones de cianuro (Treybal, 1988).

TABLA 3.1 MATERIA PRIMA QUE HA SIDO ESTUDIADA PARA LA PRODUCCIÓN DE CARBÓN ACTIVADO (REF. FAUST,1987).

Bagazo	Grafito
Estopas de coco	Desperdicios de cueros
Sangre	Melaza
Huesos	Lignina
Carbohidratos	Lignita
Cereales	Coque
Carbon	Vainas de arroz
Cáscaras de nuez	Aserrín
Olotes	Desperdicios de hule
Desperdicios de destileria	Madera

Algunas características del carbón activado son listadas en la tabla 3.2 y son muy importantes en el tratamiento de aguas. Ocasionalmente, el contenido de cenizas de carbón es un factor que afecta el tratamiento.

TABLA 3.2 CARACTERÍSTICAS DE CARBÓN ACTIVADO (FAUST,1987)

Densidad	Contenido de:
Distribución de partículas	Sulfuros
Porosidad	Sulfitos
Area superficial	Sulfatos
Tipo de filtración	Fosfatos
Polvosidad	Cloruros
Temperatura de ignición	Hierro
Conductividad eléctrica	Zinc
Retención de aceite	Calcio
Humedad	Magnesio
pH	Silicio
Total de cenizas	Cobre
Resistencia a la tracción	
Dureza	
Acidos inorgánicos solubles	

3.2 USO DE CARBON ACTIVADO Y APLICACIONES

El carbón activado es usado en la purificación del agua potable, principalmente para eliminación de olores y sabores que resultan de la presencia de gases disueltos y métodos orgánicos. Se ha encontrado que ha sido efectivo como un auxiliante de la coagulación, en la eliminación de color, en el control de algas, en la prevención o retardo de la descomposición de lodos y en la reducción de la demanda de cloro en el tratamiento de aguas. Ayuda en el acondicionamiento de agua a incrementar la turbidez y a producir una mejor sedimentación de microorganismos. Entre sus ventajas se deben de incluir las siguientes: ningún agente químico es agregado al agua; porque el carbón es filtrado, no es peligrosa una sobredosis de carbón activado; un exceso de cloración se hace más fácil. La Tabla 3.3 muestra análisis inorgánicos de carbones empleados en tratamiento de aguas.

**TABLA 3.3 CARBONES EMPLEADOS EN EL TRATAMIENTO DE AGUAS
(FAUST, 1987)**

Carbón	%de Carbón	% de Cenizas	% de solubilidad HCl	pH
Norit	93 - 96	4.5 - 6.5	2 - 3	7.8 - 8.3
Darco	65 - 70	25 - 30	0.5	4.5 - 6
Suchar	97 - 99	2 - 3.5	1.2 - 2.4	6.3 - 7
Nuchar W	95 - 98	2 - 3.5	1.3 - 2.5	7.5
Nuchar 2	90 - 95	5 - 10	3 - 7	7 - 8
Bone Char	9 - 10	90		
Cliffchar	89 - 90	3 - 4		9.5 - 10.5

En la remoción de olores y sabores de un agua por carbón activado, el material debe ser aplicado directamente como polvo seco al agua, o un agua filtrada deberá pasar a través de un lecho de carbón activado. Si se usa polvo seco deberá ser aplicado al agua, antes de la filtración ya sea sólo o con otros químicos como es el alumbre.

En general, el carbón permite la eliminación de impurezas orgánicas en solución con la reducción de DBO y DQO. Permite también la eliminación de compuestos tóxicos no biodegradables, durante el tratamiento terciario. Recientemente se ha incorporado con éxito el carbón activado al proceso de lodos activos.

Se indica a continuación compuestos orgánicos como inorgánicos que pueden ser removidos mediante carbón activado:

Compuestos orgánicos :

- Hidrocarburos halogenados (Tetracloruro de Carbono, Cloroformo, etc.)
- Acidos húmicos y fúlvicos (colores)
- Compuestos fenólicos (Fenol, Nitrofenoles, Clorofenoles, etc.)
- Pesticidas orgánicos (DDT, Lindano, Fenilaminas, etc.)
- Olores y sabores (Mercaptanos y otros sulfuros orgánicos, olores de algas, etc.)
- Detergentes

Compuestos inorgánicos :

- Efluentes industriales con Cadmio (plateados), Zinc (galvanizados), Cromo (cromados), etc.
- Cloro y derivados en aguas industriales de industrias alimenticias.
- Efluentes radiactivos
- Acido sulfhídrico

La reducción de los sabores y olores en los abastecimientos de agua por adsorción de las sustancias ofensivas sobre carbón activado es probablemente la aplicación directa más importante de la adsorción en tratamiento de aguas. Las columnas o lechos de carbón activado granular se emplean, 1) para concentrar los contaminantes orgánicos del agua para propósitos de su análisis, ó 2) para la remoción de los contaminantes. Algo de la remoción de las sustancias productoras de color y de otros contaminates durante la coagulación pueden ser el resultado de la adsorción. De hecho, algunos procesos utilizan este fenómeno para reducir la sílice, los fluoruros o sustancias radiactivas.

Los siguientes son algunos puntos que deben considerarse en la alimentación de carbón activado : (1) el carbón debe ser manejado en un cuarto separado, para minimizar las molestias de polvo, (2) la máquina de alimentación debe tener una capacidad considerable debido a las fluctuaciones de carga que vienen de la mayoría de plantas, (3) un abastecimiento abundante de agua se debe proveer, para llevar el carbón al punto de aplicación sin peligro de obstruir la tubería de flujo, (4) la fuerza de flujo donde el flujo de alimentación es usado, debe ser acerca del 4 por ciento, (5) el agitador debe situarse en el tanque para mantener una suspensión uniforme, (6) si un orificio es usado, debe ser resistente a la corrosión, (7) un ejecutor de agua debe ser usado para llevar carbón fuera de la máquina, (8) el equipo eléctrico para pruebas de polvo debe usarse en los alrededores, tuberías de hule, plásticos o acero deben usarse para el transporte de flujo (Babbit, 1962).

3.3 ADSORCION

La adsorción es un fenómeno físico que depende mucho del área superficial y del volumen de los poros. La estructura del poro limita el tamaño de las moléculas que pueden ser adsorbidas, y las áreas superficiales desarrolladas limitan la cantidad de material

que puede ser adsorbido si se supone que tiene un tamaño molecular adecuado (Austin 1994).

La adsorción puede describirse como el proceso en el que las moléculas abandonan la solución y quedan retenidas en la superficie sólida mediante enlaces físicos y químicos (Metcalf, 1977).

Las moléculas adsorbidas reciben el nombre de adsorbato y adsorbente es el material que retiene al adsorbato. Si los enlaces que se forman entre el adsorbato y el adsorbente son muy fuertes, el proceso es casi siempre irreversible y se dice que ha tenido lugar una adsorción o quimio-adsorción. Por otro lado, si los enlaces que se forman fueran débiles (formados por fuerzas de Van der Waals) se dice que ha tenido lugar una adsorción física. Las moléculas así adsorbidas se eliminan fácilmente, o son desorbidas por un cambio de concentración en la solución por lo que el proceso es reversible. La adsorción física es el proceso que ocurre más frecuentemente en la eliminación de los constituyentes de agua residual por carbón activado (Metcalf, 1977).

La cantidad de adsorbato que puede retener un adsorbente es función no solo de la concentración (C) del adsorbato sino también de la temperatura (T) (Metcalf, 1977).

Cuando el adsorbente es un sólido poroso, como el carbón granular o cuando la concentración del adsorbato es pequeña, el contacto total puede estar limitado por procesos difusionales u otros procesos de transporte. Entonces, el alcance del equilibrio es demorado (Fair, 1987).

Las posiciones de equilibrio en adsorción están dadas por las isothermas de adsorción, que relacionan la cantidad adsorbida por unidad de adsorbente con la concentración de adsorbato a temperatura constante. Generalmente se usan dos formulaciones matemáticas de estas isothermas para la adsorción de soluciones acuosas y fueron desarrolladas por Freundlich y Langmuir a partir de datos experimentales de la isoterma (Fair, 1993).

Ecuación de la Isoterma de Freundlich.

Derivada a partir de consideraciones empíricas, la isoterma de Freundlich es:

$$\frac{X}{M} = kC^{1/n} \quad \text{Ec.3.1}$$

Donde :

$\frac{X}{M}$: cantidad adsorbida por peso unitario de adsorbente

C : concentración de equilibrio del adsorbente en solución después de la adsorción

k , $1/n$: constantes empíricas

Las constantes pueden calcularse dibujando $\frac{X}{M}$ en relación a C en papel logarítmico.

En general, los valores de n son mayores que la unidad, sugiriendo que la adsorción es relativamente más eficiente a bajas concentraciones.

La forma logarítmica de la ecuación de Freundlich

$$\log \left(\frac{X}{M} \right) = \log K + \left(\frac{1}{n} \right) \log C \quad \text{Ec. 3.2}$$

indica una variación lineal de $\log \left(\frac{X}{M} \right)$ con $\log C$. Por tanto, se pueden emplear trazados logarítmicos de los datos de adsorción para correlación e interpolación, así como para la evaluación de k y n .

Con frecuencia, la ecuación de Freundlich es útil cuando se ignora la identidad real del soluto, como en la adsorción de sustancias coloreadas en materiales como soluciones de azúcar, aceites minerales y aceites vegetales. En estos casos, la concentración de soluto puede medirse mediante un colorímetro o un espectrofotómetro y expresarse en función de unidades arbitrarias de intensidad de color, siempre y cuando la escala de color utilizada varíe linealmente con las concentraciones de soluto que causa las variaciones de color (Treybal, 1988).

Ecuación de la Isotherma de Langmuir

Se basa en las suposiciones de una capa monomolecular de adsorbente, uniformidad de la superficie del adsorbente, y la no interacción entre las moléculas del adsorbato. Esta ecuación tiene la forma:

$$\left(\frac{X}{M} \right) = \frac{abC}{1+bC} \quad \text{Ec. 3.3}$$

Donde :

$\frac{X}{M}$: cantidad adsorbida por peso unitario de adsorbente

a, b : constantes empíricas

C : concentración de equilibrio del adsorbato en solución después de la adsorción.

Las constantes a y b , aún cuando tienen un significado teórico dentro de los límites de las suposiciones, generalmente se determinan en forma empírica. La ecuación predice una variación lineal de $\frac{X}{M}$ con C a concentraciones bajas cuando $bC < 1$ y una capacidad adsorptiva limitante, $\frac{X}{M} = b$, cuando $bC > 1$ a concentraciones elevadas.

Esta fórmula se desarrolló sobre las dos siguientes suposiciones: 1) que existe un número fijo de puntos accesibles disponible en la superficie adsorbente, todos los cuales poseen la misma energía, y 2) que la adsorción es reversible. El equilibrio se alcanza cuando la velocidad de adsorción de las moléculas sobre la superficie sea igual a la velocidad de desorción de las moléculas. Así pues, la velocidad a la que prosigue la adsorción es proporcional a la fuerza que la produce que es la diferencia entre la cantidad adsorbida para una concentración dada y la cantidad que pueda ser adsorbida a dicha concentración. A la concentración de equilibrio, esta diferencia es cero (Metcalf, 1977).

La figura 3.1 muestra la isoterma de Langmuir. Las constantes de la ecuación de Langmuir pueden determinarse dibujando $C/(X/M)$ frente a C y utilizando la ecuación 3.3, expresada ahora como:

$$\frac{C}{\left(\frac{X}{M}\right)} = \frac{1}{ab} + \frac{1}{a} C \quad \text{Ec.3.4}$$

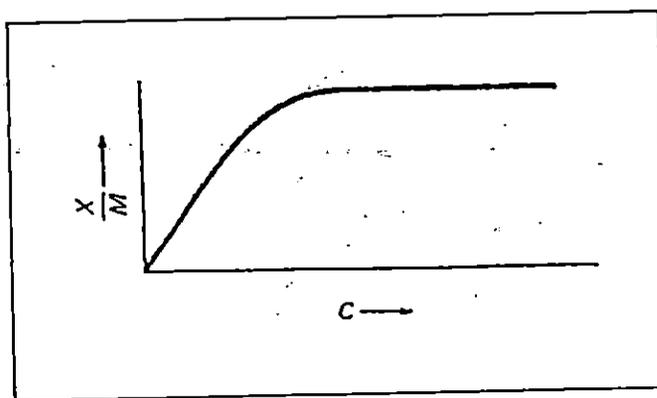


FIGURA 3.1 ESQUEMA DE LA FORMA GENERALIZADA DE LA ISOTERMA DE LANGMUIR (METCALF, 1977).

El empleo de carbón activado se puede ver también en la purificación de soluciones, como las aplicadas en la limpieza de azúcar de caña, de la remolacha y del azúcar de maíz, así como para la eliminación de los sabores y olores de las aguas para abastecimiento, de las grasas vegetales, animales y de los aceites y bebidas alcohólicas, productos químicos y farmacéuticos (Fair ,1993).

3.3.1 ADSORCION DE COLORES Y SABORES

Para describir la adsorción de los olores y sabores del agua , las ecuaciones de Freundlich y de Langmuir se transforman en las siguientes formas:

$$\frac{(C_0 - C)}{m} = kC^{(1/n)} \quad \text{Ec. 3.5}$$

$$\text{Log} \left[\frac{(C_0 - C)}{m} \right] = \text{Log}k + \left(\frac{1}{n} \right) \text{Log}C \quad \text{Ec. 3.6}$$

$$\frac{(C_0 - C)}{m} = \frac{(abC)}{(1 + bC)} \quad \text{Ec. 3.7}$$

$$\frac{1}{\left[\frac{(C_0 - C)}{m} \right]} = \frac{1}{b} + \left(\frac{1}{ab} \right) \left(\frac{1}{C} \right) \quad \text{Ec. 3.8}$$

Aquí, C_0 es el olor o sabor inicial en el agua que se va a tratar, C es el olor ó sabor, m es la concentración del adsorbente, y K , n , a , y b son coeficientes, siendo la capacidad adsortiva limitante del adsorbente.

3.4 PRINCIPIOS DEL PROCESO DE ADSORCION

El tratamiento del agua residual con carbón activo suele considerarse como un proceso de acabado del agua ya tratada por procesos de tratamiento biológico normales. En este caso, el carbón se utiliza para eliminar parte de la materia orgánica residual disuelta.

Según los medios con que el carbón se ponga en contacto con el agua, la materia particulada presente también puede ser eliminada.

Es muy probable que la operación en lecho fijo o a contracorriente se convierta en el medio efectivo y eficiente para utilizar el carbón y otros adsorbentes.

Con frecuencia se utiliza una columna como medio de contacto del agua residual con el carbón granular. Una columna típica, utilizanda en una planta piloto en Pomona, California, se representa en la figura 3.2. El agua se introduce por la parte superior de la columna y sale por la inferior. El carbón se mantiene en su lugar por medio de una rejilla en el fondo de la columna. Es necesario prever las instalaciones de lavado en superficie y a contracorriente media de 10 mg/l y es eliminada casi totalmente por el filtro. El lavado a contracorriente es necesario para evitar las pérdidas excesivas de carga. Tales columnas pueden operar solas, en serie, o en paralelo (Metcalf, 1977).

El problema de la obstrucción del lecho de carbón puede superarse parcialmente si se utiliza sin lecho expandido. Durante la operación, el afluente se introduce por el fondo de la columna y se deja que se expanda el lecho, tanto como pueda expandirse un lecho filtrante durante un lavado a contracorriente. El carbón consumido se va sustituyendo en forma continua por nuevo carbón. En un sistema tal, la pérdida de carga no aumenta con el tiempo, una vez que se ha alcanzado el punto de funcionamiento.

Otro medio de aplicación que se estudia en la actualidad es el de añadir carbón activo en polvo al afluente en tratamiento biológico. El carbón, en este caso, es añadido al afluente en un tanque de contacto. Una vez transcurrido el tiempo de contacto deseado se deja que el carbón se sedimente en el fondo del tanque y seguidamente se extrae del mismo el agua tratada. Puesto que el carbón es muy fino, puede necesitarse un coagulante.

La aplicación económica del carbón depende de la existencia de un medio eficaz para regenerarlo una vez que se haya alcanzado su capacidad de adsorción. El carbón granular se regenera fácilmente por oxidación de la materia orgánica y su posterior eliminación de la superficie del carbón en un horno. Parte del carbón (aproximadamente un 10%) se destruye en la regeneración y debe sustituirse con carbón nuevo o virgen.

Es preciso aclarar que la capacidad del carbón regenerado es ligeramente inferior a la del carbón virgen. El uso del carbón en polvo en el tratamiento de aguas residuales se ha visto algo limitado, por la falta de un medio eficaz de recogida y regeneración del carbón consumido.

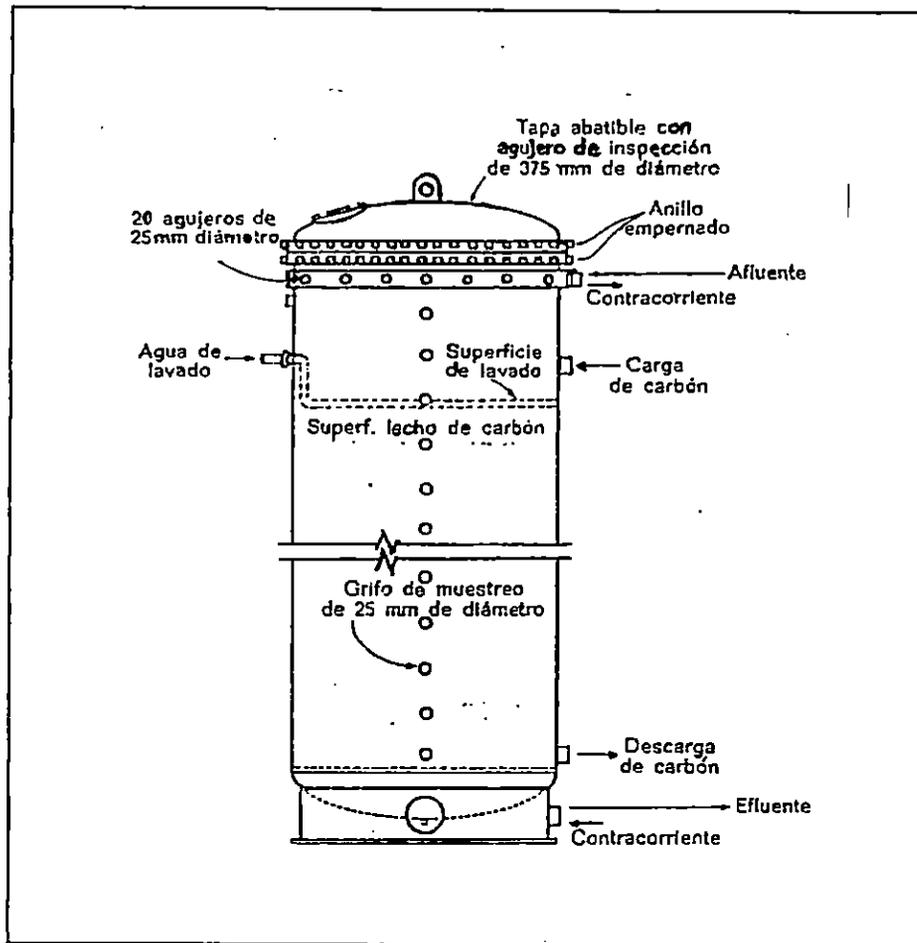


FIGURA 3.2 COLUMNA TÍPICA CON CARBÓN ACTIVADO (METCALF, 1977)

3.4.1 REGENERACION DEL CARBON ACTIVADO

La adsorción es normalmente un proceso reversible, lo que significa bajo condiciones de operación apropiadas, es posible retirar el material acumulado en el lecho y reutilizar el carbón.

En muchas ocasiones de tratamiento de agua residual con carbón activado, los contaminantes están en muy baja concentración (además suelen tener elevadas masas moleculares) y la velocidad de uso del carbón es, por tanto, también muy baja.

Para mayores sistemas de tratamiento de aguas residuales o en sistemas cuyo objetivo es la recuperación de un producto, la etapa de regeneración es una parte integral del proceso. El procedimiento más habitual de regeneración es la reactivación térmica en hornos tipos rotatorios o múltiple que operan a temperaturas entre 870 y 980°C y con tiempos de residencia para el carbón de unos 30 minutos. Además de la quema del

adsorbato siempre se quema algo de carbón, con pérdidas habituales entre un 10 y un 15%. En general, las condiciones de regeneración suelen establecerse empíricamente.

3.5 EJEMPLOS DE CALCULO DE ADSORCION EN CARBON ACTIVADO

UTILIZACION DE LA ECUACION DE FREUNDLICH (CASO 1)

Una solución acuosa que contiene un soluto valioso está coloreada con pequeñas cantidades de una impureza. Antes de la cristalización, se va a eliminar la impureza por adsorción sobre un carbón decolorante que sólo adsorbe cantidades insignificantes del soluto principal. Mediante una serie de pruebas de laboratorio se agitaron distintas cantidades del adsorbente en lotes de la solución original, hasta que se estableció el equilibrio; se obtuvieron los siguientes datos de temperatura constante:

kg carbón/ kg soluc.	0	0.001	0.004	0.008	0.02	0.04
Color en el equilibrio	9.6	8.1	6.3	4.3	1.7	0.7

La intensidad de color se midió de acuerdo con una escala arbitraria, proporcional a la concentración de la sustancia coloreada. Se desea reducir el color al 10% de su valor original, 9.6. Calcular la cantidad de carbón fresco que se requiere por 1000 kg de solución.

Solución :

Para empezar, se deben convertir los datos experimentales a una forma adecuada a fin de graficar la isoterma en el equilibrio. Con este propósito, definir X/M en unidades de color por kilogramo de solución y C como unidades de color adsorbido por kilogramo de carbón.

M Kg carbón/kg soluc.	X/M Color en el equilibrio, unidades/kg soluc.	C Concen. del adsorbato, unidades/kg carbón
0	9.6	$(C_o - C_{eq}) / M$
0.001	8.6	$(9.6-8.6)/0.001=1000$
0.04	6.3	$(9.6-6.3)/0.004=825$
0.008	4.3	$(9.6-4.3)/ 0.008=663$
0.02	1.7	$(9.6-1.7)/0.02=395$
0.04	0.7	$(9.6-0.7)/0.04=223$

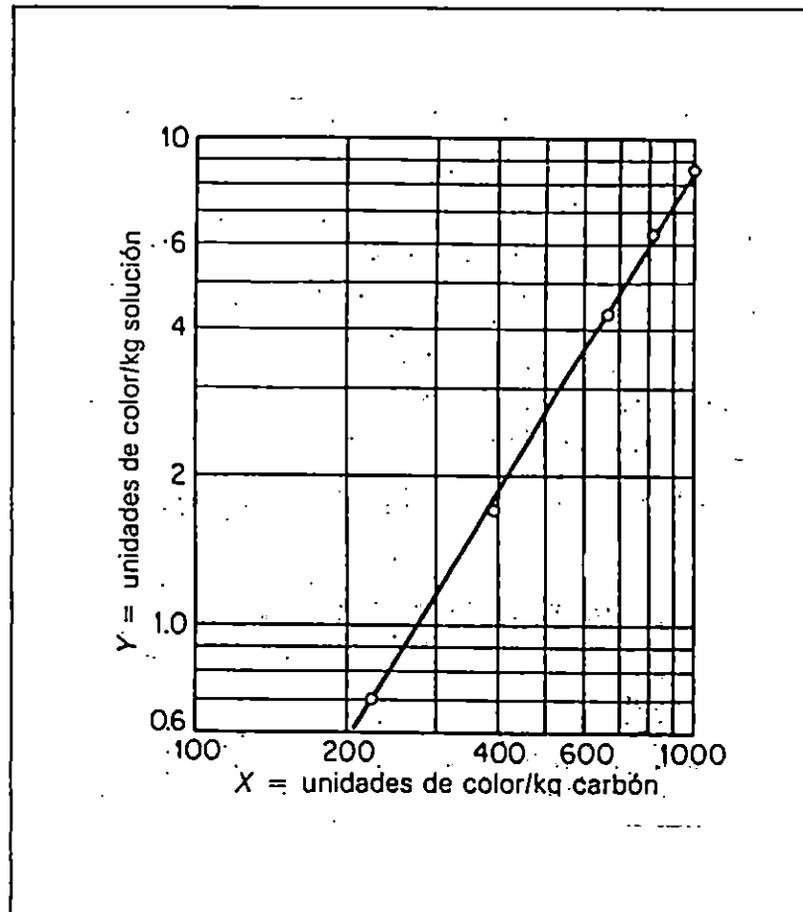


FIGURA 3.3 CÁLCULO DE CONSTANTES EMPÍRICAS PARA LA ECUACION DE FREUNDLICH (TREYBAL, 1988)

Los datos en el equilibrio, cuando se grafican sobre coordenadas logarítmicas, dan una línea recta, de forma que se aplica la ecuación de Freundlich (Figura 3.3). La pendiente de la línea es $1/n = 1.66$ y , para encontrar K que es el intercepto se despeja de la ecuación 3.1; quedando de la forma :

$$K = \frac{(X / M)}{C}$$

Ahora, sustituyendo datos para una concentración $C=663$ y $\left(\frac{X}{M}\right) = 4.3$:

$$K = \frac{4.3}{6.63^{1.66}}$$

$$K = 8.91 \times 10^{-5}$$

Teniendo las constantes empíricas $1/n$ y K se sustituyen en la ecuación 3.1

$$\frac{X}{M} = KC^{1/n}$$

Por lo tanto, la ecuación de Freundlich es

$$X/M = 8.91 \times 10^{-5} C^{1.66}$$

Los datos en el equilibrio también pueden graficarse sobre coordenadas aritméticas, dando como resultado una curva de equilibrio, que junto con la ecuación de Freundlich se usan para determinar cantidades de carbón que se requieren para cierta cantidad de solución a tratar en forma continua o por lotes.

APLICACION DE LA ECUACION DE FREUNDLICH (CASO 2)

Describanse las fases de realización de un estudio para desarrollar la isoterma de adsorción de Freundlich para la eliminación de un contaminante dado (DBO,color, o fenol) con carbón activo en polvo.

En los ensayos preliminares se usan dosis de carbón de 0, 5, 10, 20, y 50 mg / m³.

La siguiente tabla presenta los datos de un ensayo isotérmico de eliminación del color.

M Peso del Carbón, en g/100 ml de sln	C Color residual de la solución	X Color Adsorbido	X/M Color adsorbido por peso unitario
0	7.70	-	-
0.05	3.67	4.03	80.6
0.1	2.20	5.50	55.0
0.3	0.87	6.83	22.8
1.0	0.25	7.45	7.5

Solución

Con los datos anteriores se dibuja la isoterma, resultando la siguiente gráfica (X/M vrs C), Fig.3.4:

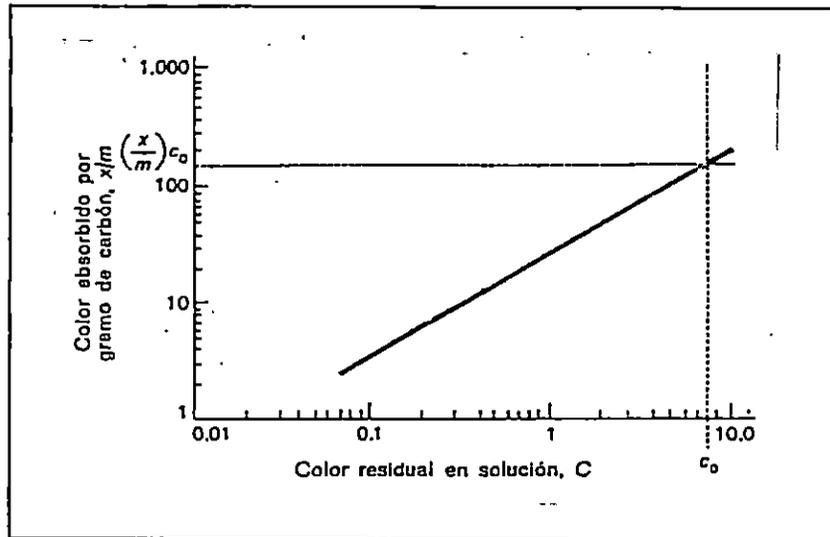


FIGURA 3.4 CÁLCULO DE CONSTANTES EMPÍRICAS (METCALF, 1977)

La pendiente $1/n$ se calcula así :

$$\frac{1}{n} = \frac{\log 80.5 - \log 22.8}{\log 3.67 - \log 0.87}$$

$$\frac{1}{n} = 0.876$$

Despejando de la ecuación 3.1 el intercepto k y sustituyendo datos para una concentración $C = 2.20$ y $\left(\frac{X}{M}\right) = 55$, se tiene :

$$K = \frac{(X/M)}{C^{1/n}}$$

$$K = \frac{55}{2.20^{0.876}}$$

$$K = 27.568$$

APLICACION DE LA ECUACION DE LANGMUIR PARA LA ADSORCION EN CARBON ACTIVADO

Determinése si los datos de la adsorción sobre carbón activo presentados a continuación siguen la isoterma de adsorción de Langmuir y, si así fué, determinése las constantes a y b

C, en mg/l	10	20	30
X/M, g/g	0,133	0,187	0,220

Solución

Se dibuja $C/(X/M)$ con relación a C (véase figura 3.5)

C, en mg/l	10	20	30
$C/(X/M)$	75	107	136

Como los datos aparecen en línea recta en la gráfica, se llega a la conclusión de que pueden describirse con la isoterma de adsorción de Langmuir

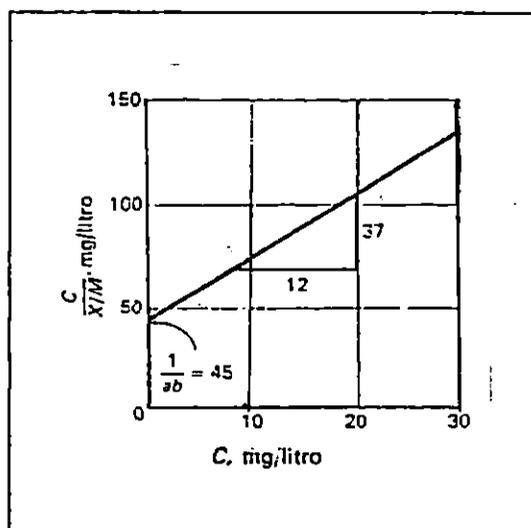


FIGURA 3.5 GRÁFICA DE LOS DATOS DE ADSORCIÓN DE LANGMUIR (METCALF, 1977)

La constante a se determina según la figura 3.5 de la siguiente manera:

$$\frac{1}{a} = \text{pendiente}$$

$$\frac{1}{a} = \frac{37}{12} = 3,08$$

Por tanto,

$$a = \frac{1}{3,08} = 0,325$$

La constante b se determina así:

$$\frac{1}{ab} = 45$$

Por tanto:

$$b = \frac{1}{45(0,325)} = 0,0685$$

Las constantes son $a = 0.325$ y $b = 0.0685$

3.6 PLANTEAMIENTO DE LO QUE APORTARA LA CODIFICACION DEL CASO

La adsorción es un tratamiento terciario que elimina restos de materia orgánica en disolución, algunos contaminantes químicos, y otros.

El adsorbente más utilizado es el carbón activado que puede ser empleado para la eliminación de los sabores y olores de las aguas, etc.

Para la adsorción de contaminantes en soluciones acuosas se considerarán dos modelos matemáticos a partir de datos experimentales, con los cuales se formarán las isothermas para la adsorción.

Ecuación de la Isotherma de Freundlich

Se realizan ensayos de adsorción en una columna de flujo continuo a fin de obtener datos para el diseño de la unidad.

El objetivo del uso de esta ecuación es encontrar las constantes empíricas (K y n); luego, esta ecuación de Freundlich servirá para encontrar la cantidad de carbón que utilizará una unidad en la operación por etapas.

Ecuación de la Isoterma de Langmuir

El uso de este modelo matemático, permitirá determinar las constantes (a, b) en forma empírica.

Esta ecuación se basa en la uniformidad de la superficie del adsorbente.

3.6.1 DEFINICION DE VARIABLES

a	= constante empírica
A_i	= $(\log C)^2$
b	= constante empírica
B_i	= $(\log C) (\log X / M)$
C_0	= concentración inicial de equilibrio (unidades / Kg carbón)
C	= concentración de equilibrio del adsorbente en solución después de la adsorción (unidades/ Kg carbón)
D	= determinante de la matriz
$D_{\log K}$	= determinante $\log K$
$D_{1/ab}$	= determinante $1/ab$
$D_{1/n}$	= determinante $1/n$
K	= constante empírica
M	= cantidad de adsorbente (kg carbón/Kg sln.) ó (Kg / ml sln.)
n	= constante empírica
N	= número de datos
S_i	= $C / (X / M)$
T_i	= $S \cdot C$
U	= C^2
X	= unidades de equilibrio
X_i	= $\log C$

$$X^* = C_0 - C_i$$

$$\frac{X}{M} = \text{cantidad adsorbida por peso unitario de adsorbente (unidades / Kg sln.)}$$

$$Y = \log X / M$$

3.7 SOLUCION DE MODELOS

3.7.1 ECUACION DE LA ISOTERMA DE FREUNDLICH

Los datos para dibujar isotermas se consiguen tratando volúmenes fijos de agua conteniendo el contaminante, con una serie de dosis de carbón conocidas (M).

La isoterma de adsorción relaciona la cantidad adsorbida por unidad de adsorbente con la concentración de adsorbato, (X/M) vrs C.

La ecuación de Freundlich es la Ec. 3.1:

$$\left(\frac{X}{M}\right) = KC^{1/n}$$

Haciendo trazos logarítmicos para una correlación se evaluará esta constante empírica (n) así como (K).

Para elaborar la isoterma, en el caso que solo se den valores de (X/M) además de las dosis de carbón (M), se encontrarán los valores de C de la siguiente manera:

$$C = \frac{C_0 - \left(\frac{X}{M}\right)}{M} \quad \text{Ec.3.9}$$

donde C_0 es la concentración inicial, que será un dato aportado y (X/M) serán datos también que se aportarán.

En el caso que solo se den valores de (M) y C, que es lo que sucede en un ensayo de laboratorio ; se tendrá que encontrar los valores de (X/M) de la siguiente manera:

$$X = C_0 - C \quad \text{Ec. 3.10}$$

La concentración inicial C_0 es un dato que se conocerá, así como los valores de C .

Habiendo obtenido los valores de X se encontrarán los valores de (X/M) dividiendo las X entre las M ya proporcionadas.

Una vez se tienen los valores de (X/M) y C , se manejarán éstos datos para el cálculo de la ecuación de regresión que en éste caso es una potencial y luego se encontrarán los valores de las constantes empíricas K y n .

Aplicando logaritmos a la ecuación de Freundlich se tiene la Ec. 3.2:

$$\log \left(\frac{X}{M} \right) = \log K + \left(\frac{1}{n} \right) \log C$$

Al derivar con respecto a K y a $1/n$ e igualar a cero las derivadas, se obtienen las ecuaciones normales siguientes:

$$\sum \log \left(\frac{X}{M} \right) = N \log K + \left(\frac{1}{n} \right) \sum \log C \quad \text{Ec.3.11}$$

$$\sum \log C \log \left(\frac{X}{M} \right) = \log K \sum \log C + \left(\frac{1}{n} \right) \sum (\log C)^2 \quad \text{Ec.3.12}$$

Ahora, se sustituyen los valores en las ecuaciones normales, para luego poder encontrar los valores de las constantes empíricas K y $1/n$. Se aplica la regla de Cramer para una matriz de coeficientes (2×2).

Los Diagramas 3.A, 3.B, 3.C ; son los flujogramas para la resolución de la ecuación de Freundlich.

3.7.2 ECUACION DE LA ISOTERMA DE LANGMUIR

La isoterma de adsorción relaciona la cantidad adsorbida por unidad de adsorbente (X/M) y la concentración del adsorbato C .

La ecuación de la isoterma de Langmuir es la Ec. 3.3:

$$\left(\frac{X}{M} \right) = \frac{abC}{1+bC}$$

Los datos de C y (X/M) deben ser conocidos, y para determinar las constantes empíricas a, b se expresa la Ec. 3.3 como:

$$\frac{C}{\left(\frac{X}{M}\right)} = \frac{1}{ab} + \frac{1}{a}C \quad \text{Ec.3.4}$$

donde $\frac{1}{ab}$ será el intercepto y $\frac{1}{a}$ será la pendiente. Para calcular la ecuación de regresión se utilizará la de la línea recta y por ser matemáticamente más exacto se utilizará el método de mínimos cuadrados.

Las ecuaciones normales para encontrar a y b son las siguiente:

$$\sum \frac{C}{\left(\frac{X}{M}\right)} = N\left(\frac{1}{ab}\right) + \left|\frac{1}{a}\right| \sum C \quad \text{Ec.3.13}$$

$$\sum \left[\frac{C}{\left(\frac{X}{M}\right)} \right] C = \frac{1}{ab} \sum C + \frac{1}{a} \sum C^2 \quad \text{Ec.3.14}$$

Al sustituir valores, nos encontramos con dos ecuaciones que se resolverán por el método de Cramer, dando una matriz (2 x 2) y se encontrará:

$$\frac{1}{ab} = ? \quad \frac{1}{a} = ?$$

Despejando a y sustituyéndola se encontrará b .

Los Diagramas 3.D y 3.E muestran los flujogramas para la resolución de la ecuación de Langmuir.

DIAGRAMAS DE FLUJO
PARA
ADSORCION EN CARBON ACTIVADO

- DIAGRAMA 3.A DETERMINACION DE CONSTANTES EMPIRICAS EN LA ECUACION DE FREUNDLICH (CASO 1)
- DIAGRAMA 3.B DETERMINACION DE CONSTANTES EMPIRICAS EN LA ECUACION DE FREUNDLICH (CASO 2)
- DIAGRAMA 3.C RESOLUCION MATRICIAL POR EL METODO DE CRAMER (FREUNDLICH)
- DIAGRAMA 3.D DETERMINACION DE LAS CONSTANTES EMPIRICAS EN LA ECUACION DE LANGMUIR
- DIAGRAMA 3.E RESOLUCION MATRICIAL POR EL METODO DE CRAMER (LANGMUIR)

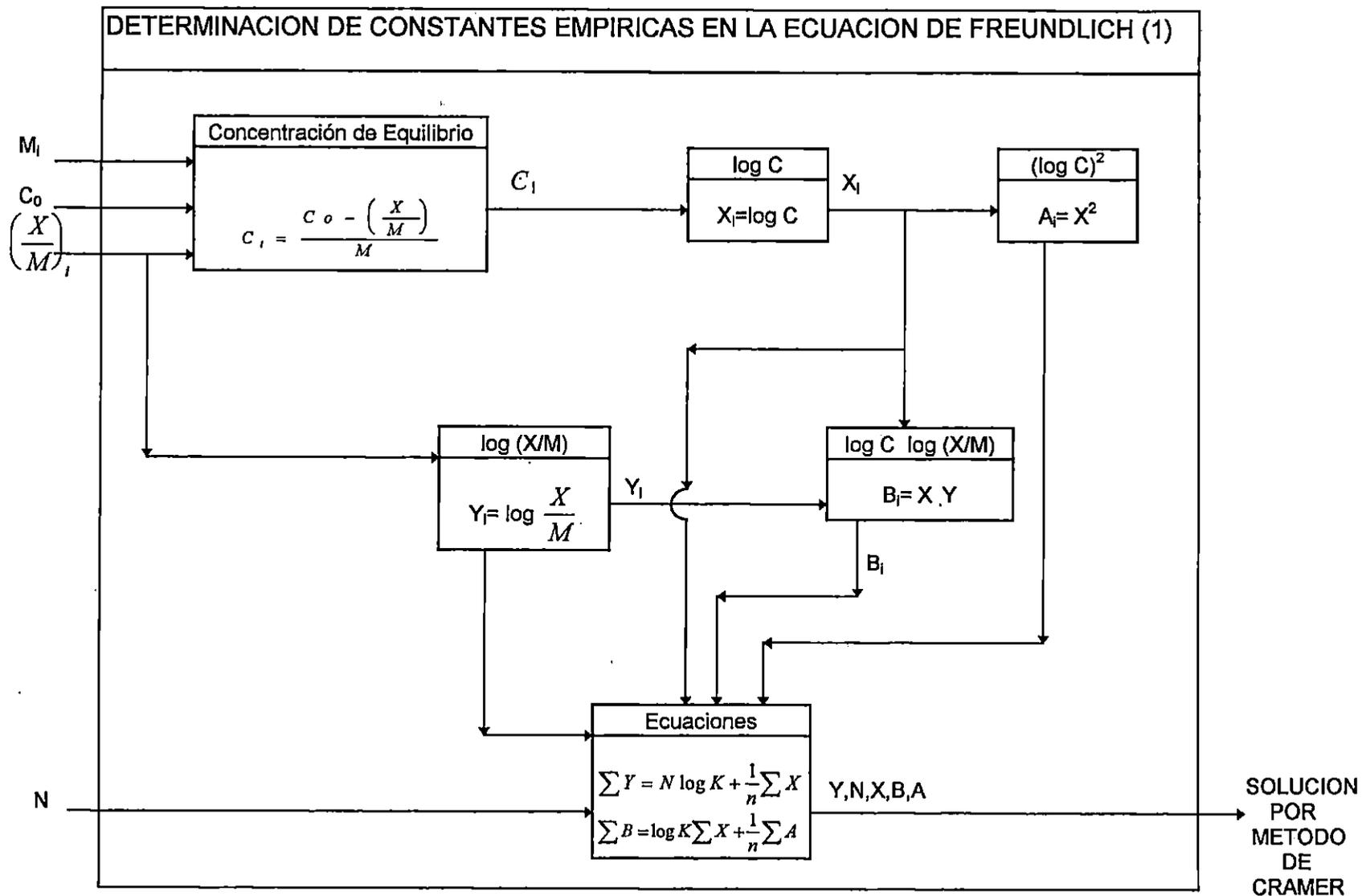


DIAGRAMA 3.A CONSTANTES EMPIRICAS PARA LA EC. DE FREUNDLICH (CASO 1)

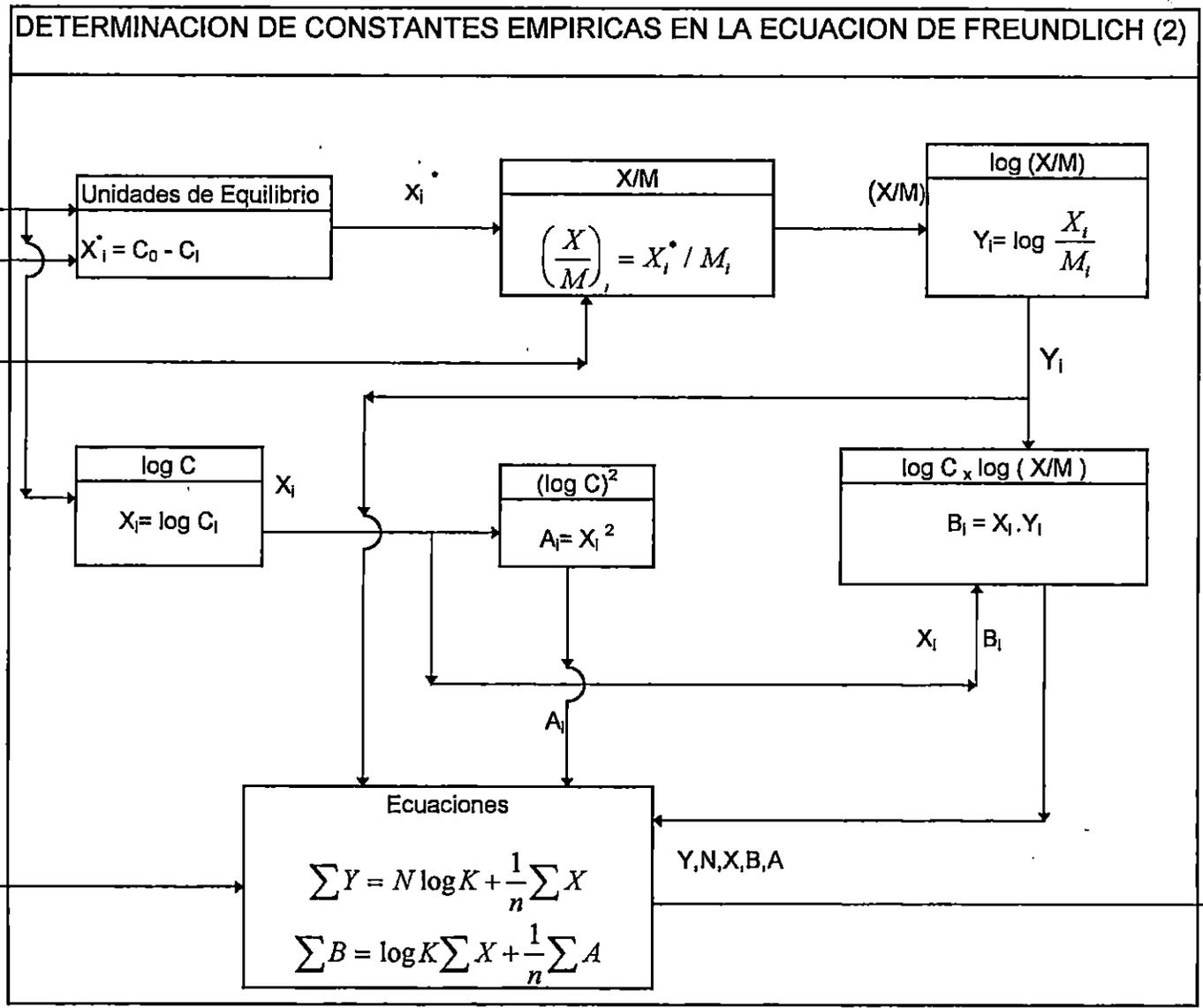


DIAGRAMA 3B CONSTANTES EMPIRICAS PARA LA EC. DE FREUNDLICH (CASO 2)

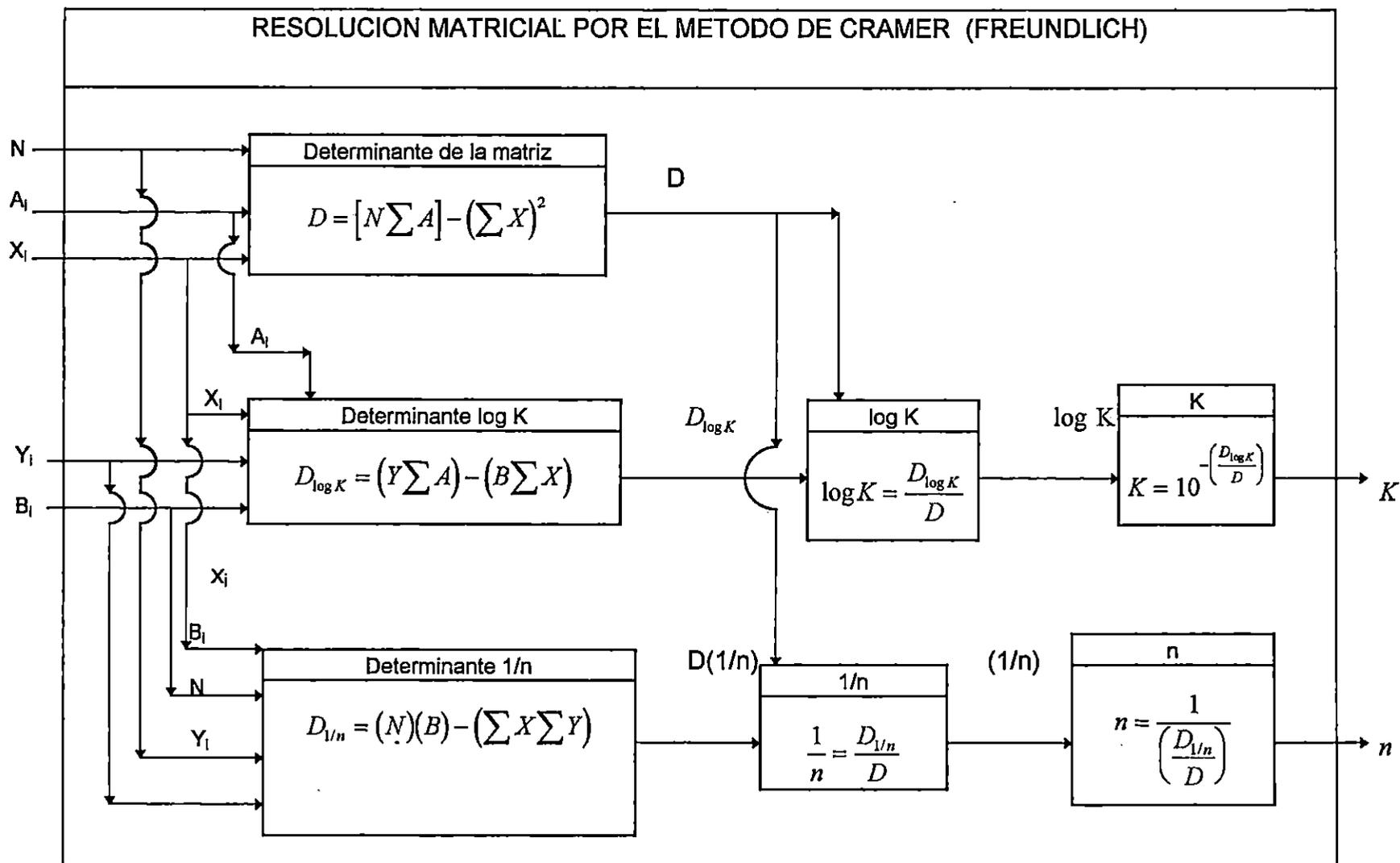


DIAGRAMA 3.C SOLUCION POR METODO DE CRAMER (FREUNDLICH)

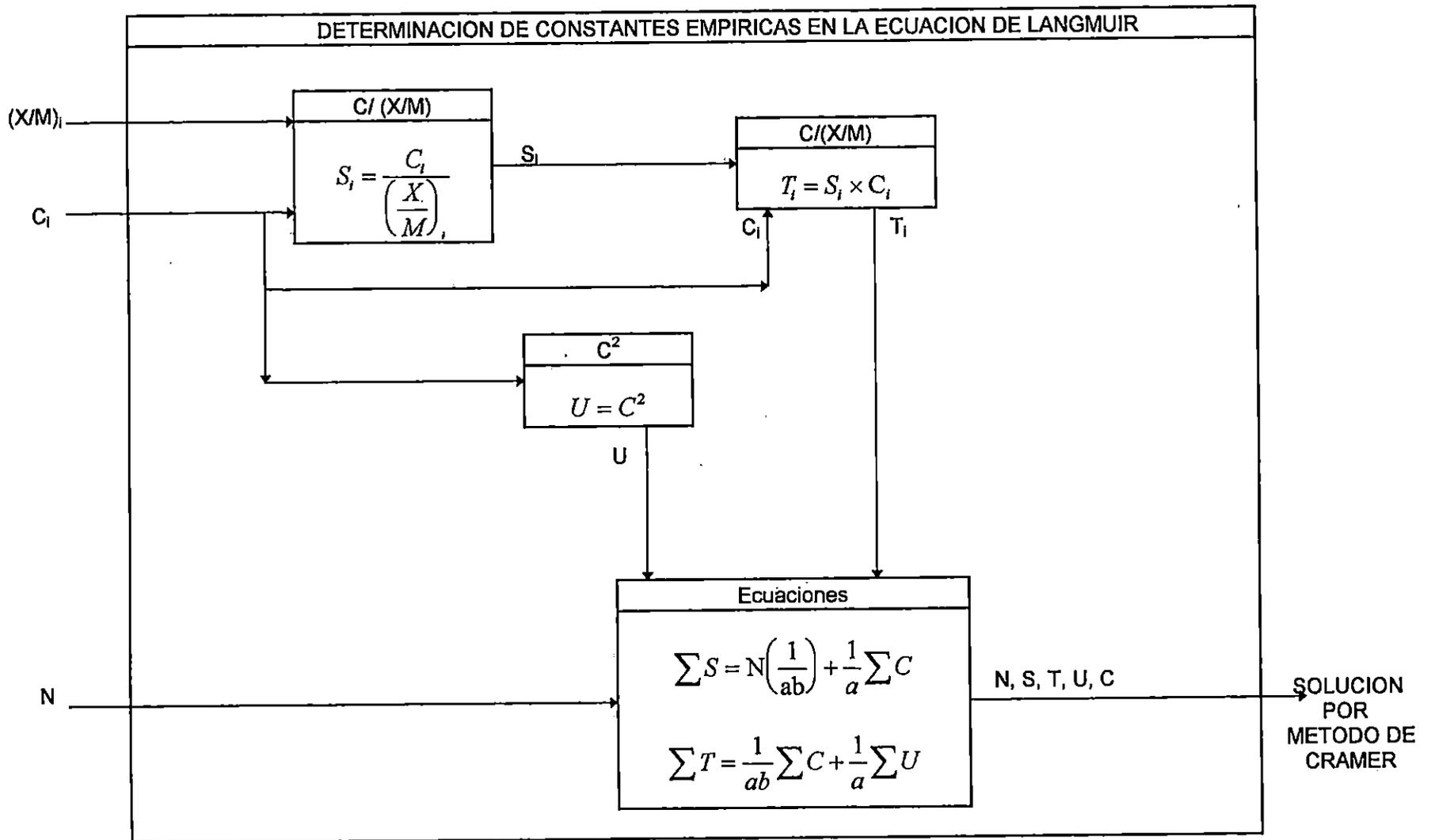


DIAGRAMA 3.D CONSTANTES EMPIRICAS PARA LA EC. DE LANGMUIR

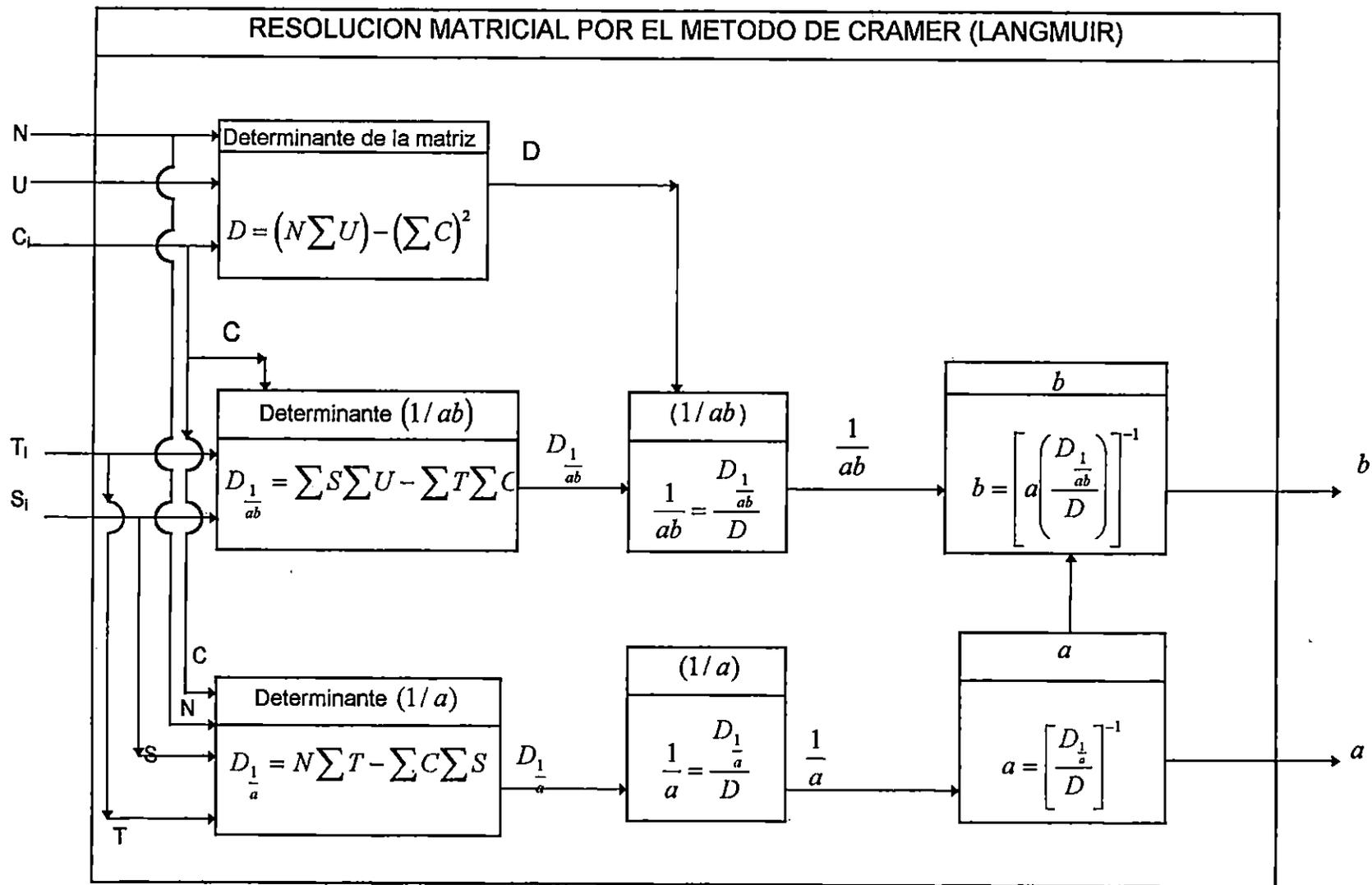


DIAGRAMA 3.E SOLUCION POR EL METODO DE CRAMER (LANGMUIR)

3.8 PROGRAMA PARA LA RESOLUCION DE ECUACIONES EN CARBON ACTIVADO EN QUICK C

```

#include "conio.h"
#include "stdio.h"
#include "stdlib.h"
#include "dos.h"
#include "stdarg.h"
#include "string.h"
/* ----- INCLUDE PROPIO ----- */
#include "math.h"
/* ----- DEF FUNC COMUNES ----- */
int CreaVentanaMensaje(int MensajeUnico, int NumeroPalabras,...);
void Cursor(int row, int col);
void BorraBloque(int top, int left, int btm, int right, int attr);
void DespliegaCadena(char *str, int attr, int row, int col);
void pca(char c, int attr, int row, int col);
void Ventana(int trow, int lcol, int brow, int rcol, int attr, int border,
char *title);
int getattr(void);
void cls(int attr);
int getvmode(int *iscolor);
void CreaOpcionesVertical(int Columna, int Fillni, int NumeroOpciones, ...);
/* ----- DEF FUNC. PROPIAS ----- */
int CapturaNumero(char *fld, int row, int col, int maxden, int attr);
void MenuCarbon(void);
int InicializaMatrizCarbon(void);
int ConstruyeMarcoMatriz(int FilSuplq, int ColSuplq, int FillnDch,
int NumeroColumnas, ...);

int Freundlich(int HallaX);
int Langmuir(void);
/* ----- DEFINE's COMUNES ----- */
#define IZQUIERDA 75
#define DERECHA 77
#define ARRIBA 72
#define ABAJO 80
#define F3 61
#define F1 59
#define ENTER 13
#define ESCAPE 27
#define WHITE 15
#define BLACK 0
#define BLUE 1
/* ----- DEFINE'S PROPIOS ----- */
#define OPCION_111 "Ec. de Freundlich (Busca x)"
#define OPCION_112 "Ec. de Freundlich (Busca y)"
#define OPCION_113 "Ec. de Langmuir (Busca x)"
#define OPCION_114 "Ec. de Langmuir (Busca y)"
#define MENSAJE_111 "Fórmula de Freundlich para la Adsorción de Carbón Activado. Halla C."
#define MENSAJE_112 "Fórmula de Freundlich para la Adsorción de Carbón Activado. Halla X/M."
#define MENSAJE_113 "Fórmula de Langmuir para la Adsorción de Carbón Activado."
/* #define MENSAJE_114 "Fórmula de Langmuir para la Adsorción de Carbón Activado. Halla X/M." */
#define MENSAJE_ERR01 "ERR01!: El Dato debe ser >= 0 y <= 100. Presione Cualquier Tecla."
#define MENSAJE_ERR02 "ERR02!: El Dato debe ser > 0. Presione Cualquier Tecla."
#define MENSAJE_ERR03 "ERR03!: Error en el Valor Ingresado. Presione Cualquier Tecla."
#define MENSAJE_ERR04 "ERR04!: Memoria Insuficiente para esta Opción. Presione ESCAPE."
#define MAGENTA 5
/* ----- VARIABLES COMUNES ----- */
int i, PosicionMenu;
int attr0, attr1, attr2, attr3, iscolor;
char temp[80], Imprimir[1];
FILE *pfile5;
/* ----- VARIABLES PROPIAS ----- */
int SalirOpcion_11, NumeroCorrecto, NumeroDatos, attr4;
double Coef1, Coef2, Coef3, Coef4, Const1, Const2;

```

```
double D,Dk,Dn,K,n,Dab,Da,a,b,ValorInicial;
char NumeroEnCadena[80],Etiqueta[7][14];
double MC[16][6];
```

```
/* _____ FUNCIONES COMUNES _____ */
/* _____ */
void Cursor(int row, int col)
/* _____ */
{
    /* envia el Cursor a la Posición (row=0,col=0 Origin) */
    union REGS rin,rout;
    rin.h.ah=2;
    rin.h.bh=0;
    rin.h.dh=row;
    rin.h.dl=col;
    int86(0x10,&rin,&rout);
}

/* _____ */
void DespliegaCadena(char *str, int attr, int row, int col)
/* _____ */
{
    /* Escribe una Cadena con Atributo y coordenadas */
    while (*str)
        pca(*str++,attr,row,col++);
}

/* _____ */
void BorraBloque(int top, int left, int btm, int right, int attr)
/* _____ */
{
    /* Clears Screen Block w/Attribute, Cursor Inside */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.al=0;
    rin.h.bh=attr;
    rin.h.ch=top; /* top row */
    rin.h.cl=left; /* top column */
    rin.h.dh=btm; /* bottom row */
    rin.h.dl=right; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(top+1,left+1);
}

/* _____ */
int getatr(void)
/* _____ */
{
    union REGS rin,rout;
    rin.h.ah=8; /* get attribute at Cursor */
    rin.h.bh=0;
    int86(0x10,&rin,&rout);
    return((int)rout.h.ah);
}

/* _____ */
void pca(char c, int attr, int row, int col)
/* _____ */
{
    union REGS rin,rout;
    Cursor(row,col); /* set cursor */
    rin.h.ah=9; /* print char with attribute */
    rin.h.al=c;
    rin.h.bh=0;
    rin.h.bl=attr;
    rin.x.cx=1;
    int86(0x10,&rin,&rout);
}

/* _____ */
void Ventana(int trow, int lcol, int brow, int rcol, int attr,
int border, char *title)
/* _____ */
```

```

{
    /* Set Simple Window and Title */
    int tl,tr,bl,br,lr,tb,i,length;
    BorraBloque(trow,lcol,brow,rcol,attr);
    if(border==1){ /* Single Border */
        tl=218; tr=191; bl=192; br=217; lr=196; tb=179;
    }
    else{ /* Double Border */
        tl=201; tr=187; bl=200; br=188; lr=205; tb=186;
    }
    pca(tl,attr,trow,lcol);
    pca(bl,attr,brow,lcol);
    pca(tr,attr,trow,rcol);
    pca(br,attr,brow,rcol);
    for(i=lcol+1; i<rcol; i++){
        pca(lr,attr,trow,i);
        pca(lr,attr,brow,i);
    }
    for(i=trow+1; i<brow; i++){
        pca(tb,attr,i,lcol);
        pca(tb,attr,i,rcol);
    }
    i=0;
    while(title[i++]!='\0');
    length=i-1;
    DespliegaCadena(title,attr,trow,(rcol+lcol)/2-length/2);
}

/*-----*/
void cls(int attr)
/*-----*/
{
    /* Clears Screen w/Attribute, Cursor to Origin */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.al=0;
    rin.h.bh=attr;
    rin.h.ch=0; /* top row */
    rin.h.cl=0; /* top column */
    rin.h.dh=24; /* bottom row */
    rin.h.dl=79; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(0,0);
}

/*-----*/
int getvmode(int *iscolor)
/*-----*/
{
    union REGS rin,rout;
    int mode;
    rin.h.ah=15; /* get display mode */
    int86(0x10,&rin,&rout);
    mode=(int)rout.h.al;
    *iscolor=1;
    if(mode==0 || mode==2 || mode==7)
        *iscolor=0;
    return(mode); /* sets color flag, returns video mode */
}

/*-----*/
int CreaVentanaMensaje(int Unico, int NumeroPalabras, ...)
/*-----*/
{
    int Bändera = 1;
    int Longitud;
    char *Rotulo;

    va_list parameters;
    va_start(parameters, NumeroPalabras);

    strcpy(temp,"");
}

```

```

BorraBloque(22,1,22,78,attr2);
if (Unico)
{
    Rotulo = va_arg(parameters, char *);
    DespliegaCadena(Rotulo,attr1,22,3);
    return(0);
}
Longitud = 3;
for (i=0; i<NumeroPalabras; i++)
{
    Rotulo = va_arg(parameters, char *);
    if (Bandera)
    {
        DespliegaCadena(Rotulo,attr3,22,3+Longitud);
        Bandera = 0;
    }
    else
    {
        DespliegaCadena(Rotulo,attr1,22,3+Longitud);
        Bandera = 1;
    }
    Longitud = Longitud + strlen(Rotulo) + 2;
}
return(0);
}

/* ----- */
void CreaOpcionesVertical(int Columna, int Fillni, int NumeroOpciones, ...)
/* ----- */
{
    int i,j;
    char *Rotulo;

    va_list parameters;
    va_start(parameters, NumeroOpciones);
    Rotulo = va_arg(parameters, char *);
    j = Fillni;
    DespliegaCadena(Rotulo,attr0,Fillni,Columna);
    Fillni ++;
    for (i = 0; i < NumeroOpciones - 1; i++)
    {
        Rotulo = va_arg(parameters, char *);
        DespliegaCadena(Rotulo,attr1,Fillni+i,Columna);
    }
    va_end(parameters);
    Cursor(j,Columna);
}

/* ----- FUNCIONES PROPIAS ----- */

/* ----- */
int CapturaNumero(char *fld, int row, int col, int maxlen, int attr)
/* ----- */
/* Simple Numeric Input Routine
 * fld storage for number as char string
 * row,col starting cursor position for field
 * maxlen field length
 * attr entry field attribute
 * returns entry length(x), -entry length(y), 0(ESC hit) */
{
    int i,j,yflag,expflag,pass;
    char c;

    for(j=0;j<maxlen;j++) pca(32,attr,row,col+j);
    Cursor(row,col);
    i=j=0;
    while(i<maxlen)
    {
        c=getch();
        pass=0;

```

```

if((c>='0' && c<='9') || c=='.' || pass)
{
    pca(c,attr,row,col+i); /* print character */
    Cursor(row,col+i+1); /* advance cursor */
    i++;
    if(!pass) fld[j++] = c;
}
else
    if (c==27) return(-2);
    else
        if((c==10 || c==13 || i==maxden) && il=0) break;
        else
            if(c=='\b' && i>0)
            {
                i--;
                pca(32,attr,row,col+i);
                if(j!=0)
                {
                    j--;
                    fld[j]=0;
                }
            }
            else if(c==27) return(0);
}
fld[j]='\0';
return(i);
}

/* ----- */
int InicializaMatrizCarbon(void)
/* ----- */
{
    int i,j;

    /* Reservando en Memoria */
    if (malloc(96 * sizeof(double)) == NULL) return (-1);

    for (i = 0; i <= 15; i++)
        for (j = 0; j <= 6; j++)
            MC[i][j] = 0;

    return 0;
}

/* ----- */
int ConstruyeMarcoMatriz(int FilSuplzq, int ColSuplzq, int FilInfDch,
    int NumeroColumnas, ...)
/* ----- */
{
    int i,j,ArregloAnchoCol[10],ColumnaActual, FilaLimite;
    va_list parameters;
    va_start(parameters, NumeroColumnas);

    if (!NumeroColumnas) return (-1);

    FilaLimite = FilSuplzq + 2;

    Cursor(FilSuplzq,ColSuplzq);
    putchar('E');
    Cursor(FilSuplzq+1,ColSuplzq);
    putchar('O');
    Cursor(FilaLimite,ColSuplzq);
    putchar('I');
    for (i = FilaLimite + 1; i < FilInfDch; i++)
    {
        Cursor(i,ColSuplzq);
        putchar('O');
    }
    Cursor(FilInfDch,ColSuplzq);
    putchar('E');
}

```

```

for (i = 0; i < NumeroColumnas; i++)
    ArregloAnchoCol[j] = va_arg(parameters, int);

ColumnaActual = ColSuplzq + 1;

while(NumeroColumnas)
{
    j = 0;
    for(i = 1; i <= ArregloAnchoCol[j]; i++)
    {
        Cursor(FilSuplzq,ColumnaActual);
        putch('I');
        Cursor(FilaLimite,ColumnaActual);
        putch('I');
        Cursor(FilInfDch,ColumnaActual);
        putch('I');
        ColumnaActual = ColumnaActual + 1;
    }

    if (NumeroColumnas > 1)
    {
        Cursor(FilSuplzq,ColumnaActual);
        putch('E');
        Cursor(FilSuplzq + 1, ColumnaActual);
        putch('°');
        Cursor(FilaLimite,ColumnaActual);
        putch('I');
        for(i = FilaLimite + 1; i < FilInfDch; i++)
        {
            Cursor(i,ColumnaActual);
            putch('°');
        }
        Cursor(FilInfDch,ColumnaActual);
        putch('E');
        ColumnaActual = ColumnaActual + 1;
    }

    NumeroColumnas = NumeroColumnas - 1;
    j = j + 1;
}

Cursor(FilSuplzq,ColumnaActual);
putch('»');
Cursor(FilSuplzq + 1,ColumnaActual);
putch('°');
Cursor(FilaLimite, ColumnaActual);
putch('I');
for(i = FilaLimite + 1; i < FilInfDch; i++)
{
    Cursor(i,ColumnaActual);
    putch('°');
}
Cursor(FilInfDch,ColumnaActual);
putch('¼');

return 0;
}

/* _____ */
int Freundlich(int HallaX)
/* _____ */
{
    char NumeroImpreso[80];
    int i;
    InicializaMatrizCarbon();
    CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones. Recuerde Y = X/M");

    if (malloc(40 * sizeof(char)) == NULL) return (-1);

```

```

BorraBloque(0,0,20,79,attr2);

/* Etiqueta[0][0] = 'M';
Etiqueta[0][1] = '\0'; */
strcpy(Etiqueta[0], "M");
Cursor(1,10);
DespliegaCadena(Etiqueta[0],attr2,1,10);
/* putchar(Etiqueta[0][0]); */
strcpy(Etiqueta[3], "log(C)");
DespliegaCadena(Etiqueta[3],attr2,1,37);
strcpy(Etiqueta[4], "log Y");
DespliegaCadena(Etiqueta[4],attr2,1,48);
strcpy(Etiqueta[5], "log Clog Y");
DespliegaCadena(Etiqueta[5],attr2,1,56);
strcpy(Etiqueta[6], "(log C)Y");
DespliegaCadena(Etiqueta[6],attr2,1,67);

if(HallaX) /* Quiere Hallar C */
{
strcpy(Etiqueta[1], "Y");
strcpy(Etiqueta[2], "C");
/* Etiqueta[2][0] = 'C';
Etiqueta[2][1] = '\0'; */
DespliegaCadena(Etiqueta[1],attr2,1,20);
DespliegaCadena(Etiqueta[2],attr2,1,30);
/* Cursor(1,30);
putch(Etiqueta[2][0]); */
}
else /* Quiere hallar X/M */
{
/* Etiqueta[1][0] = 'C';
Etiqueta[1][1] = '\0'; */
strcpy(Etiqueta[1], "C");
DespliegaCadena(Etiqueta[1],attr2,1,20);
strcpy(Etiqueta[2], "Y");
/* Cursor(1,20);
putch(Etiqueta[1][0]); */
DespliegaCadena(Etiqueta[2],attr2,1,30);
}

ConstruyeMarcoMatriz(0,5,18,7,9,9,9,9,21,9);

strcpy(temp, "");
DespliegaCadena("Nºmero de Datos:",attr2,19,1);
NumeroDatos = 0;
while (!NumeroDatos)
{
NumeroCorrecto = CapturaNumero(NumeroEnCadena,19,18,2,attr0);
NumeroDatos = atoi(NumeroEnCadena);
if (NumeroDatos == 0) return 0;
else
if ((NumeroDatos > 15) || (NumeroDatos < 0))
{
NumeroDatos = 0;
CreaVentanaMensaje(1,1,"Numero debe ser > 0 y <= 15. Presiones Cualquier Tecla.");
while(!kbhit());
CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones.");
}
}

strcpy(temp, "");
DespliegaCadena("Valor Inicial(Co):",attr2,19,25);
ValorInicial = 0;
NumeroCorrecto = 0;
while(!NumeroCorrecto)
{
NumeroCorrecto = CapturaNumero(NumeroEnCadena,19,43,9,attr0);
if (NumeroCorrecto > 0) ValorInicial = atof(NumeroEnCadena);
else

```

```

{
    NumeroDatos = 0;
    CreaVentanaMensaje(1,1,"Numero debe ser <= 15 y > 0. Presiones Cualquier Tecla.");
    while(!kbhit());
    CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones.");
}
}

for(i=0; i <= (NumeroDatos-1); i++)
{
    NumeroCorrecto = CapturaNumero(NumeroEnCadena,3+i,6,9,atr0);
    if (NumeroCorrecto>0) MC[i][0] = atof(NumeroEnCadena);
    else if (NumeroCorrecto==2) return(0);

    NumeroCorrecto = 0;
    while (!NumeroCorrecto)
    {
        NumeroCorrecto = CapturaNumero(NumeroEnCadena,3+i,16,9,atr0);
        if (NumeroCorrecto>0)
        {
            if (atof(NumeroEnCadena) > ValorInicial)
            {
                NumeroCorrecto = 0;
                CreaVentanaMensaje(1,1,"Los Cn no deben ser mayores que Co. Presione cualquier Tecla.");
                while(!kbhit());
                CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones.");
            }
            else MC[i][1] = atof(NumeroEnCadena);
        }
        else if (NumeroCorrecto==2) return(0);
    }

    MC[i][2] = (ValorInicial - MC[i][1]) / MC[i][0];

    if (HallaX)
    {
        MC[i][3] = log10(MC[i][2]);
        MC[i][4] = log10(MC[i][1]);
        MC[i][5] = log10(MC[i][1]) * log10(MC[i][2]);
        MC[i][6] = log10(MC[i][2]) * log10(MC[i][2]);
    }
    else
    {
        MC[i][3] = log10(MC[i][1]);
        MC[i][4] = log10(MC[i][2]);
        MC[i][5] = log10(MC[i][1]) * log10(MC[i][2]);
        MC[i][6] = log10(MC[i][1]) * log10(MC[i][1]);
    }
    printf(NumeroImpreso,"%5.3f",MC[i][2]);
    DespliegaCadena(NumeroImpreso,atr2,3+i,26);
    printf(NumeroImpreso,"%5.3f",MC[i][3]);
    DespliegaCadena(NumeroImpreso,atr2,3+i,36);
    printf(NumeroImpreso,"%5.3f",MC[i][4]);
    DespliegaCadena(NumeroImpreso,atr2,3+i,46);
    printf(NumeroImpreso,"%5.3f",MC[i][5]);
    DespliegaCadena(NumeroImpreso,atr2,3+i,56);
    printf(NumeroImpreso,"%5.3f",MC[i][6]);
    DespliegaCadena(NumeroImpreso,atr2,3+i,66);

    /* #logC */
    MC[15][3] = MC[15][3] + MC[i][3];
    /* #log(X/M) */
    MC[15][4] = MC[15][4] + MC[i][4];
    /* #logClog(X/M) */
    MC[15][5] = MC[15][5] + MC[i][5];
    /* #logC y */
    MC[15][6] = MC[15][6] + MC[i][6];
}
}

```

```

CreaVentanaMensaje(1,1,"Datos Procesados. Presione Cualquier Tecla.");
while(!kbhit());

/* M,todo de Cramer */
Const1 = MC[15][4];
Coef1 = NumeroDatos;
Coef2 = MC[15][3];
Const2 = MC[15][5];
Coef3 = MC[15][3];
Coef4 = MC[15][6];
D = (Coef1 * Coef4) - (Coef3 * Coef2);
Dk = (Const1 * Coef4) - (Const2 * Coef2);
Dn = (Coef1 * Const2) - (Coef3 * Const1);
K = pow(10,Dk / D);
n = Dn / D; /* En realidad despliega 1/n */

BorraBloque(0,0,20,79,attr2);
strcpy(temp,"");
Ventana(5,20,10,60,attr4,2,temp);
DespliegaCadena("Soluci3n a la Ecuaci3n X/M = K(00C) :",attr4,6,21);
sprintf(NumeroImpreso,"Valor de K: %3.8f",K);
DespliegaCadena(NumeroImpreso,attr4,7,25);
sprintf(NumeroImpreso,"Valor de 1/n: %5.3f",n);
DespliegaCadena(NumeroImpreso,attr4,8,25);

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,attr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]?",attr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getch();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(22,3,22,70,attr2);
        DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5,"***** Metodo de Freundlich *****\n");
    fprintf(pfile5,"Solucion a la Ecuacion :\n");
    fprintf(pfile5,"Valor Inicial (Co): %5.3f\n",ValorInicial);
    fprintf(pfile5,"Numero de Datos : %2i \n",NumeroDatos);
    fprintf(pfile5,"Valor de K : %3.8f \n",K);
    fprintf(pfile5,"Valor de 1/n : %5.3f \n",n);
    fclose(pfile5);
}

CreaVentanaMensaje(1,1,"Presione ESC para salir.");
while(getch() != ESCAPE);
return 0;
}

/* ----- */
int Langmuir(void)
/* ----- */
{
    char NumeroImpreso[80];
    int i;
    InicializaMatrizCarbon();
    CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones. Recuerde Y = X/M.");

    if (malloc(40 * sizeof(char)) == NULL) return (-1);

```

```

BorraBloque(0,0,20,79,attr2);

strcpy(Etiqueta[0],"X/M");
Cursor(1,10);
DespliegaCadena(Etiqueta[0],attr2,1,9);
/* Etiqueta[1][0] = 'C';
Etiqueta[1][1] = '/0';
Cursor(1,19);
putch(Etiqueta[1][0]); */
strcpy(Etiqueta[1],"C");
DespliegaCadena(Etiqueta[1],attr2,1,19);
strcpy(Etiqueta[2],"C/Y");
DespliegaCadena(Etiqueta[2],attr2,1,29);
strcpy(Etiqueta[3],"Cy/Y");
DespliegaCadena(Etiqueta[3],attr2,1,39);
strcpy(Etiqueta[4],"Cý");
DespliegaCadena(Etiqueta[4],attr2,1,49);

ConstruyeMarcoMatriz(0,5,18,5,9,9,9,9);

strcpy(temp,"");
DespliegaCadena("NEmero de Datos:",attr2,19,1);
NumeroDatos = 0;
while (!NumeroDatos)
{
    NumeroCorrecto = CapturaNumero(NumeroEnCadena,19,18,2,attr0);
    NumeroDatos = atoi(NumeroEnCadena);
    if (NumeroDatos == 0) return 0;
    else
        if ((NumeroDatos > 15) || (NumeroDatos < 0))
        {
            NumeroDatos = 0;
            CreaVentanaMensaje(1,1,"Numero debe ser <= 15 y > 0. Presione Cualquier Tecla.");
            while(!kbhit());
            CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones.");
        }
}

for(i=0; i <= (NumeroDatos-1); i++)
{
    NumeroCorrecto = 0;
    while(!NumeroCorrecto)
    {
        NumeroCorrecto = CapturaNumero(NumeroEnCadena,3+i,6,9,attr0);
        if (NumeroCorrecto>0) MC[i][0] = atof(NumeroEnCadena);
        else
        {
            NumeroCorrecto = 0;
            CreaVentanaMensaje(1,1,"Numero > 0. Presione Cualquier Tecla.");
            while(!kbhit());
            CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones.");
        }
    }
}

NumeroCorrecto = 0;
while(!NumeroCorrecto)
{
    NumeroCorrecto = CapturaNumero(NumeroEnCadena,3+i,16,9,attr0);
    if (NumeroCorrecto>0) MC[i][1] = atof(NumeroEnCadena);
    else
    {
        NumeroCorrecto = 0;
        CreaVentanaMensaje(1,1,"Numero > 0. Presione Cualquier Tecla.");
        while(!kbhit());
        CreaVentanaMensaje(1,1,"Ingrese los datos de sus observaciones.");
    }
}

MC[i][2] = MC[i][1] / MC[i][0];

```

```

MC[i][3] = MC[i][1] * MC[i][2];
MC[i][4] = MC[i][1] * MC[i][1];
sprintf(NumeroImpreso,"%5.3f",MC[i][2]);
DespliegaCadena(NumeroImpreso,atr2,3+i,26);
sprintf(NumeroImpreso,"%5.3f",MC[i][3]);
DespliegaCadena(NumeroImpreso,atr2,3+i,36);
sprintf(NumeroImpreso,"%5.3f",MC[i][4]);
DespliegaCadena(NumeroImpreso,atr2,3+i,46);

/* a(C) */
MC[15][1] = MC[15][1] + MC[i][1];
/* a(C/(X/M)) */
MC[15][2] = MC[15][2] + MC[i][2];
/* a(Cy/(X/M)) */
MC[15][3] = MC[15][3] + MC[i][3];
/* a(Cy) */
MC[15][4] = MC[15][4] + MC[i][4];
}

CreaVentanaMensaje(1,1,"Datos Procesados. Presiones Cualquier Tecla.");
while(!kbhit());

/* M,todo de Cramer */
Const1 = MC[15][2];
Coef1 = NumeroDatos;
Coef2 = Coef3 = MC[15][1];
Const2 = MC[15][3];
Coef4 = MC[15][4];
D = (Coef1 * Coef4) - (Coef3 * Coef2);
Dab = (Const1 * Coef4) - (Const2 * Coef2);
Da = (Coef1 * Const2) - (Coef3 * Const1);
a = D / Da;
b = D / (a * Dab);

BorraBloque(0,0,20,79,atr2);
Ventana(5,18,10,62,atr4,2,temp);
DespliegaCadena("Soluci3n a la Ecuaci3n X/M = (abC)/(1+bC):",atr4,6,20);
sprintf(NumeroImpreso,"Valor de a: %5.3f",a);
DespliegaCadena(NumeroImpreso,atr4,7,25);
sprintf(NumeroImpreso,"Valor de b: %5.3f",b);
DespliegaCadena(NumeroImpreso,atr4,8,25);

BorraBloque(21,0,23,79,atr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,atr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",atr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getch();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(22,3,22,70,atr2);
        DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",atr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5,"***** Metodo de Langmuir *****\n");
    fprintf(pfile5,"Soluci3n a la Ecuacion :\n");
    fprintf(pfile5,"Numero de Datos : %2i \n",NumeroDatos);
    fprintf(pfile5,"Valor de a : %5.3f \n",a);
    fprintf(pfile5,"Valor de b : %5.3f \n",b);
    fclose(pfile5);
}

```

```

CreaVentanaMensaje(1,1,"Presione ESC para salir.");
while(getch() != ESCAPE);

return 0;
}

/* ----- */
void MenuCarbon(void)
/* ----- */
{
    int Reconstruye = 1;
    BorraBloque(0,0,23,79,attr2);
    attr3=WHITE(5<<4); /* Atributos del Mensaje abajo en la Pantalla */
    attr2=WHITE(BLACK<<4); /* Atributos del Marco de Ventana Blanco/Negro */
    attr1=WHITE(BLACK<<4); /* Atributos del Texto Desactivado Blanco/Negro */
    attr0=WHITE(BLUE<<4); /* Atributos del Texto Activado Blanco/Azul */
    /* -----este atributo es propio ----- */
    attr4=WHITE(MAGENTA<<4); /* Atributos del Texto para respuestas Blanco/Cyan */
    /* ----- */
    while(!SalirOpcion_11)
    {
        if (Reconstruye)
        {
            SalirOpcion_11 = 0;
            PosicionMenu = 111;
            strcpy(temp,"Variables de Contexto:");
            Ventana(15,0,20,79,attr2,1,temp);
            DespliegaCadena("M = Dosis de Carbón",attr1,16,3);
            DespliegaCadena("C = Concentración de Eq. del Adsorbente",attr1,17,3);
            DespliegaCadena("Y = (X/M) Cantidad adsorbida por peso unitario de adsorbente",attr1,18,3);
            DespliegaCadena("Co = Concentración inicial de Eq.",attr1,19,3);
            DespliegaCadena("a,b = constante empírica (Langmuir)",attr1,16,37);
            DespliegaCadena("K,1/n = constantes empíricas (Freundlich)",attr1,19,37);
            DespliegaCadena("X = Unidades de Equilibrio",attr1,17,51);
            strcpy(temp,"");
            Ventana(6,27,10,55,attr2,1,temp);
            CreaOpcionesVertical(28,7,3,OPCION_111,OPCION_112,OPCION_113);
            Ventana(21,0,23,79,attr1,1,temp);
            CreaVentanaMensaje(1,1,MENSAJE_111);
            Cursor(7,28);
            Reconstruye = 0;
        }

        switch(getch())
        {
            case 72:
            {
                switch(PosicionMenu)
                {
                    case 111:
                    {
                        DespliegaCadena(OPCION_111,attr1,7,28);
                        CreaVentanaMensaje(1,1,MENSAJE_113);
                        DespliegaCadena(OPCION_113,attr0,9,28);
                        PosicionMenu = 113;
                        break;
                    }
                    case 112:
                    {
                        DespliegaCadena(OPCION_112,attr1,8,28);
                        CreaVentanaMensaje(1,1,MENSAJE_111);
                        DespliegaCadena(OPCION_111,attr0,7,28);
                        PosicionMenu = 111;
                        break;
                    }
                    case 113:
                    {
                        DespliegaCadena(OPCION_113,attr1,9,28);
                        CreaVentanaMensaje(1,1,MENSAJE_112);
                        DespliegaCadena(OPCION_112,attr0,8,28);
                    }
                }
            }
        }
    }
}

```



```
    }
    case 113:
    {
        Langmuir();
        BorraBloque(0,0,23,79,attr2);
        Reconstruye = 1;
        break;
    }
/*
    case 114:
    {
        Langmuir();
        BorraBloque(0,0,23,79,attr2);
        Reconstruye = 1;
        break;
    }*/
}
break;
}
case ESCAPE:
{
    BorraBloque(0,0,23,79,attr2);
    SalirOpcion_11 = 1;
}
}
}
BorraBloque(0,0,23,79,attr2);
}

void main(void)
{
    MenuCarbon();
}
```

CAPITULO IV

INTERCAMBIO IONICO

4.0 INTRODUCCIÓN

La desmineralización como aplicación al tratamiento de aguas, es esencialmente la remoción de todas las sales inorgánicas, disueltas en el agua, por medio de intercambio iónico. En este proceso, el intercambio del ión hidrógeno que esta en la resina convierte las sales disueltas a su correspondiente ácido, y el intercambio en la resina aniónica remueve estos ácidos. El otro proceso comercial que produce agua de pureza comparable a la del agua desmineralizada es la destilación. La desmineralización ha alcanzado un amplio uso, debido en gran parte a que (en la mayoría de suministros de agua natural) su costo es sustancialmente menor que la destilación

4.1 PRINCIPIOS DEL INTERCAMBIO IONICO.

El intercambio iónico es un fenómeno natural que ocurre en suelos, minerales, y tejidos tanto de plantas como de animales.

En éste proceso intervienen un gran número de factores que imposibilitan dar una sola teoría completa y satisfactoria. Sin embargo el estudio sistemático de diversos parámetros permite relacionar los fenómenos concernientes a las resinas con los aspectos fundamentales sobre soluciones y coloides. En la práctica, el intercambio iónico es un fenómeno dinámico, y su eficacia depende del tiempo de contacto entre la solución y la resina. El principal interés en este capítulo no es estudiar los efectos cinéticos y de equilibrio que puedan ocurrir en el intercambio iónico, más bien es el de relacionar los efectos globales del proceso, es decir lo que ocurre cuando se hace uso del intercambio iónico para la remoción de las sales minerales disueltas en el agua. Bajo este criterio se abordará esta sección con el objeto de establecer algunos parámetros de interés práctico para la aplicación del intercambio iónico en lo que se refiere al proceso de desmineralización total mediante el uso de resinas sintéticas.

4.1.1 DEFINICIÓN DE INTERCAMBIO IONICO

Se da el nombre de "intercambio iónico" a la reacción de intercambio (de iones) entre dos fases no miscibles que puede observarse entre ciertos compuestos sólidos, o más raramente entre líquidos y una solución iónica generalmente acuosa. Si llamamos M y M' a dos iones diferentes, se puede escribir la siguiente reacción de intercambio.



en donde:

RM y AM': son los resinosos que consiste en la estructura orgánica polimérica (macro-ion) conteniendo al ion M antes de la reacción, y al ion M' después de la reacción.

AM' y AM: son los electrólitos antes y después de la reacción .

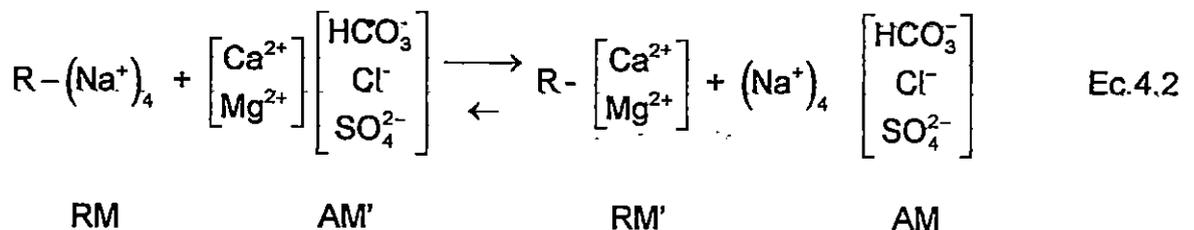
M y M' : son de signo contrario al macro-ion (resina sintética) y se conocen con el nombre de contraiones.

A :que es del mismo signo del macro-ion se conoce con el nombre de co-ion.

La ecuación 4.1 es válida tanto para el intercambio catiónico como el aniónico, por lo que no es necesario escribir los símbolos químicos de los iones M y M'.

La reacción de intercambio mostrada tiene por resultado la sustitución de un ion en disolución por otro introducido por un sólido al que se le llama "intercambiador" (resina). Por ejemplo, el proceso de suavización es completado cuando una resina catiónica en la forma de sodio se pone en contacto con agua que contiene iones calcio y magnesio (dureza). Los iones divalentes de calcio y magnesio se intercambian por el ion sodio sobre la resina sólida, con los iones sodio pasando adentro de la solución acuosa en lugar del calcio y magnesio, removiendo así la dureza y produciendo agua suave o suavizada.

La reacción puede ser representada como sigue.



La ecuación 4.2 muestra una reacción de intercambio balanceada con cuatro iones sodio monovalentes sobre la resina que están siendo reemplazados con cada uno de los iones divalentes calcio y magnesio, y los iones resultantes en el agua serán iones positivos y negativos balanceados puesto que este es un requerimiento necesario en todas las reacciones de intercambio iónico (ley de electroneutralidad).

Por otra parte, existe un buen número de condiciones que afectan las reacciones de intercambio iónico y que tienen influencia sobre la dirección en la cual procederán las reacciones y en el grado de conversión obtenido.

4.1.2 PROPIEDADES FISICAS Y QUIMICAS DE LAS RESINAS DE INTERCAMBIO IONICO

Los intercambiadores ionicos actuales (resinas) han venido sufriendo mejoras en lo relativo a capacidad de intercambio y las eficiencias de regeneración .

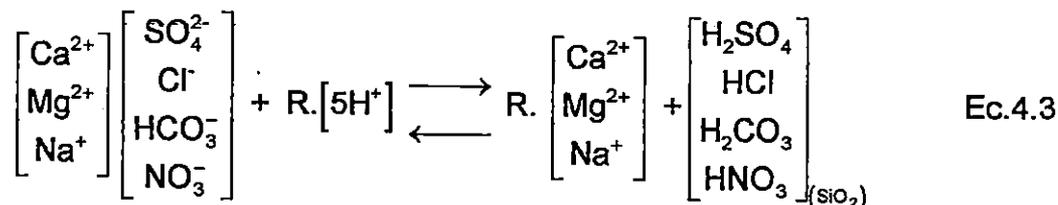
Comercialmente existen en la actualidad un gran número de resinas de intercambio ionico fabricadas por diversas compañías. Pueden encontrarse resinas de ácido y base fuertes, de ácido y base débiles, y resinas para usos especiales. Las resinas más resistentes son fabricadas de estireno divinilbenceno (DVC).

En las tablas 4.1 y 4.2 se presenta información de resinas catiónicas de ácido fuerte y resinas aniónicas de base fuerte, relativa a diferentes marcas comerciales de estos tipos de resinas

4.2 LA DESMINERALIZACIÓN.

4.2.1 PRINCIPIOS DE LA DESMINERALIZACIÓN TOTAL.

El proceso de desmineralización consiste en el intercambio de uno o más iones en columnas de resinas, que contienen al menos una resina catiónica fuertemente ácida y una resina aniónica, La resina catiónica intercambia hidrógenos (H^+) por los cationes del agua cruda como se muestra en la reacción.



Donde R representa la estructura de la resina para el intercambio catiónico.

Como puede observarse de la ecuación 4.3, el efluente de agua que sale de una columna de resina catiónica es ácido.

Donde R representa la estructura de la resina para el intercambio aniónico.

Si la resina de intercambio aniónica es fuertemente básica, esta removerá además componentes débilmente ionizados como el ácido carbónico y sílice, de acuerdo con la ecuación 4.5

TABLA 4.1 ALGUNOS TIPOS DE RESINAS ANIONICAS DE BASE FUERTE DISPONIBLES COMERCIALMENTE
(Fuentes et al., 1994)

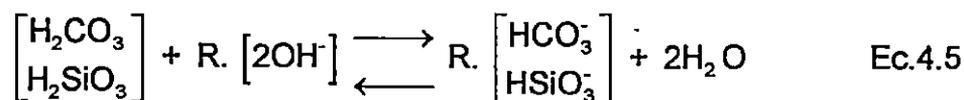
Matriz	Grupo iónico	Nombre comercial	Fabricante	Capacidad		Contenido de humedad %peso	Temperatura máxima °C	Rango de PH	Forma física	Observaciones
				meq/g resina seca	meq/ml lecho resina					
Resinas de poliestireno	$-N(CH_3)_3^+$	Amberlite IRA-400	Rohn and hass Co.	2.6	1.2	42-48	60	0-12	perlas esféricas	Resina estándar, aprox. 8% DVB
		Amberlite IRA-401	Rohn and hass Co.	3.0	1.0	54-59	60	0-12	perlas esféricas	Resina porosa
		Amberlite IRA-425	Rohn and hass Co.	3.0	1.3	50-53	60	0-12	perlas esféricas	Alta capacidad
		D-Acidite FF	Permutit Co. Ltd. Inglaterra	4.0	1.6	42-48	60	0-12	perlas esféricas	Resina estándar, 7-9% DVB
		Dovex 21K	Dow Chemical Co.	4.5	1.2	57	50	0-12	perlas esféricas	Estabilidad mecánica mejorada
		Duolite A-42	Chemical process Co.	2.3	0.7	57	60	0-12	perlas esféricas	Resina porosa, baja capacidad
		Duolite 101D	Chemical process Co.	4.2	1.4	57	60	0-14	perlas esféricas	Más porosa que la A-42
		Lewatit MP-500	Farbenfabriken Bayer, Alemania	4.0	1.2	55-60	70	0-14	perlas esféricas	Naturaleza del grupo iónico desconocido
		Permutit ESB-26	Permutit A.G., Berlin, Alemania	3.3	1.3	60-65	70	1-14	perlas esféricas	Resina estándar con 8% DVB, más porosa
		Resanex HBL	Jos. Crosfield, Inglaterra	3.5	1.5	60-65	60	1-14	perlas esféricas	Resina estándar con 8% DVB, porosa

DVB= Divinilbenceno

TABLA 4.2 ALGUNOS TIPOS DE RESINAS DE ACIDO FUERTE DISPONIBLES COMERCIALMENTE
(Fuentes et al., 1994)

Matriz	Grupo iónico	Nombre comercial	Fabricante	Capacidad		Contenido de humedad %peso	Temperatura máxima °C	Rango de PH	Forma física	Observaciones
				meq/g resina seca	meq/ml lecho resina					
Resinas de poliestireno	- SO ₃	Amberlite IR-120	Rohn and hass Co.	4.3-5	1.9	44-48	120	0-14	perlas esféricas	Resina estándar, aprox. 8% DVB
		Amberlite IR-122	Rohn and hass Co.	4.3-5	2.1	40-44	120	0-14	perlas esféricas	Aprox. 10% DVB, alta resistencia a oxidación
		Amberlite IR-124	Rohn and hass Co.	4.3-5	2.1	37-41	120	0-14	perlas esféricas	Aprox. 12% DVB, alta resistencia a oxidación
		Amberlite 200	Rohn and hass Co.	4.3	1.75	47-52	120	0-14	perlas esféricas	Más alta estabilidad mecánica y química
		Amberlite XE-100	Rohn and hass Co.	4.9	1.2	58-65	120	0-14	perlas esféricas	Aprox. 4% DVB, baja capacidad
		Amberlyst 15	Rohn and hass Co.	4.9	1.2	60-66	120	0-14	perlas esféricas	Resinas macroreticular
		Duolite C-20	Chemical process Co.	5.1	2.2	45-51	150	0-14	perlas esféricas	Resina estándar aprox. 8% DVB
		Duolite C-25	Chemical process Co.	5.1	1.7	55-62	120	0-14	perlas esféricas	Resina porosa
		Duolite C-27	Chemical process Co.	5.0	2.1	45-50	150	0-14	perlas esféricas	Resina estándar aprox. 8% DVB
		Imac C-12	Activit, Holland	4.5	2.0	---	120	0-14	perlas esféricas	Resina estándar aprox. 8% DVB

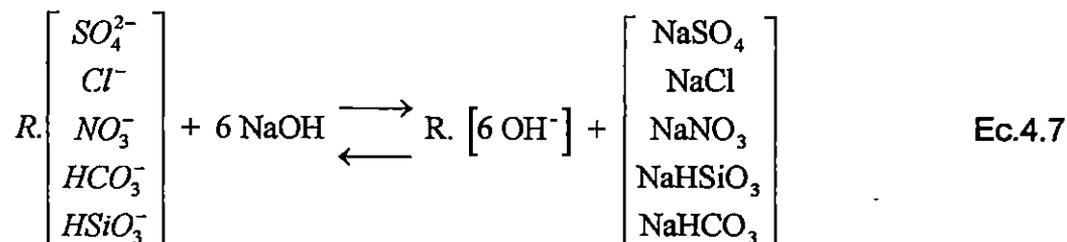
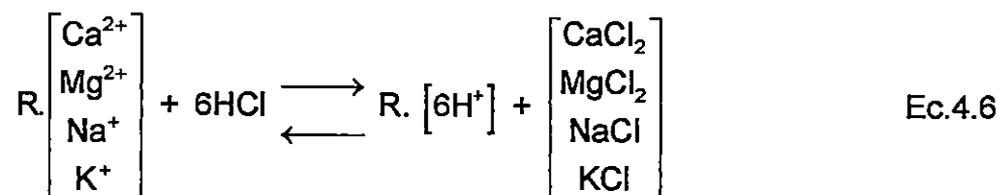
DVB= Divinilbenceno



Si la resina aniónica es débilmente básica, esta no removerá los ácidos orgánicos y la sílice.

Cuando la remoción de la sílice es requerida, se usan resinas aniónicas de base fuerte y en este caso los parámetros más importantes para evaluar la calidad final del agua-producto son el contenido de sílice y la conductividad.

La parte más importante del proceso de desmineralización es la regeneración de las resinas por medio de la cual se revierten las reacciones que se describieron anteriormente. La regeneración ocurre de acuerdo a las ecuaciones 4.6 y 4.7.



En ambas reacciones la fuerza guiadora, necesaria para que se verifiquen, esta determinada por la concentración relativamente alta de las sustancias regenerantes. Esta es una condición definida por los principios del intercambio iónico. HCl y el NaOH en las ecuaciones de regeneración son los regenerantes más comúnmente utilizados.

4.3 DIMENSIONAMIENTO DE UN SISTEMA DESMINERALIZADOR

El dimensionamiento consiste en la especificación de todas las partes componentes del sistema desmineralizador.

El primer paso consiste en determinar el volumen o cantidad de resinas (aniónica o catiónica) necesaria, flujo de operación y tamaño de los tanques que contendrán a las resinas.

4.3.1 CALCULOS PARA DETERMINAR EL TAMAÑO DEL DESMINERALIZADOR DE LECHO DOBLE.

Para la especificación de un desmineralizador, lo primero que debe hacerse es determinar las características químicas (o calidad) del agua que se desea someter al proceso (agua cruda).

El cálculo de las unidades catiónica y aniónica se harán en base a lo establecido por la guía de dimensionamiento para sistemas de desmineralización de la Culligan Company de los Estados Unidos de Norte America.

Previo al cálculo numérico, se debe analizar los datos técnicos específicos de las resinas a utilizar; así como lo concerniente a los niveles de regeneración y sus efectos en las diferentes etapas del proceso de desmineralización.

4.3.2 DATOS TECNICOS DE LAS RESINAS.

Dado que existen diversos tipos de intercambiadores de ácido y base fuerte se hace necesario especificar las características de cada resina a utilizar.

4.3.2.1 RESINA CATIONICA

En la tabla 4.3 se presentan las características físicas principales de esta resina, y en la tabla 4.4 se sugieren las condiciones de operación para obtener optimos resultados.

4.3.2.2 RESINA ANIONICA

Cuando se requiere una remoción de sílice muy completa, como sucede en aguas de alimentación para calderas de alta presión, esta resina de base fuerte, con una regeneración apropiada producirá un efluente de excelente calidad.

TABLA 4.3 CARACTERISTICAS FISICAS DE LA RESINA CATIONICA DE ACIDO FUERTE (Fuentes et al, 1994)

Tipo	Gel
Forma iónica	Sodio o Hidrógeno
Forma	Partículas esféricas
Humedad	44 a 48%
Capacidad de intercambio	1.9 meq/ml
Densidad	48 a 54 lbs/pie ³
Peso de embarque	53 lbs/pie ³
Tamaño efectivo	0.45 a 0.60 mm
Clasificación del tamaño	16 a 50 mesh
Coefficiente de uniformidad	1.8 máximo
Coefficiente de finos	0.7% máximo (en 50 mesh)

TABLA 4.4 CONDICIONES DE OPERACION SUGERIDAS PARA LA RESINA CATIONICA DE ACIDO FUERTE (Fuentes et al, 1994)

CICLO DE HIDROGENO	
pH	1.0 a 14.0
Temperatura máxima	250°F ó 121°C
Altura mínima del lecho	24 plgs.
Velocidades de flujo de retrolavado	6 gal/pie ² .min a 72°F a 22°C
Concentración del regenerante	10% p/p HCl ó 5% p/p H ₂ SO ₄
Velocidades de flujo de regeneración	0.5 a 0.75 gal/pie ³ .min
Velocidades de flujo de lavado	0.5 a 0.75 gal/pie ³ .min inicialmente, luego 1.5 gal/pie ³ .min
Requerimiento del agua de lavado	25-50 galones/pie ³
Velocidad de flujo de servicio	2 gal/pie ³ .min

La tabla 4.5 muestra las propiedades físicas y químicas de la resina aniónica de base fuerte, y la tabla 4.6 las condiciones de operación sugeridas para obtener resultados óptimos.

TABLA 4.5 PROPIEDADES FISICAS Y QUIMICAS DE LA RESINA ANIONICA DE BASE FUERTE (Fuentes et al, 1994)

Tipo	Gel
Forma iónica	cloruro
Estructura	Partículas esféricas
Humedad	42 a 48%
Densidad	40 a 46 lbs/pie ³
Peso de embarque	44 lbs/pie ³
Medida efectiva	0.41 a 0.48 mm
Graduación en malla	16 a 50 mesh (U.S. tamaño estándar)
Coefficiente de uniformidad	1.8 máximo
Contenido de finos	3.7% máximo a través de 50 mesh
Hinchamiento	Máximo 22% sobre completa conversión desde la forma de cloruro a la forma oxhidrilo (OH ⁻)
CAPACIDAD TOTAL DE INTERCAMBIO	
a) Volumétrica	1.4 meq/ml mínimo 30.5 kgr/pie ³ como CaCO ₃ 4.4 lb eq/pie ³ como CaCO ₃
b) Peso	3.8 meq/g de resina seca en la forma de cloruro (mínimo)

TABLA 4.6 CONDICIONES DE OPERACION SUGERIDAS PARA LA RESINA ANIONICA FUERTEMENTE BASICA

Limitaciones de pH	ninguno
Altura de lecho mínimo	24 plg
Velocidad de flujo de retrolavado	2 a 3 gal/pie ² /min a 70°F
Concentración de regenerante	4% p/p NaOH
Velocidad de flujo de regeneración	0.25 a 1.0 gal/pie ³ /min
Velocidad de flujo de lavado	0.25 a 0.5 gal/pie ³ /min inicialmente, para desplazar el regenerante 2.5 gal/pie ³ /min
Requerimiento del agua de lavado	40 a 90 gal/pie ³
Velocidad de flujo de servicio	2 a 5 gal/pie ³ /min

Al igual que con la resina catiónica de ácido fuerte, el retrolavado de la resina aniónica de base fuerte debe efectuarse a una velocidad de flujo que proporcione una expansión del lecho del 50% al 75% en un tiempo de 10 a 15 minutos.

4.4 GUIA DE DIMENSIONAMIENTO DE DESMINERALIZADORES

Tal y como se mencionó en la sección 4.3.1, el diseño para un desmineralizador, esta basado en la guía de dimensionamiento de sistemas desmineralizadores presentada por la Culligan Company de los Estados Unidos.

La información básica y los datos de esta guía, permiten predecir la capacidad y calidad del agua producida por un desmineralizador.

Todas las curvas de capacidad y fuga están basados para un agua con un contenido típico de minerales disueltos, como se muestra en el siguiente análisis.

La selección del tamaño y tipo apropiado de un desmineralizador para una aplicación específica, debe considerar los siguientes factores:

- 1) El uso esperado del agua
- 2) Calidad requerida del agua
- 3) Velocidad de flujo requerida (máximos y mínimos)
- 4) La capacidad (de galonaje) requerida entre regeneraciones
- 5) Limitaciones de operación de la unidad desmineralizadora

4.4.1 CALIDAD DEL AGUA

Las unidades catiónica y aniónica de base fuerte, son diseñadas para mantener una fuga de sodio y sílice mínima. Además remueven completamente el dióxido de carbono.

Todas las gráficas de capacidad y fuga tomadas de la guía de dimensionamiento para sistemas desmineralizadores, presentada por la Culligan Co. de los Estados Unidos, están basadas en un agua que tiene un contenido típico de minerales disueltos como el mostrado en el ejemplo de la tabla 4.7

TABLA 4.7 EJEMPLO DE ANALISIS FISICOQUIMICO PARA EL DIMENSIONAMIENTO DE SISTEMAS DESMINERALIZADORES PRESENTADO POR LA CULLIGAN CO. (Fuentes et al., 1994)

Los valores de gpg estan dados como CaCO ₃			
Cationes	Concentración	Aniones	Concentración
Calcio	5.0 gpg	Cloruros	2.4 gpg
Magnesio	2.4 gpg	Sulfatos	2.4 gpg
Sodio	2.5 gpg	Nitratos	0.2 gpg
Potasio	0.1 gpg	Hidróxido	0.0
		Carbonatos	0.0
		Bicarbonatos	5.0
Total de Cationes	10 gpg	Total de Aniones	10 gpg
Para calculos de desmineralización total:			
Sílice	0.5 gpg	Factor de carga Y	10.0
Dióxido de Carbono	0.9 gpg	Factor de carga Z	11.4

La tabla 4.8 muestra los parametros de calidad para un agua desmineralizada en base a las fugas esperadas por un sistema de base fuerte.

TABLA 4.8 DATOS DE CALIDAD PARA UN AGUA DESMINERALIZADA EN BASE A LAS FUGAS ESPERADAS (Fuentes et al., 1994)

Componentes	Dos lechos base fuerte
NaCl	0
NaOH	1-10
CO ₂	0
SiO ₂	usualmente 0.2 ppm
pH	7.5 - 10
Ohms	50,000 - 200,000

4.4.2 CAPACIDAD

Siguiendo los pasos adecuados, y usando los gráficos apropiados, puede diseñarse fácilmente una unidad para cualquier situación dada.

Una vez que las capacidades por pie³ han sido determinadas, debe realizarse un nuevo cálculo basado en los pie³ totales de la resina de la nueva unidad.

Los cálculos de la capacidad y la fuga deben hacerse para el agua cruda que se desea tratar, ya que para cada tipo de agua se tendrán capacidades diferentes.

4.4.3 CALCULOS PARA LA UNIDAD CATIONICA

Para la determinación de la capacidad de operación de una unidad catiónica, se utiliza el siguiente procedimiento:

- A) A partir del análisis de agua cruda, se determina el porcentaje de sodio y alcalinidad.

$$\% \text{sodio} = \frac{\text{gpg sodio} + \text{gpg potasio}}{\text{Factor de carga Y}} \times 100 \quad \text{Ec.4.8}$$

$$\% \text{Alcalinidad} = \frac{\text{gpg carbonato} + \text{gpg bicarbonato}}{\text{Factor de carga Y}} \times 100 \quad \text{Ec. 4.9}$$

donde los factores de carga para el cálculo de las capacidades de intercambio de la resina catiónica de ácido fuerte y aniónica de base fuerte son:

$$\text{Factor Y} = \sum \text{Cationes} \quad \text{Ec. 4.10}$$

$$\text{Factor Z} = \sum \text{Aniones} + \text{gpg SiO}_2 + \text{gpg CO}_2 \quad \text{Ec. 4.11}$$

- B) Utilizando la gráfica 4.1, se ubica en la parte inferior del gráfico, eje de las X, el porcentaje de sodio calculado y moviéndose verticalmente hacia arriba se intercepta la curva del porcentaje de alcalinidad. Continuando horizontalmente hacia la izquierda del gráfico se lee, en el eje de las Y, la capacidad en kilogramos por pie³ de resina.

En la mayoría de los casos será necesario interpolar entre las curvas de % de alcalinidad. La interpolación es estimada entre dos curvas dadas.

- C) Multiplicar la capacidad calculada del catión por el número total de pie³ de la unidad escogida para determinar la capacidad total de la resina catiónica en granos como CaCO₃.

$$\text{Capacidad total} = \text{Capacidad unidad} \times \text{Volumen de resina} \quad \text{Ec. 4.12}$$

- D) Dividir la capacidad de la unidad (en granos) por el factor de carga "Y" para encontrar la capacidad en galones.

$$\text{Galones} = \frac{\text{Capacidad total}}{\text{Factor de carga Y}} \quad \text{EC.4.13}$$

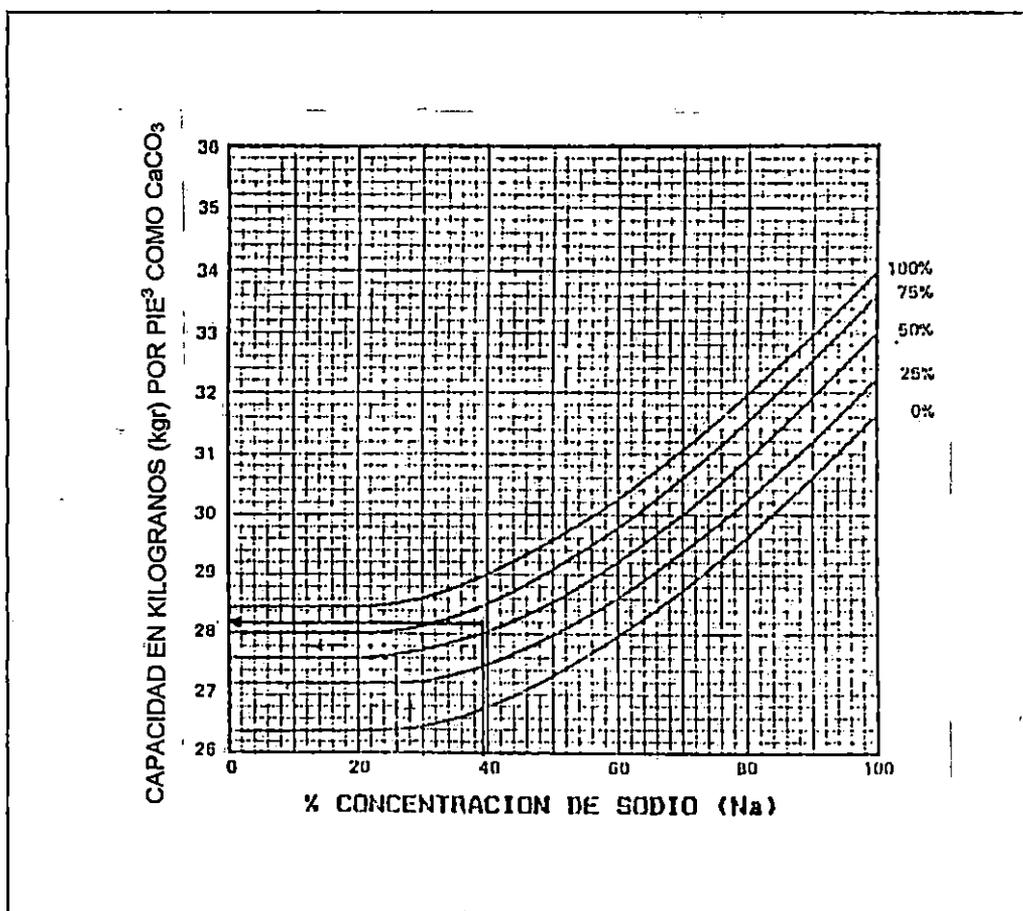


FIGURA 4.1 CAPACIDAD DE RESINA CATIONICA (Fuentes et al., 1994)

Esta es la capacidad total de la unidad. Pero debido a que la unidad catiónica suministra el agua para la regeneración de la unidad aniónica, esa agua de regeneración no está disponible para el ciclo de servicio y debe ser descontada. Por cada pie³ de resina aniónica, descontar 100 galones de agua decaionizada. La capacidad catiónica neta disponible en galones será:

$$\text{Galonaje neto} = \text{Capacidad en galones} - (\text{pie}^3 \text{ de resina} \times 100 \text{ gal}) \quad \text{Ec.4.14}$$

La calidad del efluente de un desmineralizador de dos lechos es función de la fuga de la unidad catiónica. La fuga del catión está influenciada por la concentración de sodio, la concentración de la alcalinidad y el nivel de regeneración. La fuga de los desmineralizadores será considerada solamente en el nivel de regeneración estándar de 9 lbs de HCl.

Para determinar la fuga de sodio desde la unidad catiónica, se utiliza el siguiente procedimiento:

- A) El % de sodio y alcalinidad fueron determinados en el paso A del cálculo de la unidad catiónica
- B) Utilizando el gráfico 4-2 se localiza el porcentaje de sodio en la parte inferior del gráfico, siguiendo verticalmente hacia arriba se intercepta la curva del porcentaje de alcalinidad (interpolando si es necesario). Continuando horizontalmente hacia la izquierda del gráfico lea el promedio de fuga de sodio como porcentaje de cationes totales.
- C) Multiplique el factor de carga Y por el porcentaje de cationes totales y por 17.1 para determinar la fuga de sodio (en ppm) de la unidad catiónica.

$$\text{Fuga de sodio} = \frac{\% \text{ cationes}}{100} \times \text{Factor de carga Y} \times 17.1 \quad \text{Ec.4.15}$$

Aunque el gráfico de fuga indique la posibilidad de un 0% de fuga, rara vez se produce menos de 1ppm. Por lo tanto nunca debe usarse menos de 1ppm de fuga de sodio calculada, además se redondea el valor de fuga de sodio al próximo mayor a 1ppm. (Fuentes et al., 1994).

4.4.4 CALCULOS PARA LA UNIDAD ANIONICA

Las capacidades aniónicas de base fuerte en esta guía de dimensionamiento están basadas en punto final donde la fuga de sílice ha incrementado a 0.3 ppm sobre su promedio de fuga calculado.

La capacidad de la resina esta directamente relacionada con la completa remoción de sílice desde el lecho de resina durante la regeneración. La remoción de sílice y la fuga de sílice son influenciadas por la dosis de regeneración, temperatura y tiempo de contacto. cuando la sílice es superior al 10% se utiliza un nivel regeneración de 8 lbs de NaOH/pie³.

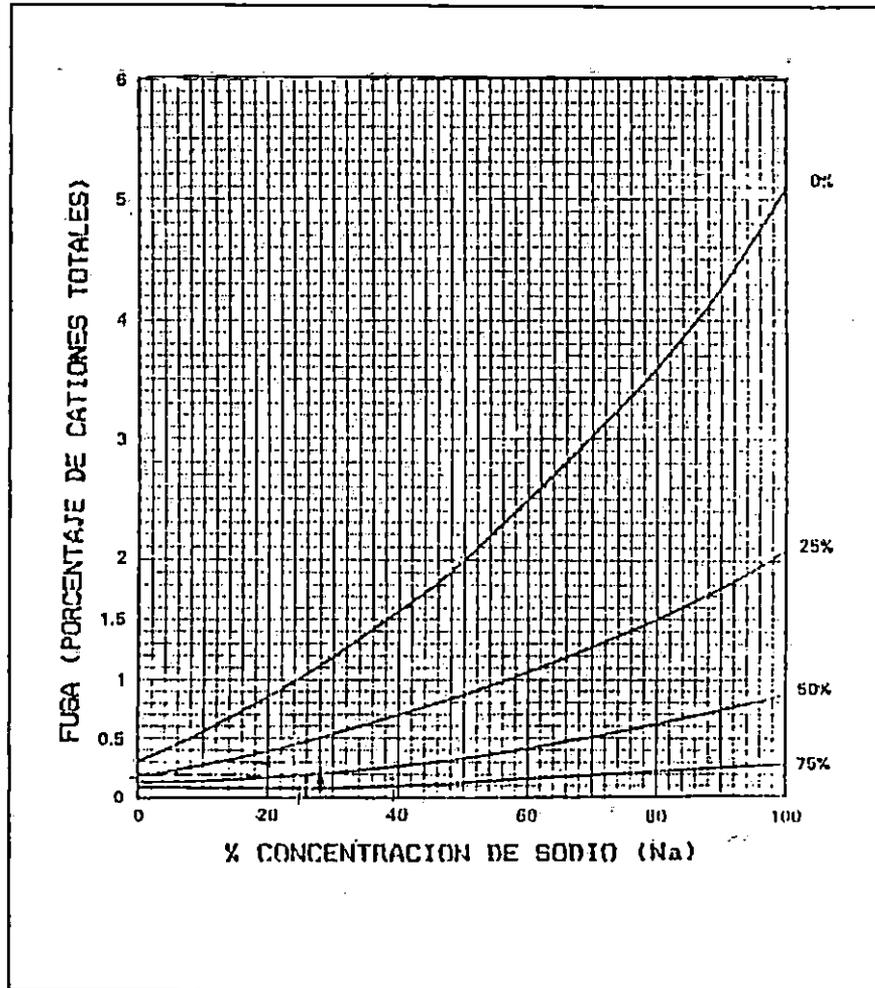


FIGURA 4.2 FUGA DE SODIO (Fuentes et al.,1994)

Para determinar la capacidad de la unidad aniónica fuertemente básica del desmineralizador, utilice los siguientes pasos:

- A) Del análisis del agua cruda de la guía de dimensionamiento encuentre el porcentaje de ácido carbónico, el porcentaje de iones monovalentes, y el % de sílice en relación a los aniones totales intercambiables (factor de carga Z). Así:

$$\% \text{Acido carbonico} = \frac{\text{gpg CO}_3^{2-} + \text{gpg HCO}_3^- + \text{gpg CO}_2}{\text{Factor de carga Z}} \quad \text{Ec.4.16}$$

$$\% \text{iones monovalentes} = \frac{\text{gpg cloruro} + \text{gpg nitrato}}{\text{Factor de carga Z}} \times 100 \quad \text{Ec.4.17}$$

$$\% \text{silice} = \frac{\text{gpg silice}}{\text{Factor de carga Z}} \times 100 \quad \text{EC.4.18}$$

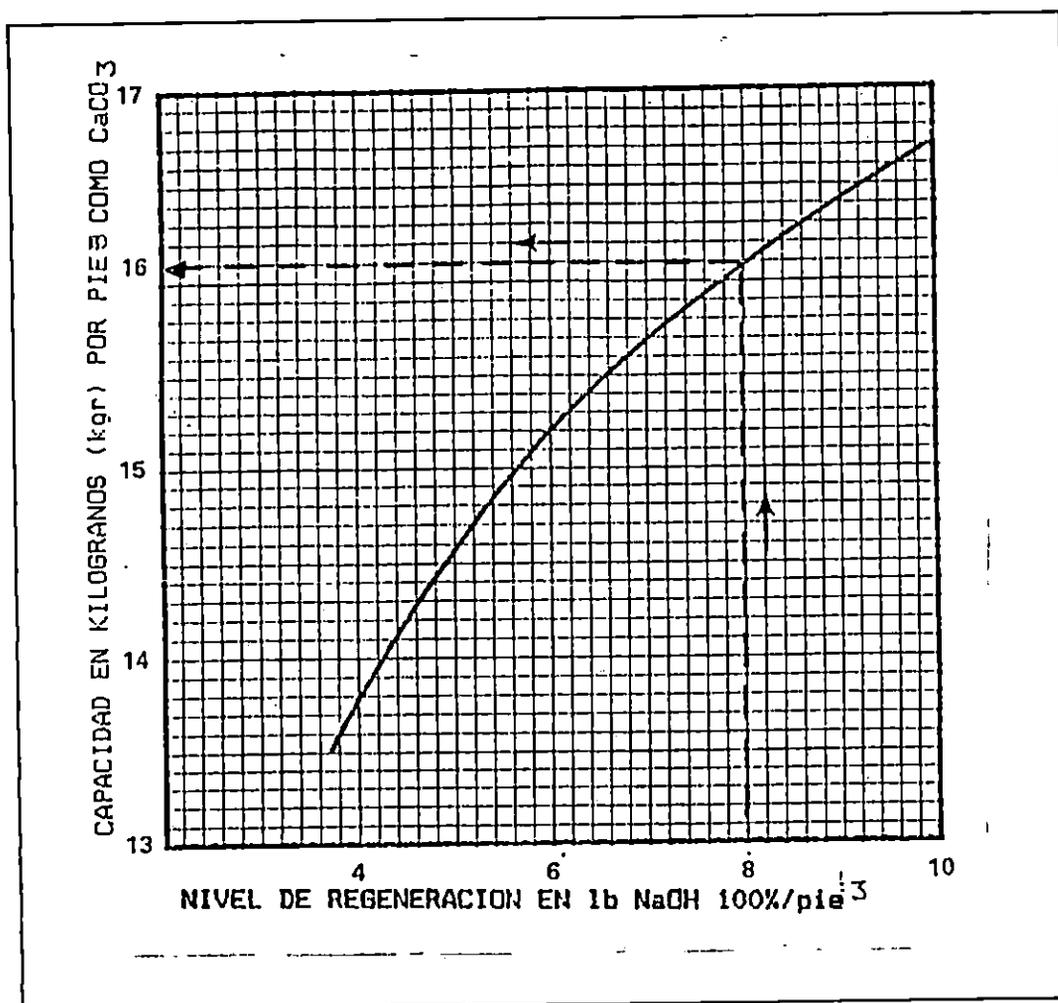
- B) Utilizando el nivel de regeneración deseado (nivel estándar de 8 lbs de NaOH/pie³ de resina) se encuentra la capacidad del anión por pie³ de resina en el gráfico 4.3. Ubicando las 8lbs del nivel de regeneración (a menos que otro nivel sea elegido) en el eje de las X, moviéndose verticalmente hacia arriba hasta interceptar la curva, y desplazándose horizontalmente hacia la izquierda del gráfico lea en el eje de las Y el valor de la capacidad de intercambio. Los valores de corrección se seleccionan de la tabla 4.5, para %ácido carbónico, %iones monovalentes y % de sílice. Multiplicando la capacidad por cada factor de corrección se tiene:

$$\text{Capacidad corregida} = \text{Capacidad} \times F_1 \times F_2 \times F_3 \quad \text{Ec.4.19}$$

Divida la capacidad de la unidad por el factor de carga Z para encontrar el volumen de agua desmineralizada por ciclo de regeneración.

$$\text{Galones} = \frac{\text{Capacidad (gr)}}{\text{Factor de carga Z}} \quad \text{Ec.4.20}$$

- D) El volumen de agua producida por un desmineralizador de lecho doble se registrará siempre por la resina aniónica quien es la que se agota primero.



GRAFICA 4.3 NIVELES DE REGENERACION (Fuentes et al., 1994)

4.5 DETERMINACION DE FUGA DE SILICE

La fuga desde un desmineralizador de lecho doble consistirá de hidróxido de sodio y sílice. Donde el sodio proveniente de la unidad catiónica es convertido a hidróxido de sodio en la unidad aniónica de base fuerte. Por lo tanto la cantidad de hidróxido de sodio en el agua desmineralizada es igual a la fuga de sodio desde la unidad catiónica.

La resistencia específica de esta agua reflejará la presencia de hidróxido de sodio solamente después que todo el ácido carbónico (dióxido de carbono) sea removido por la resina aniónica de base fuerte (véase gráfico 4.4). Para encontrar la conductividad específica causada por el hidróxido de sodio, siga los pasos siguientes:

TABLA 4.9 FACTORES DE CORRECCION PARA LA CAPACIDAD DE RESINA ANIONICA BASE FUERTE

ACIDO CARBONICO		IONES MONOVALENTES		% DE SILICE			
% H ₂ CO ₃	F ₁	% IONES	F ₂	4 lbs NaOH	6 lbs NaOH	8 lbs NaOH	F ₃
0-2	1.00	0.6	0.99	0-1.3	0-4	0-8	0.99
2-4	1.01	6-12	0.98	1.3-2.5	4-8	8-14	0.98
4-6	1.02	12-20	0.97	2.5-4	8-11	14-20	0.97
6-8	1.03	20-30	0.96	4-5	11-15	20-25	0.96
8-10	1.04	30-40	0.95	5-7	15-19	25-31	0.95
10-13	1.05	40-53	0.94	7-9	19-22	31-36	0.94
13-17	1.06	53-65	0.93	9-10.5	22-25	36-41	0.93
17-20	1.07	65-80	0.92	10.5-12	25-29	41-45	0.92
20-25	1.08	80-100	0.91	12-14	29-32	45-50	0.91
25-30	1.09			14-17	32-35		0.90
30-36	1.10			17-19	35-39		0.89
36-44	1.11			19-21	39-42		0.88
44-54	1.12			21-24	42-45		0.87
54-65	1.13			24-26	45-50		0.86
65-78	1.14			26-29			0.85
78-100	1.15			29-32			0.84
				32-35			0.83
				35-38			0.82
				38-42			0.81
				42-44			0.80

Ubique la concentración (ppm) de fuga de sodio en el eje de las X del gráfico 4.4, diríjase verticalmente hacia arriba e intercepte la línea para sistemas fuertemente básicos y luego desplácese horizontalmente hacia la izquierda y lea la conductividad específica.

La fuga de sílice esta directamente relacionada con la cantidad de sílice que permanece en la resina de la regeneración anterior y la fuga de sodio desde la unidad catiónica.

Para determinar la fuga de sílice desde el intercambiador aniónico fuertemente básico, se utiliza el siguiente procedimiento:

- A) Encuentre el porcentaje de sílice y las ppm de fuga de sodio (ambas encontradas en los procedimientos descritos anteriormente).

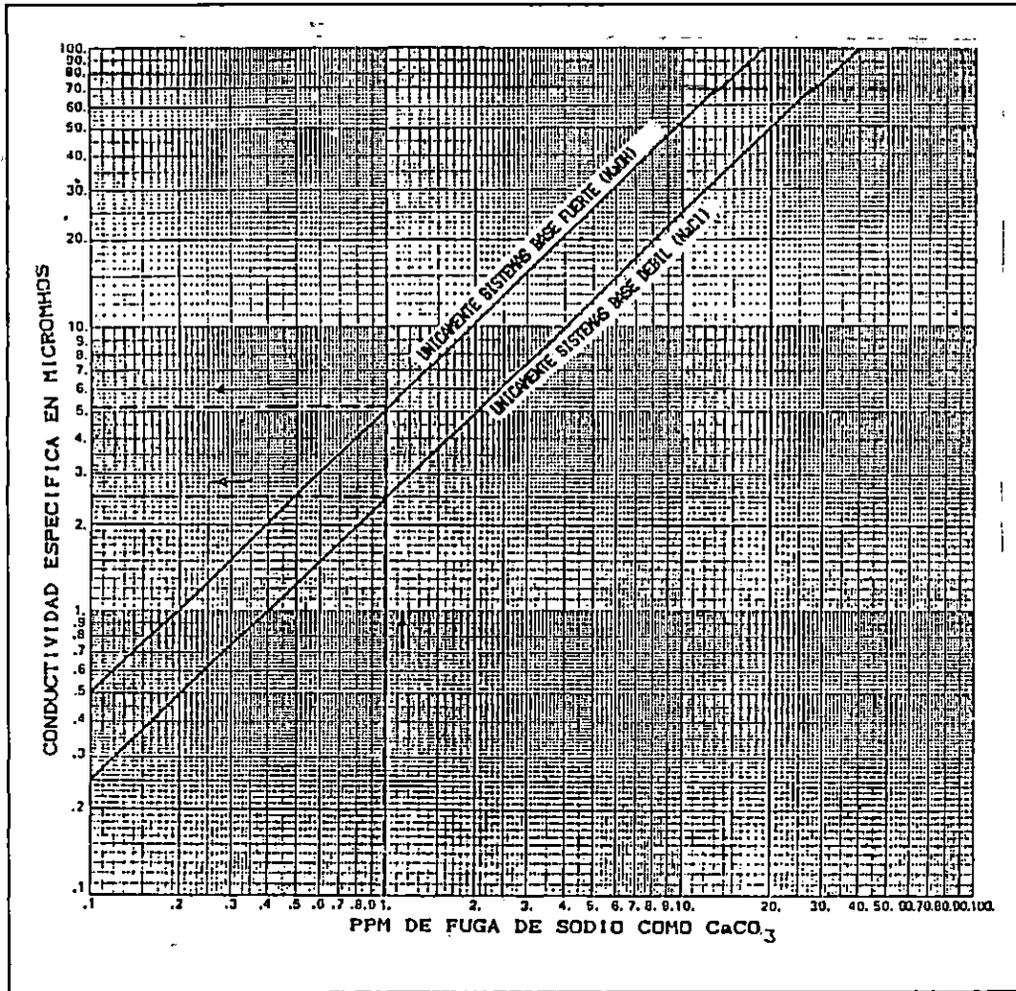
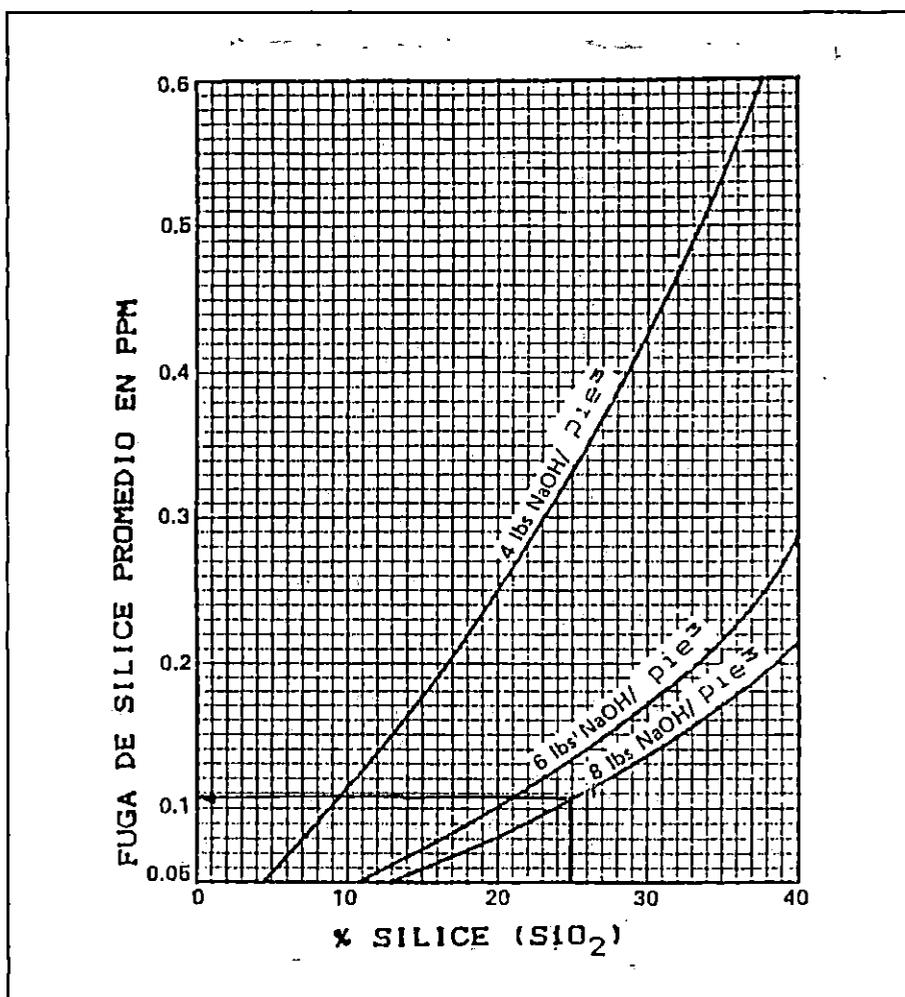


GRAFICO 4.4 CONDUCTIVIDAD ESPECIFICA (Fuentes et al., 1994)

- B) Utilizando la gráfica 4.5, encuentre el valor aproximado de fuga de sílice. Ubique el % de sílice en el eje de las X, luego moviéndose verticalmente hacia arriba hasta interceptar la curva del nivel de regeneración utilizado, luego desplazándose horizontalmente hacia la izquierda lea el promedio de fuga de sílice. En los casos donde el % de sílice no puede interceptar la curva del nivel de regeneración, use un promedio de fuga mínimo de 0.05 ppm.
- C) El factor de corrección de la fuga de sílice se encuentra en el gráfico 4.6. Ubicando las ppm de fuga de sodio en el eje de las X desplácese verticalmente hacia arriba hasta interceptar con la curva del nivel de regeneración elegido, luego desplácese horizontalmente hacia la izquierda para encontrar el factor de corrección.



GRAFICA 4.5 FUGA DE SILICE (Fuentes et al., 1994)

Divida el promedio de fuga de sílice por el factor de corrección para llegar a la fuga de sílice corregida.

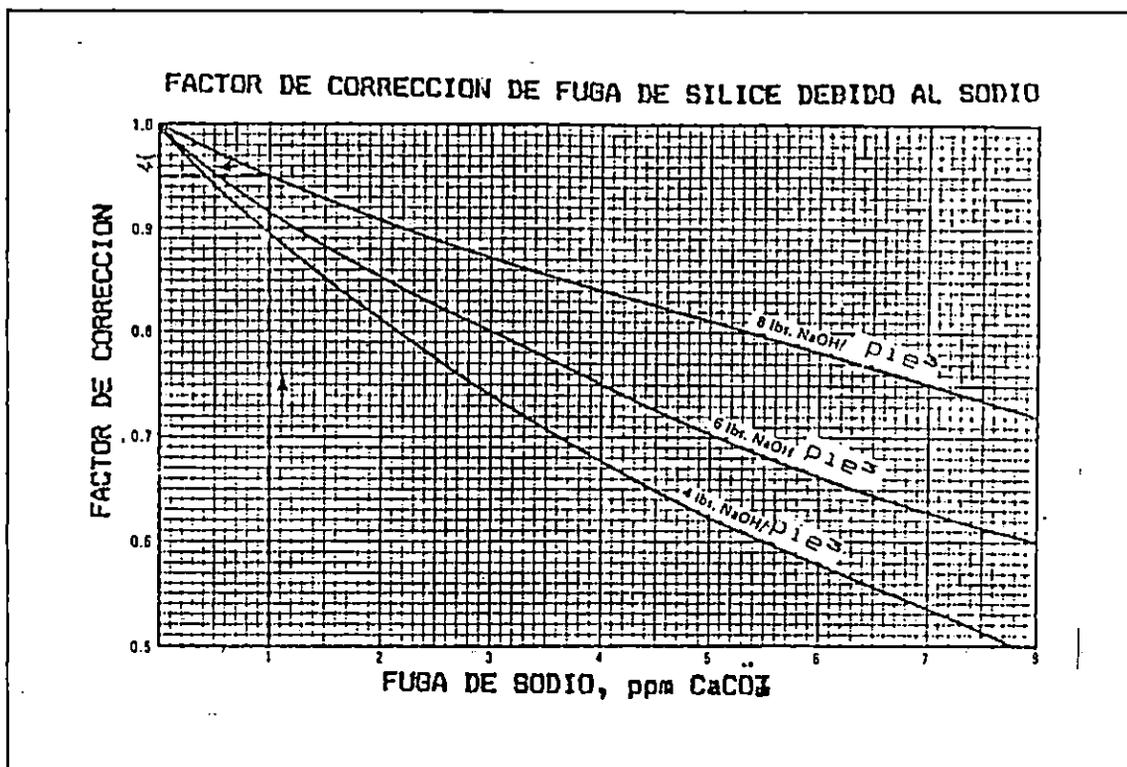
$$\text{Fuga corregida} = \frac{\text{Promedio de fuga de sílice}}{\text{Factor de correccion}} \quad \text{Ec.4.21}$$

Para corregir el pH del efluente final utilice la siguiente tabla:

TABLA 4.10 CORRECCION DEL pH (Fuentes et al., 1994)

ppm fuga de sodio	1	2	3	4	5	6	7	8	9	10
pH	7.7	8.2	8.6	9.0	9.3	9.6	9.8	10.0	10.2	10.3

Utilice la fuga de sodio encontrada en la unidad catiónica y lea el pH aproximado en la tabla



GRAFICA 4.6 FACTOR DE CORRECCION DE SILICE DEBIDO AL SODIO
(Fuentes et al., 1994)

4.6 EJEMPLO DE CALCULO

Se determinará el volumen o cantidad de resina (aniónica y catiónica) necesaria para producir 100 galones de agua desmineralizada, de un agua cruda suministrada por ANDA, además se deberá hacer la corrección de este volumen (si fuese necesario) con el objeto de adecuarlo al tamaño de tanques para resina disponibles en el mercado. Finalmente debe calcularse el nuevo galonaje que teóricamente se obtendría con el nuevo volumen de resina.

Los datos necesarios para el cálculo se presentan en la tabla 4.8, los cuales corresponden a un promedio del análisis físico-químico del agua cruda suministrada a la Planta Piloto

TABLA 4.11 ANALISIS FISICOQUIMICO DEL AGUA CRUDA SUMINISTRADA POR LA ANDA A LAS INSTALACIONES DE LA PLANTA PILOTO. DE LA UNIVERSIDAD DE EL SALVADOR (Fuentes et al., 1994)

CATIONES	ppm CaCO ₃	gpg CaCO ₃	ANIONES	ppm CaCO ₃	gpg CaCO ₃
Calcio	122.87	7.16	Cloruros	50.67	2.95
Magnesio	141.7	8.26	Sulfatos	66.27	3.86
Sodio	157.87	9.21	Nitratos	11.53	0.67
Potasio	13.07	0.76	Bicarbonato	275.93	16.09
Total de Cationes	435.51	25.39	Total de Aniones	404.40	23.57
Para calculos de desmineralización total:					
Sílice: 159.57 ppm CaCO ₃ 9.3 gpg CaCO ₃			Dióxido de Carbono: 75.67 ppm CaCO ₃ 4.41 gpg CaCO ₃		

4.6.1 PROCEDIMIENTO PARA EL CALCULO DE UN DESMINERALIZADOR

El primer paso que debe realizarse para poder especificar un desmineralizador es determinar las características (o calidad) del agua que ha de someterse al proceso (agua cruda). Este último paso debe ser desarrollado por el usuario o en su defecto que posea la información necesaria, a fin de poder desarrollar los calculos requeridos.

Los factores de carga se calculan utilizando las ecuaciones 4.10 y 4.11.

$$\text{Factor Y} = \sum \text{Cationes}$$

$$\text{Factor Y} = (7.16 + 8.26 + 9.21 + 0.76) \text{ gpg CaCO}_3$$

$$\text{Factor Y} = 25.39 \text{ gpg CaCO}_3$$

$$\text{Factor Z} = \sum \text{Aniones} + \text{gpg SiO}_2 + \text{gpg CO}_2$$

$$\text{Factor Z} = (23.57 + 9.3 + 4.41) \text{ gpg CaCO}_3$$

$$\text{Factor Z} = 37.28 \text{ gpg CaCO}_3$$

El usuario debe tener presente que en todos los cálculos especificados en la guía de dimensionamiento, la concentración de todas las especies químicas involucradas tienen que estar expresadas en granos por galón como CaCO_3 .

4.6.1.1 DETERMINACIÓN DE UNIDAD CATIONICA

- a) Para determinar la capacidad de operación de la unidad catiónica es necesario conocer el % de Sodio y % de Alcalinidad. Utilizando las ecuaciones 4.8 y 4.9, se tiene:

$$\% \text{sodio} = \frac{\text{gpg sodio} + \text{gpg potasio}}{\text{Factor de carga Y}} \times 100$$

$$\% \text{ sodio} = (9.21+0.76)/25.39 \times 100$$

$$\% \text{ sodio} = 39.26$$

$$\% \text{ Alcalinidad} = \frac{\text{gpg carbonato} + \text{gpg bicarbonato}}{\text{Factor de carga Y}} \times 100$$

$$\% \text{ Alcalinidad} = (0 + 16.09)/25.39 \times 100$$

$$\% \text{ Alcalinidad} = 63.37$$

- b) Utilizando la gráfica 4.1, se ubica el valor del % Sodio en el eje de las X, luego se sigue verticalmente hasta interceptar la curva del %Alcalinidad y finalmente, siguiendo una línea recta hasta el eje de las Y, se encuentra la capacidad en kilogramos como $\text{CaCO}_3/\text{pie}^3$ de resina catiónica. En la mayoría de los casos será necesario interpolar entre dos curvas del %Alcalinidad.

De acuerdo a lo anterior se tiene:

$$\text{Capacidad de la resina catiónica} = 28.3 \text{ kgr } \text{CaCO}_3/\text{pie}^3$$

Como se tomo una base de 100 galones de agua a tratar, el volumen de resina necesario sera:

$$\text{Volumen de resina catiónica} = 100 \text{ gal} \times 25.39 \text{ gr/gal} \times 1 \text{ pie}^3/28,300 \text{ gr}$$

$$\text{Volumen de resina} = 0.09 \text{ pie}^3 \approx 0.1 \text{ pie}^3$$

4.6.1.2 DETERMINACIÓN DE UNIDAD ANIÓNICA

Para el cálculo de la capacidad de la resina aniónica, también se hará uso del nivel de regeneración máximo recomendado (8 lbs de NaOH 100% /pie³ de resina).

De acuerdo a la guía de dimensionamiento, los parámetros requeridos para el cálculo de la capacidad son el % de ácido carbónico, % iones monovalentes y el % de sílice. Para el cálculo de estos porcentajes se hace uso de las ecuaciones 4.16, 4.17 y 4.18, respectivamente.

$$\% \text{Acido carbonico} = \frac{\text{gpg CO}_3^{2-} + \text{gpg HCO}_3^- + \text{gpg CO}_2}{\text{Factor de carga Z}}$$

$$\begin{aligned} \% \text{Acido carbonico} &= (0+16.09+4.41)/37.28 \times 100 \\ &= 54.99 \end{aligned}$$

$$\% \text{iones monovalentes} = \frac{\text{gpg cloruro} + \text{gpg nitrato}}{\text{Factor de carga Z}} \times 100$$

$$\begin{aligned} \% \text{iones monovalentes} &= (2.95+0.67)/37.28 \times 100 \\ &= 9.7 \end{aligned}$$

$$\% \text{silice} = \frac{\text{gpg silice}}{\text{Factor de carga Z}} \times 100$$

$$\begin{aligned} \% \text{silice} &= (9.3/37.28) \times 100 \\ &= 24.95 \end{aligned}$$

Luego utilizando el gráfico 4.3, se ubica el nivel de regeneración deseado (8 lbs de NaOH 100% /pie³) en el eje de las X y se lee verticalmente hacia arriba hasta interceptar la curva y finalmente se proyecta una línea recta hacia el eje de las y para encontrar la capacidad de la resina aniónica.

Siguiendo el procedimiento anterior se obtiene la siguiente capacidad:

Capacidad de la resina aniónica = 16 kgr CaCO_3 /pie³

Esta capacidad obtenida debe ser corregida en base a los porcentajes de ácido carbónico, iones monovalentes y sílice, utilizando los valores proporcionados por la tabla 4.7, así:

54.18% de Acido Carbónico : $F_1 = 1.13$

10.02% de iones monovalentes : $F_2 = 0.98$

25.26% Sílice y 8 lbs de NaOH 100% : $F_3 = 0.96$

La capacidad corregida se obtiene por la ecuación 4.19, multiplicando la capacidad inicial por cada uno de los factores de corrección.

Capacidad corregida: $16,000 \times 1.13 \times 0.98 \times 0.96$

Capacidad corregida : 17,009.7 granos/pie³

Finalmente el volumen requerido de resina para tratar los 100 galones de agua será:

Volumen de resina aniónica = $100 \text{ gal} \times 37.28 \text{ gr/gal} \times 1 \text{ pie}^3 / 17,009.7 \text{ gr}$

Volumen de resina aniónica = 0.22 pie³

De donde puede observarse que el volumen de resina aniónica, para tratar un mismo galonaje de agua cruda, es mayor que el volumen de resina catiónica. Esto se debe a que la capacidad de intercambio de la resina aniónica es menor que la capacidad de intercambio de la resina catiónica. En parte también se debe a que el factor de carga Z es mayor que el factor de carga Y.

El volumen de resina obtenido es relativamente pequeño, por lo que se debe adecuar a un tanque comercial de tamaño inmediatamente superior, además por consideraciones hidráulicas, no es posible manejar volúmenes diferentes de resinas en un sistema desmineralizador, razón por la cual deben seleccionarse del mismo tamaño.

Tomando un tamaño de tanque comercial de 0.5 pie³, se diseña el sistema de manera que cuente con 0.5 pie³ de resina catiónica de ácido fuerte y 0.5 pie³ de resina aniónica de base fuerte.

El galonaje que producirán 0.5 pie³ de resina catiónica es:

$$\begin{aligned}\text{Gal agua decationizada} &= (\text{capac. de resina}) \times (\text{vol. de resina}) / \text{Factor de carga Y} \\ &= 28,300 \text{ gr/pie}^3 \times 0.5 \text{ pie}^3 / 25.39 \text{ gr/pie}^3 \\ &= 557\end{aligned}$$

El volumen calculado corresponde al galonaje total de agua decationizada que teóricamente producirá la resina catiónica por ciclo de regeneración. Sin embargo, dado que la resina aniónica se regenera con agua decationizada en una relación de 100 galones de agua decationizada por pie³ de resina aniónica, por lo que el galonaje neto será:

$$\text{Galonaje neto} = 557 \text{ gal} - 100 \text{ gal/pie}^3 \times 0.5 \text{ pie}^3$$

$$\text{Galonaje neto} = 507$$

Como también se utilizó 0.5 pie³ de resina aniónica. Los galones de agua producidos por este volumen de resina aniónica será:

$$\text{Gal unidad aniónica} = 17,009.7 \text{ gr/pie}^3 \times 0.5 \text{ pie}^3 / 37.28 \text{ gr/gal}$$

$$\text{Galones de agua desmineralizada} = 228$$

Este volumen obtenido estará libre de aniones y cationes dado que pasa a través del lecho aniónico es agua decationizada, resultando un efluente de agua desmineralizada.

4.7 PLANTEAMIENTO DE LO QUE APORTARA LA CODIFICACION DEL CASO

A continuación se presenta una guía para el dimensionamiento de sistemas desmineralizadores, presentada por la Culligan Co. de los Estados Unidos.

La información generada en la presente sección es muy útil en el cálculo de las capacidades para la resina aniónica y catiónica, así como también en la predicción de la calidad de agua producida por un desmineralizador. Los desmineralizadores deben ser diseñados de manera que su regeneración no exceda de una vez por día.

El procedimiento para la determinación de la capacidad de la unidad de desmineralización se presenta a continuación:

Todas las ecuaciones (modelos) generados a partir de las curvas presentes en el capítulo 4, han sido desarrolladas para aguas con un contenido típico de minerales disueltos como el mostrado en la tabla 4.8

En todos los cálculos especificados en la guía de dimensionamiento, la concentración de todas las especies químicas esta expresada en granos por galón (gpg) como CaCO_3 .

4.7.1 SOLUCION DE MODELOS

De acuerdo a la guía de dimensionamiento presentada en la sección 4.4, los factores de carga se calculan por las ecuaciones 4.10 y 4.11, respectivamente

$$\text{Factor Y} = \sum \text{Cationes}$$

$$\text{Factor Z} = \sum \text{Aniones} + \text{gpg SiO}_2 + \text{gpg CO}_2$$

4.7.2 CALCULO PARA LA UNIDAD CATIONICA

Los parámetros que determinan la capacidad de operación de la unidad catiónica son el % de sodio y el % de alcalinidad, calculados por las ecuaciones 4.8 y 4.9 descritas en la sección 4.3.

$$\% \text{sodio} = \frac{\text{gpg sodio} + \text{gpg potasio}}{\text{Factor de carga Y}} \times 100$$

$$\% \text{Alcalinidad} = \frac{\text{gpg carbonato} + \text{gpg bicarbonato}}{\text{Factor de carga Y}} \times 100$$

Del análisis de regresión practicado a los puntos leídos del grafico 4.1 se obtuvieron los siguientes modelos, según el nivel de regeneración elegido, así:

0% de Alcalinidad

La gráfica de 0% de alcalinidad se seccionó en tres partes, así:

para una concentración de sodio de 0-24%, se observa un valor constante,

$$Y = 26.4 \quad \text{Ec.4.22}$$

b. para una concentración de sodio de 24-60%, se selecciono la siguiente ecuación:

$$Y = a + bX + cX^3 + \frac{d}{X^2} \quad \text{Ec.4.23}$$

en donde:

Y = Capacidad de la resina cationica en Kgr / pie³ como CaCO₃

X = % de concentracion de sodio

$$a = 25.238214446$$

$$b = 0.027806796$$

$$c = 4.59484 \times 10^{-6}$$

$$d = 247.6672681$$

La gráfica que representa los datos reales correlacionados por la ecuación 4.23 se muestra en la figura 4.7

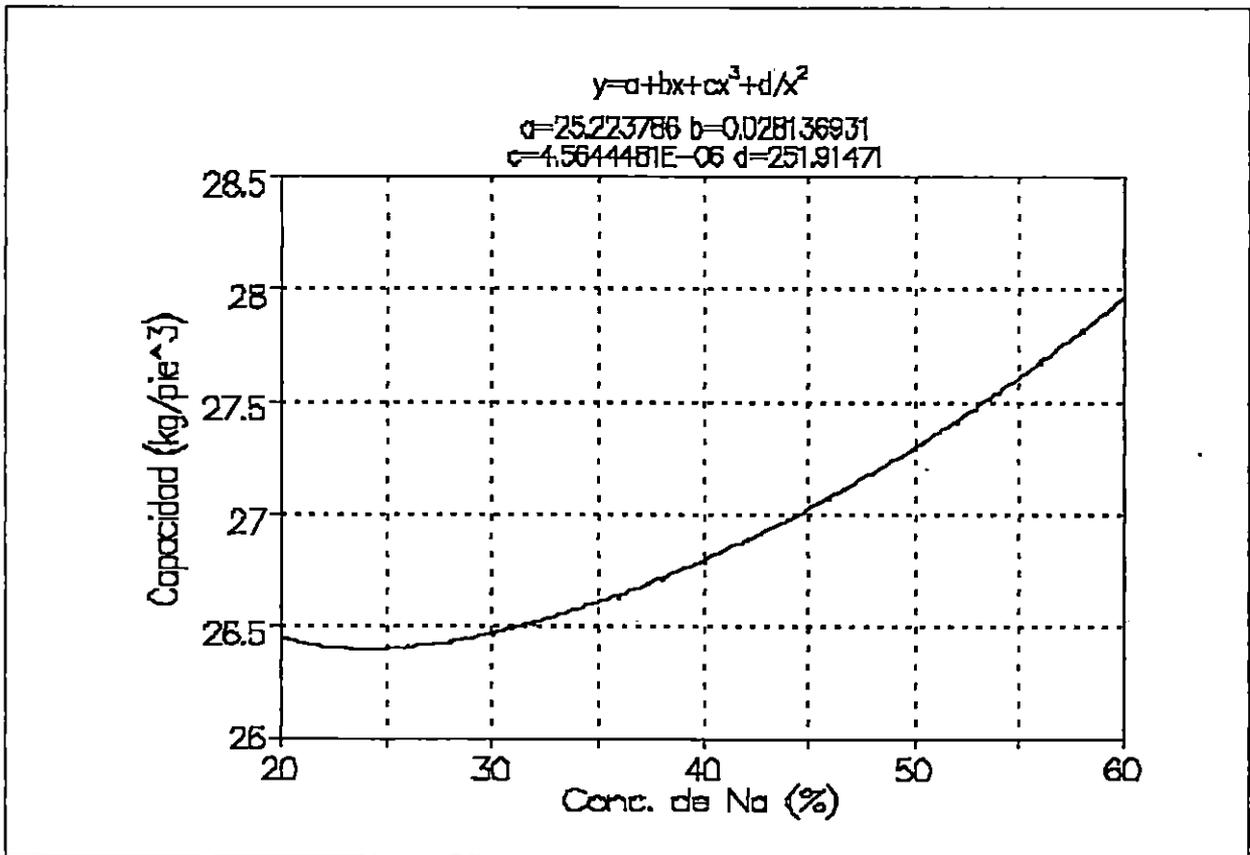


FIGURA 4.7 CAPACIDAD DE LA RESINA CATIONICA A 0% DE ALCALINIDAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 24-60%

c. Para una concentración de sodio de 60-100%, se obtuvo la siguiente ecuación:

$$Y = a + bX + cX^2 + \frac{d}{X} + \frac{e}{X^2} \quad \text{Ec.4.24}$$

Donde:

$$a = -71.822295$$

$$b = 0.88683521$$

$$c = -0.0024141135$$

$$d = 4752.6507$$

$$e = -85180.761$$

La gráfica que representa los datos reales correlacionados por la ecuación 4.24 se muestra en la figura 4.8.

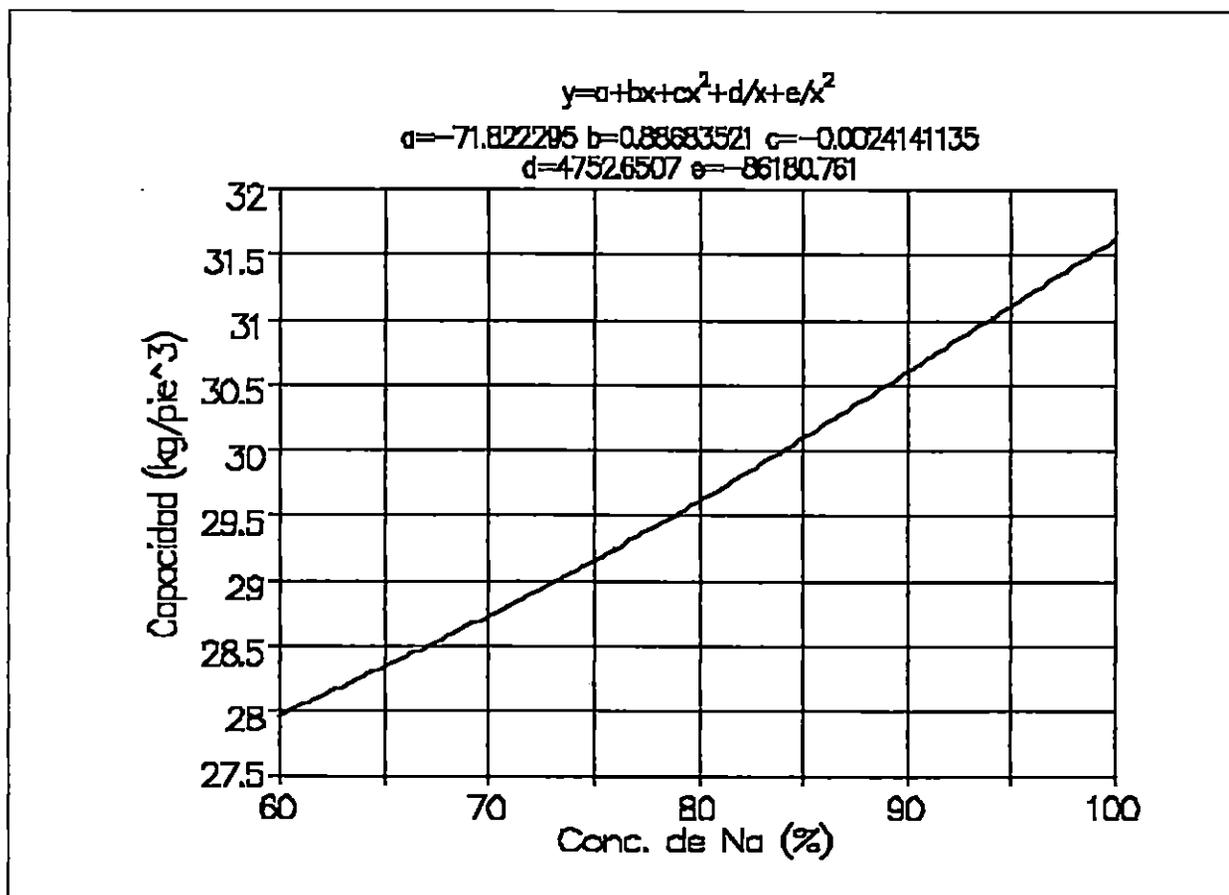


FIGURA 4.8 CAPACIDAD DELA RESINA CATIONICA A 0% DE ALCALINIDAD PARA UN RANGO DE CONCENTRACION DE SODIO DEL 60-100%.

25% de Alcalinidad

La gráfica 4.1 para una alcalinidad del 25%, se seccionó en dos partes.

- a. Para una concentración de sodio de 0-24%, se presenta un valor constante

$$Y = 27.2 \quad \text{Ec.4.25}$$

- b. Para un rango de concentración de sodio de 24-100% la ecuación obtenida es la siguiente,

$$Y = a + bX + cX^2 + \frac{d}{X} \quad \text{Ec.4.26}$$

Donde:

$$a = 24.49623135$$

$$b = 0.34378918$$

$$c = 0.000391892$$

$$d = 40.17298637$$

La gráfica que representa los datos reales correlacionados por la ecuación 4.27 se muestra en la figura 4.9

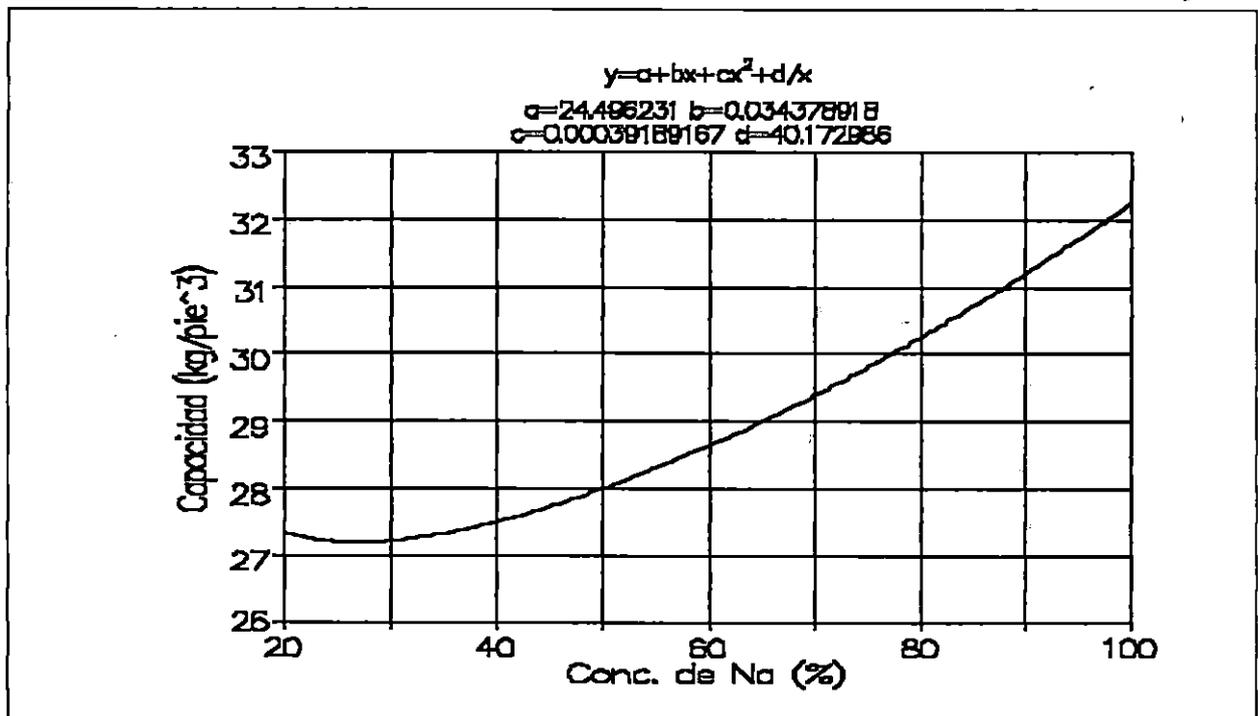


FIGURA 4.9 CAPACIDAD DE LA RESINA CATIONICA A 25% DE ALCALINIDAD PARA UN RANGO DE 0-24% DE SODIO

50% de Alcalinidad

La gráfica 4.1 para una alcalinidad del 50%, se seccionó en dos partes:

- a. Para un rango de concentración de sodio de 0-20% el valor es constante

$$Y = 28 \quad \text{Ec.4.27}$$

- b. Para el rango de concentración de sodio de 20-100% se obtuvo la ecuación:

$$Y = a + bx + cX\sqrt{X} + \frac{d}{X^2} \quad \text{Ec.4.28}$$

Donde:

$$a = 28.36890088$$

$$b = -0.10008536$$

$$c = 0.014745126$$

$$d = -34.4221078$$

La figura 4.10 muestra los datos reales correlacionados por la ecuación 4.27

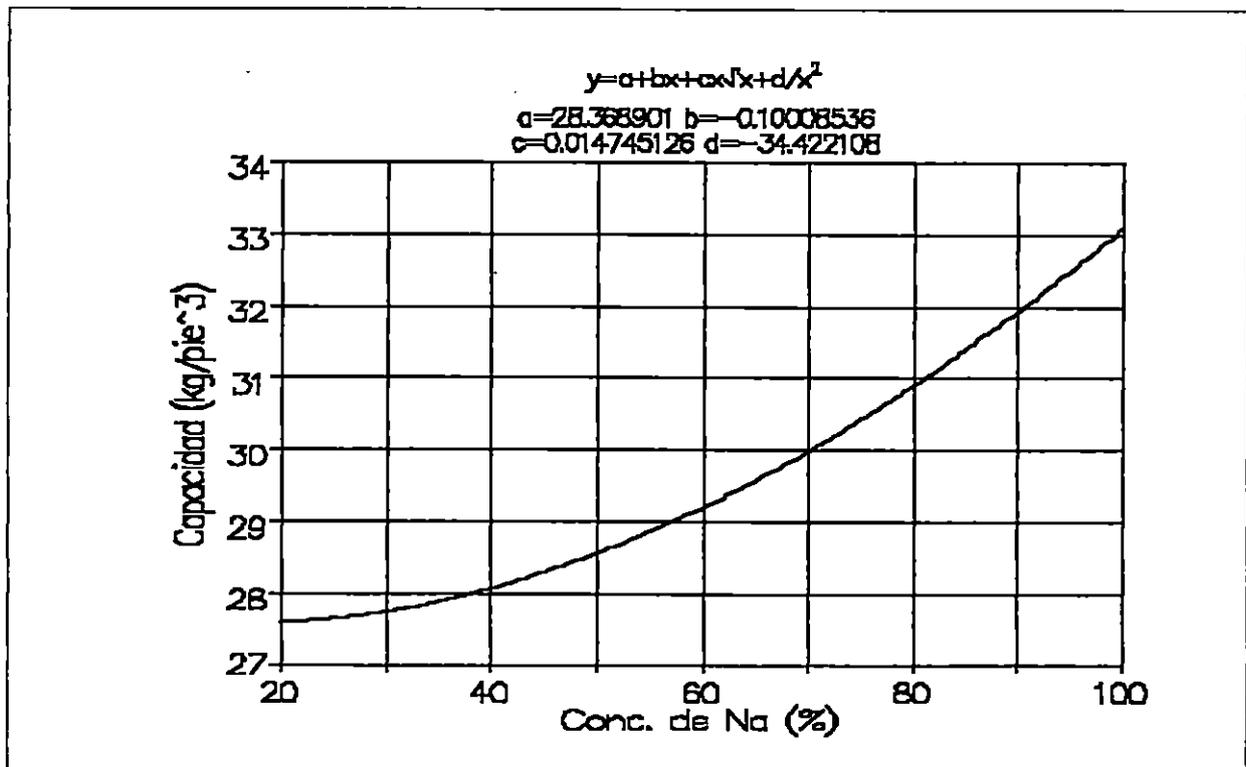


FIGURA 4.10 CAPACIDAD DE LA RESINA CATIONICA A 50% DE ALCALINIDAD PARA UN RANGO DE CONCENTRACION DE SODIO DE 20-100%

75% de Alcalinidad

La gráfica 4.1 para una alcalinidad del 75%, se secciono en dos partes, así:

a. Para un rango de concentración de sodio de 0-20% el valor es constante, donde:

$$Y=28.0 \quad \text{Ec.4.29}$$

b. Para el rango de concentración de sodio de 20-100%

$$Y = a + bX^2\sqrt{X} + cX^3 + \frac{d}{X^2} \quad \text{Ec.4.30}$$

Donde:

$$a = 27.54811495$$

$$b = 0.000144398$$

$$c = -8.3747 \times 10^{-6}$$

$$d = 103.8796246$$

La figura 4.11 representa los datos reales correlacionados por la ecuación 4.30

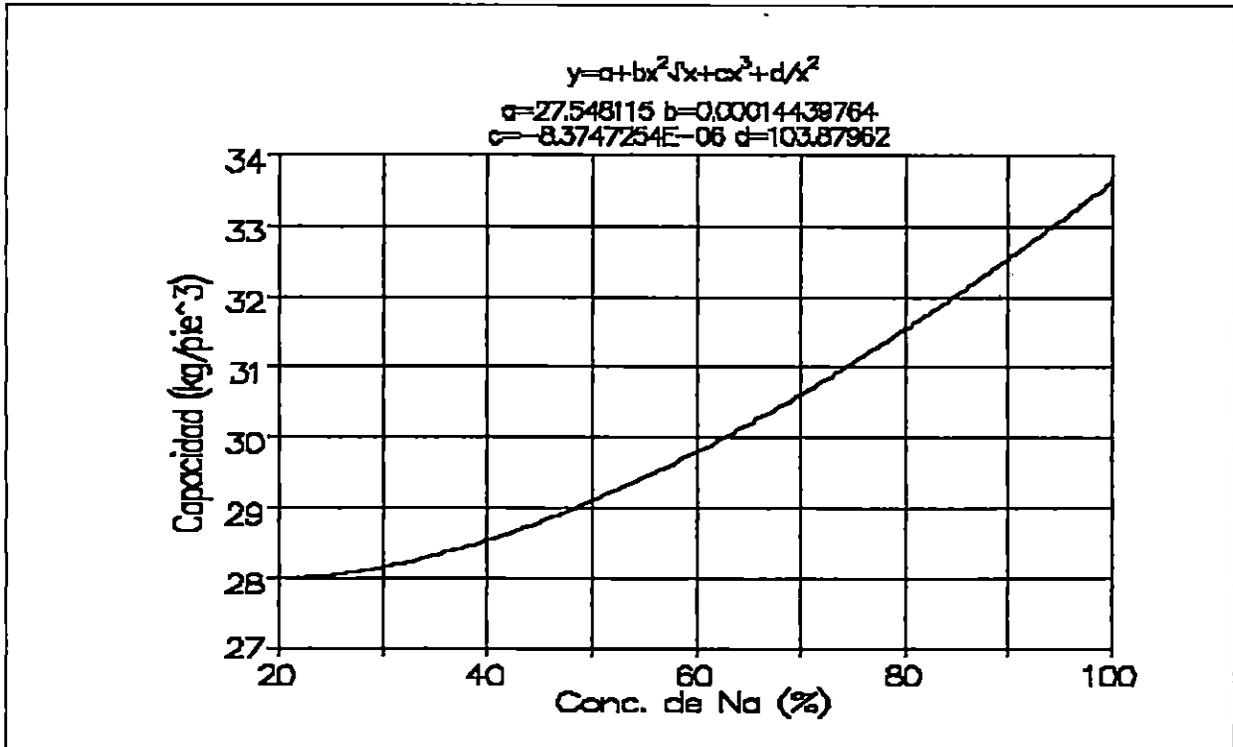


FIGURA 4.11 CAPACIDAD DE LA RESINA CATIONICA PARA 75% DE ALCALINIDAD Y UN RANGO DE CONCENTRACION DE SODIO DE 20-100%.

100% de Alcalinidad

La gráfica 4.1 para una alcalinidad del 100% se secciono en dos partes:

- a. En el rango de 0-20% de concentración de sodio, se presenta un valor constante.

$$Y=28.46 \quad \text{Ec.4.31}$$

Para el rango de concentración de sodio de 20-100%, se obtuvo la expresión:

$$Y = a + bX + cX^2 + dX^3 \quad \text{Ec.4.32}$$

Donde:

$$a = 28.79843563$$

$$b = -0.04218745$$

$$c = 0.001380818$$

$$d = -4.3808 \times 10^{-6}$$

La figura 4.12 muestra los datos reales correlacionados por por la ecuación 4.32

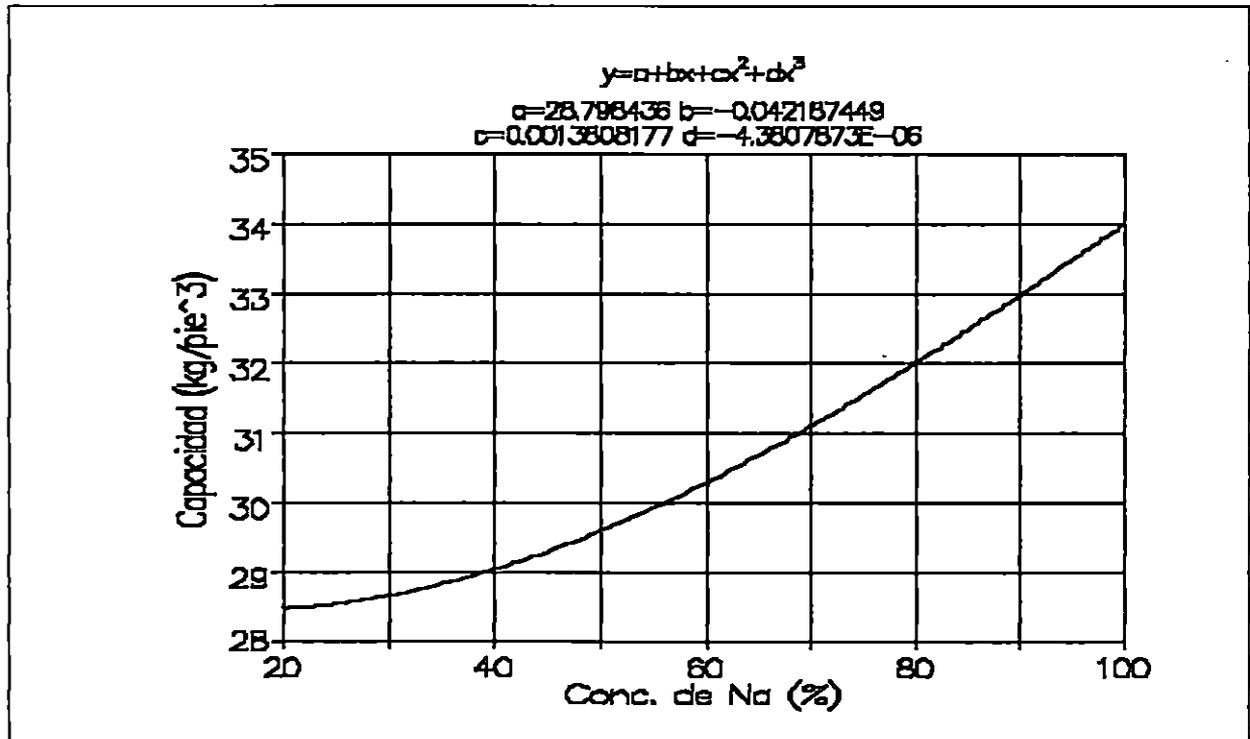


FIGURA 4.12 CAPACIDAD DE LA RESINA CATIONICA PARA 100% DE ALCALINIDAD Y UN RANGO DE CONCENTRACION DE SODIO DE 20-100%.

4.7.3 CALCULO PARA LA UNIDAD ANIONICA

El cálculo de la capacidad para la resina aniónica puede realizarse para 3 niveles de regeneración 4, 6 y 8 libras de NaOH 100% /pie³ de resina

Los parámetros requeridos:

$$\% \text{acido carbonico} = \frac{\text{gpg CO}_3^{2-} + \text{gpg HCO}_3^- + \text{gpg CO}_2}{\text{Factor Z}} \times 100 \quad \text{Ec.4.33}$$

$$\% \text{ de iones monovalentes} = \frac{\text{gpg Cl}^- + \text{gpg NO}_3^-}{\text{Factor Z}} \times 100 \quad \text{Ec.4.34}$$

$$\% \text{ de silice} = \frac{\text{gpg SiO}_2}{\text{Factor Z}} \times 100 \quad \text{Ec.4.35}$$

La capacidad de la resina anionica dependerá del nivel de regeneración , así:

- para 4 lbs de NaOH 100%/pie³ 7,800 gpg como CaCO₃
- para 6 lbs de NaOH 100%/pie³ 15,200 gpg como CaCO₃
- para 8 lbs de NaOH 100%/pie³ 16,100 gpg como CaCO₃

La capacidad de la resina obtenida debe ser corregida en base a los % de ácido carbónico, iones monovalentes y silice. Estos factores de corrección correspondientes a cada porcentaje se muestran en la tabla 4.9, y se representan por F₁, F₂, F₃.

La capacidad corregida se obtiene multiplicando la capacidad inicial por cada uno de los factores de corrección, por tanto, se tiene que:

$$\text{Cap. corregida} = \text{Cap. de la resina aniónica} \times F_1 \times F_2 \times F_3 \quad \text{Ec.4.36}$$

El volumen requerido de resina para tratar un volumen determinado de agua sera:

$$\text{Vol. resina aniónica} = G' \times \text{Factor Z} \times 1/\text{Cap. corregida} \quad \text{Ec.4.37}$$

donde :

G' :es el volumen de agua cruda en galones que se desea tratar

TABLAS DE CAPACIDADES

Los volúmenes de resinas calculados anteriormente son relativamente pequeños por lo que resulta difícil encontrar, en el mercado nacional, tanques prefabricados que tengan las dimensiones adecuadas para manejar dichos volúmenes. Para facilitar la labor del usuario del programa se incluye en el ANEXO 3 un listado de volúmenes de tanques fabricados por la compañía MATT-SON, INC.

Una vez calculado el volumen de resina para la unidad aniónica, debe seleccionarse el tanque inmediatamente superior.

El galonaje de agua decationizada que producirán X pie³ de resina catiónica se determina por la expresión:

$$\text{Galones de agua decationizada} = \frac{(\text{Cap. de resina}) \times (\text{Vol. de resina})}{\text{Factor de carga Y}} \quad \text{Ec.4.38}$$

El volumen calculado por la ecuación anterior es el galonaje total de agua decationizada que teóricamente producirá la resina catiónica por ciclo de regeneración. Sin embargo para la regeneración de la resina aniónica se utiliza agua decationizada en un promedio de 100 galones por pie³ de resina aniónica por lo que el galonaje neto de la unidad catiónica será:

$$\text{Galonaje neto} = \text{Gal. de agua decationizada} - \frac{100 \text{ gal}}{\text{pie}^3} \times \text{vol. de resina} \quad \text{Ec.4.39}$$

Debido a que también se maneja el mismo volumen para la resina aniónica, los galones de agua producidos por esta serán:

$$\text{Gal. unidad anionica} = \frac{\text{cap. corregida} \times \text{vol. de resina}}{\text{Factor Z}} \quad \text{Ec.4.40}$$

Este volumen estará libre de aniones y cationes dado que el agua que pasa por la unidad aniónica es agua decationizada, resultando de esta manera un efluente de agua desmineralizada.

DETERMINACION DE FUGAS

Dos de los parámetros más importantes para evaluar el punto final del ciclo de servicio son la fuga de sodio desde la unidad catiónica y la fuga de sílice desde la unidad aniónica.

a. Fuga de sodio de la unidad catiónica.

Con los % de sodio y alcalinidad determinados con las ecuaciones 4.8 y 4.9 y utilizando los modelos matemáticos resultantes de la correlación de los datos leídos del gráfico 4.2, se tiene que:

$$\text{Fuga de sodio} = \frac{\% \text{ fuga}}{100} \times \text{Factor Y} \times 17.1 \quad \text{Ec.4.41}$$

b. Fuga de sílice desde la unidad aniónica.

Para determinar la fuga de sílice se utiliza el % de sílice y las ppm de fuga de sodio determinadas anteriormente.

los modelos matemáticos obtenidos de la correlación de los datos de la gráfica 4.5 se muestran a continuación:

Para un nivel de regeneración de 4 lbs de NaOH se obtuvo la siguiente expresión,

$$Y = a + bX + cX^2 + dX^3 + eX^4 \quad \text{Ec.4.42}$$

Donde:

$$a = .002678794$$

$$b = 0.010032353$$

$$c = 0.000109585$$

$$d = -5.9017 \times 10^{-7}$$

$$e = 4.94686 \times 10^{-8}$$

Para un nivel de regeneración de 6 lbs de NaOH, se obtuvo la expresión,

$$Y = a + bX^3 + c(e^X) + d\sqrt{X} \quad \text{Ec.4.43}$$

Donde:

$$a = -0.0569158$$

$$b = 1.97013 \times 10^{-6}$$

$$c = 7.03027 \times 10^{-20}$$

$$d = 0.031997364$$

para un nivel de regeneración de 8 lbs de NaOH, se obtuvo la ecuación,

$$Y = a + bX + cX^2 + dX^3 + eX^4 \quad \text{Ec.4.44}$$

Donde:

$$a = 0.000581559$$

$$b = 0.003391996$$

$$c = 5.76217 \times 10^{-5}$$

$$d = -2.2318 \times 10^{-6}$$

$$e = 5.0477 \times 10^{-8}$$

Haciendo uso de las ecuaciones anteriores para un nivel de regeneración (libras de NaOH) seleccionado, y con el valor de % de sílice se calcula fácilmente el valor de la fuga promedio de sílice.

El valor obtenido para la fuga de sodio debe ser corregido, para ello se utiliza el mismo valor de fuga de sodio utilizado en el cálculo anterior, y las expresiones que se obtuvieron de correlacionar los puntos leídos del gráfico 4.6. Las ecuaciones resultantes son las siguientes:

La figura 4.13 muestra la correlación de los datos reales de la gráfica 4.6, para un nivel de regeneración de 4 lbs

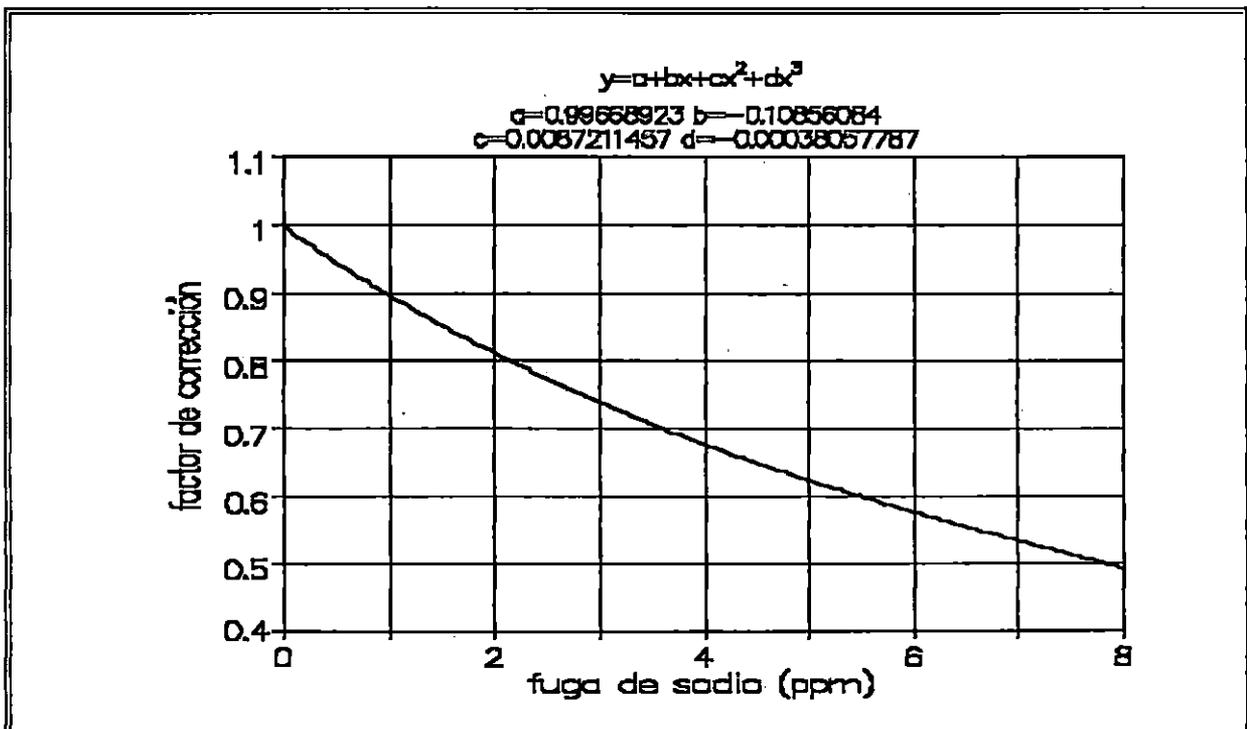


FIGURA 4.13 FACTOR DE CORRECCION PARA LA FUGA DE SILICE DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION DE 4 LBS.

a. Para un nivel de regeneración de 4 lbs de NaOH

Rango de fuga de sodio 0 a 4 ppm como CaCO_3

$$Y = a + bX + cX^2 + dX^3 \quad \text{Ec.4.45}$$

en donde:

$$a = 0.996578845$$

$$b = -0.10827244$$

$$c = 0.008594325$$

$$d = -0.00036668$$

b. para un nivel de regeneración de 6 lbs de NaOH

• rango de fuga de sodio de 0 a 4 ppm como CaCO_3

$$Y = a + bX + cX^2 + dX^3 + eX^4 \quad \text{Ec.4.46}$$

en donde:

$$a = 1.000283537$$

$$b = -0.10753709$$

$$c = -0.00585503$$

$$d = 0.000507665$$

La figura 4.14 muestra la correlación de los datos reales de la gráfica 4.6, para un nivel de regeneración de 6 lbs

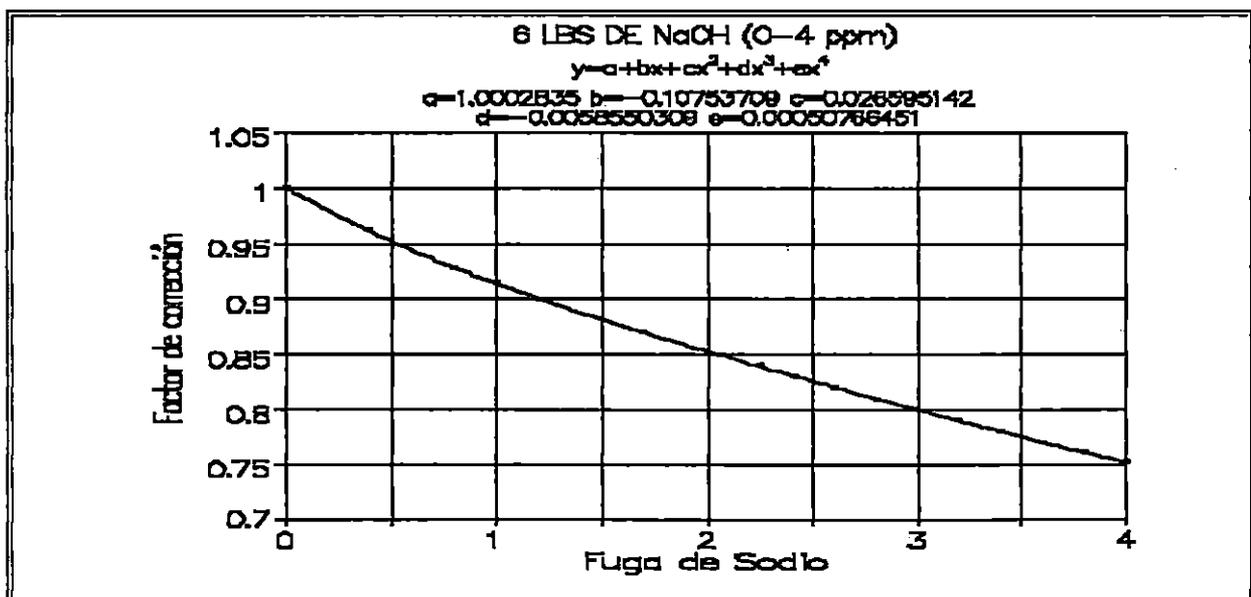


FIGURA 4.14 FACTOR DE CORRECCION DE FUGA DE SILICE DEBIDO AL SODIO PARA UN NIVEL DE REGENERACION DE 6 LBS, Y UN RANGO DE OPERACION DE 0-4 ppm DE FUGA DE SODIO.

Para un rango de fuga de sodio de 4 a 8 ppm como CaCO_3 , se obtuvo la expresión:

$$Y = a + bX + cX^2 + \frac{d}{X} + \frac{e}{X^2} \quad \text{Ec.4.47}$$

Donde:

$$a = 4.15008908$$

$$b = -0.44925802$$

$$c = 0.019416879$$

$$d = -11.5270744$$

$$e = 15.50218589$$

La figura 4.15 muestra la correlación de los datos reales de la gráfica 4.6, para un nivel de regeneración de 6 lbs y un rango de operación de regeneración de 4-8 ppm.

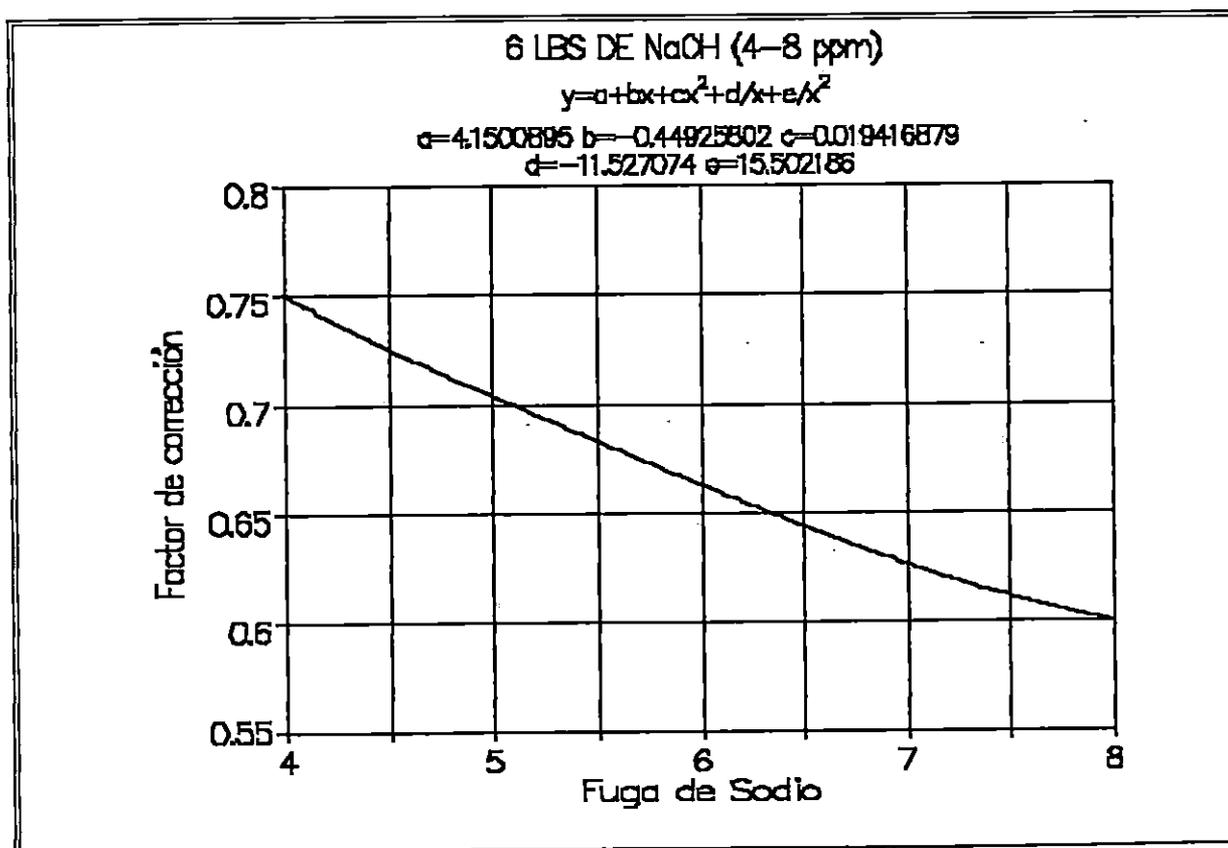


FIGURA 4.15 FACTOR DE CORRECCION DE FUGA DE SILICE DEBIDO AL SODIO, Y UN NIVEL DE REGENERACION DE 6 LBS, (4-8 ppm DE FUGA DE SODIO).

- c. para un nivel de regeneración 8 lbs de NaOH
- para rango de fuga de sodio de 0 a 4 ppm como CaCO_3 se obtuvo la siguiente expresión:

$$Y = a + bX + cX^2 + dX^3 \quad \text{Ec.4.48}$$

Donde:

$$a = 0.999962002$$

$$b = -0.0567203$$

$$c = 0.006175276$$

$$d = -0.00048514$$

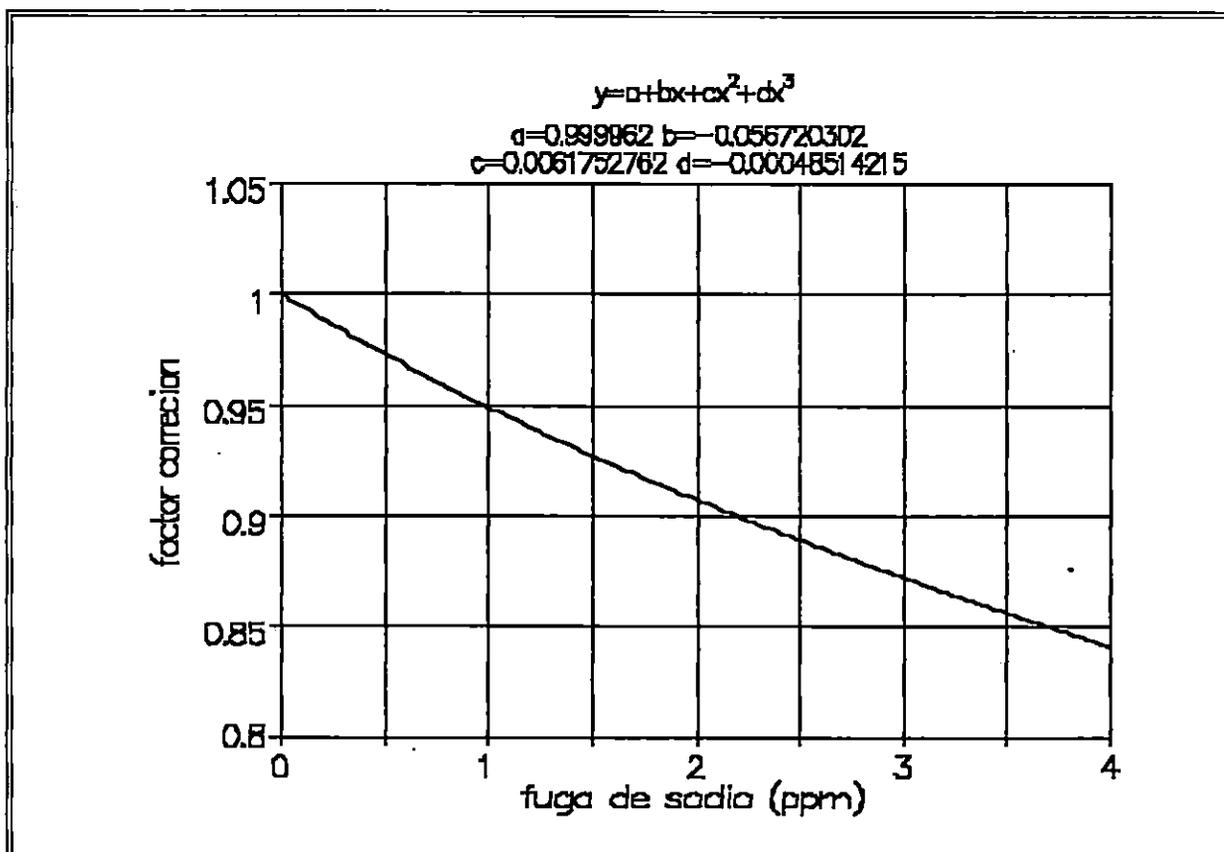


FIGURA 4.16 FACTOR DE CORRECCION DE FUGA DE SILICE DEBIDO AL SODIO, PARA UN NIVEL DE REGENERACION DE 8 LBS Y UN RANGO DE APLICACION DE FUGA DE SODIO DE 0-4 ppm.

- para un rango de fuga de sodio de 4 a 8 ppm como CaCO_3 se obtuvo la expresión:

$$Y = a + bX + cX^2 + dX^3 + eX^4$$

Donde:

$$a = 0.807138711$$

$$b = 0.086232677$$

$$c = -0.03236319$$

$$d = 0.003930056$$

$$e = -0.0001753$$

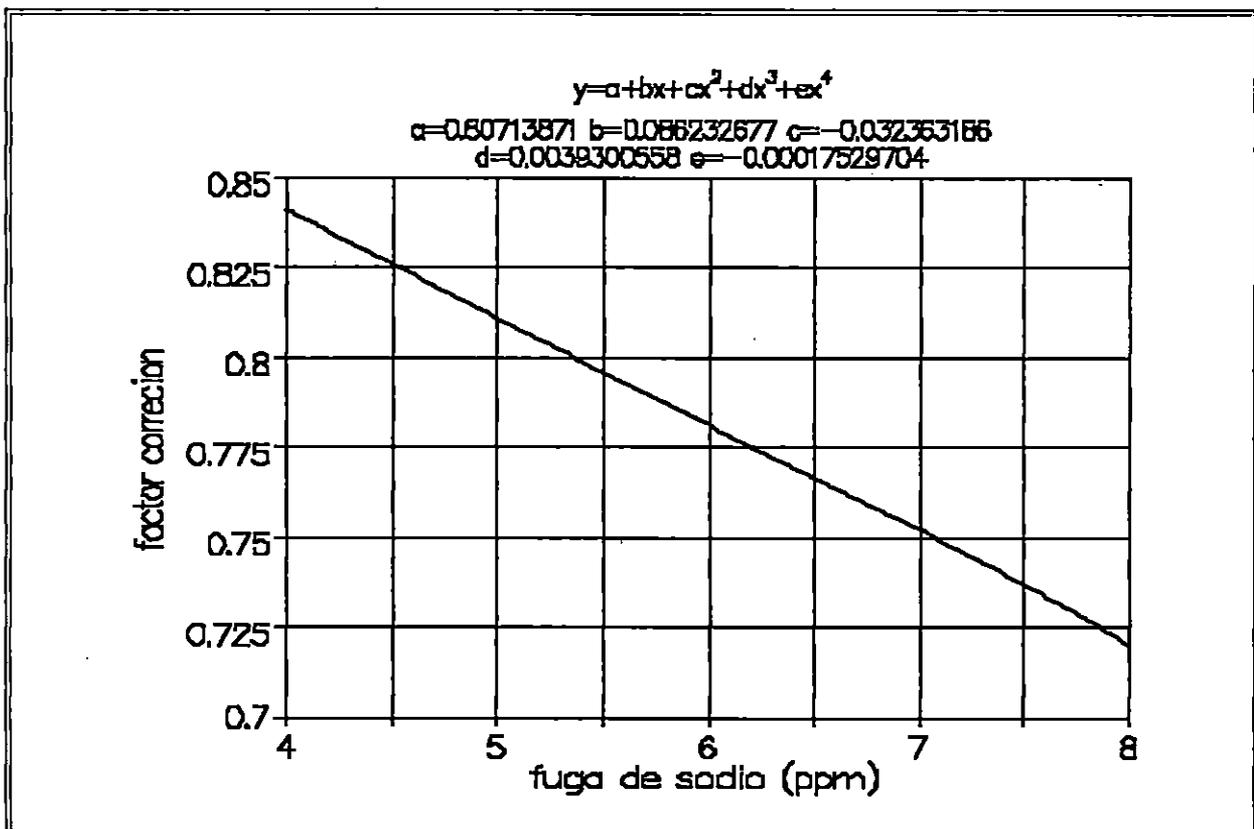


FIGURA 4.17 FACTOR DE CORRECCION DE FUGA DE SILICE DEBIDO AL SODIO, PARA UN NIVEL DE REGENERACION DE 8 LBS, Y UN RANGO DE APLICACION DE 4-8 ppm.

El cálculo del factor de corrección para la fuga de sodio se obtiene al sustituir el valor de la fuga de sodio en las ecuaciones anteriores para un nivel de regeneración especificado

Entonces, al dividir la fuga promedio de sílice entre el factor de corrección se encuentra la fuga corregida:

$$\text{Fuga de sílice corregida} = \frac{\text{Fuga promedio de sílice}}{\text{Factor de corrección}} \times 100 \quad \text{Ec.4.50}$$

Los valores obtenidos a partir de la ecuación anterior se estima son los que teóricamente ocurrirán en el punto final del ciclo de servicio, es decir cuando la resina aniónica se haya agotado.

4.8 PROGRAMA DE INTERCAMBIO IONICO EN QUICK C

```

#include "conio.h"
#include "stdio.h"
#include "stdlib.h"
#include "dos.h"
#include "stdarg.h"
#include "math.h"

void Intercambiolonico();
int PrmtrsAnionCation(void);
int UnidadCationica();
int FugaSodio(void);
int FugaSilice(void);
double FPS_Nivel4(double x);
double FPS_Nivel6(double x);
double FPS_Nivel8(double x);
double FC_Nivel4(double x);
double FC_Nivel6(double x);
double FC_Nivel8(double x);
int UnidadAnionica();
int CapturaNumero(char *fld, int row, int col, int maxlen, int attr);
double CapacidadResinaPct0(double x);
double RetomaDbf(int NumeroEval, int fil, int col, int cifras);
double CapacidadResinaPct25(double x);
double CapacidadResinaPct50(double x);
double CapacidadResinaPct75(double x);
double CapacidadResinaPct100(double x);
double PctCationesPct0(double x);
double PctCationesPct25(double x);
double PctCationesPct50(double x);
double PctCationesPct75(double x);
int InicializaMatrizAnionica(void);
int BuscaFactores(void);
/* ----- FUNCIONES COMUNES ----- */
int CreaVentanaMensaje(int MensajeUnico, int NumeroPalabras,...);
void Cursor(int row, int col);
void BorraBloque(int top, int left, int btm, int right, int attr);
void DespliegaCadena(char *str, int attr, int row, int col);
void pca(char c, int attr, int row, int col);
void Ventana(int trow, int lcol, int brow, int rcol, int attr, int border,
char *title);
int getatr(void);
void cls(int attr);
int getvmode(int *iscolor);

#define IZQUIERDA 75
#define DERECHA 77
#define ARRIBA 72
#define ABAJO 80
#define F3 61
#define F1 59
#define ENTER 13
#define ESCAPE 27
#define WHITE 15
#define BLACK 0
#define BLUE 1
/* ----- Define's propios ----- */
#define OPCION_121 "C lculo para la Unidad Catiónica"
#define OPCION_122 "Fuga de Sodio"
#define OPCION_123 "C lculo para la Unidad Aniónica"
#define OPCION_124 "Fuga de Silice"
#define MENSAJE_121 "C lculo Para la Unidad Catiónica."
#define MENSAJE_122 "C lculo de la Fuga de Sodio."
#define MENSAJE_123 "C lculo Para la Unidad Aniónica."
#define MENSAJE_124 "C lculo de la Fuga de Silice."
#define MENSAJE_ERR01 "ERR01: El Dato debe ser >= 0 y <= 100. Presione Cualquier Tecla."
#define MENSAJE_ERR02 "ERR02: El Dato debe ser > 0. Presione Cualquier Tecla."

```

```

#define MENSAJE_ERR03 "ERR03!: Error en el Valor Ingresado. Presione Cualquier Tecla."
#define MENSAJE_ERR04 "ERR04!: Memoria Insuficiente para esta Opci3n. Presione ESCAPE."
#define MAGENTA 5
/*-----*/

int i,PosicionMenu;
int attr0,attr1,attr2,attr3,iscolor;
char temp[80],Imprimir[1];
FILE *pfile5;
/*----- Variables propias -----*/
int SalirOpcion_12,NumeroCorrecto,Reconstruye,attr4,NivelRegeneracion;
int VarRetorno,HacerFugaSilice,HacerFugaSodio,HacerUniAnionica;
char NumeroEnCadena[80];
double PPMsodio,PPMPotasio,FactorY,FactorZ;
double PPMCalcio,PPMMagnesio,PPMHidroxido,PPMSulfato,PPMManganeso,PPMHierro;
double PPMCO3,PPMHCO3,PPMCO2,PPMCloruro,PPMNitrato,PPMSilice;
double PctAcidoCarbonico,PctIonesMonovalentes,PctSilice;
double PctSodio,PctAlcalinidad,CapacidadDeResina,DiferenciaDeCapacidad;
double CapacidadPorUnidad,VolumenResina,VolumenTanque,FugaPromedioSilice;
double PctCationes,FugaDeSodio,FugaDeSodioEntera,GalonesH2O,GalH2ODecat;
double GalNeto,CapacidadDelIntercambio,CapacidadCorregida,FugaSiliceCorregida;
double GalUAnionico,TabCRAnionica[20][7],Factor1,Factor2,Factor3,FactorCorreccion;

/*-----*/
void Cursor(int row, int col)
/*-----*/
{
    /* envia el Cursor a la Posici3n (row=0,col=0 Origen) */
    union REGS rin,rout;
    rin.h.ah=2;
    rin.h.bh=0;
    rin.h.dh=row;
    rin.h.dl=col;
    int86(0x10,&rin,&rout);
}

/*-----*/
void DespliegaCadena(char *str, int attr, int row, int col)
/*-----*/
{
    /* Escribe una Cadena con Atributo y coordenadas */
    while (*str)
        pca(*str++,attr,row,col++);
}

/*-----*/
void BorraBloque(int top, int left, int btm, int right, int attr)
/*-----*/
{
    /* Clears Screen Block w/Attribute, Cursor Inside */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.al=0;
    rin.h.bh=attr;
    rin.h.ch=top; /* top row */
    rin.h.cl=left; /* top column */
    rin.h.dh=btm; /* bottom row */
    rin.h.dl=right; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(top+1,left+1);
}

/*-----*/
int getattr(void)
/*-----*/
{
    union REGS rin,rout;
    rin.h.ah=8; /* get attribute at Cursor */
    rin.h.bh=0;
    int86(0x10,&rin,&rout);
    return((int)rout.h.ah);
}

```

```

/*-----*/
void pca(char c, int attr, int row, int col)
/*-----*/
{
    union REGS rin,rout;
    Cursor(row,col);      /* set cursor */
    rin.h.ah=9;          /* print char with attribute */
    rin.h.al=c;
    rin.h.bh=0;
    rin.h.bl=attr;
    rin.x.cx=1;
    int86(0x10,&rin,&rout);
}

/*-----*/
void Ventana(int trow, int lcol, int brow, int rcol, int attr,
int border, char *title)
/*-----*/
{
    /* Set Simple Window and Title */
    int tl,tr,bl,br,lr,tb,i,length;
    BorraBloque(trow,lcol,brow,rcol,attr);
    if(border==1){      /* Single Border */
        tl=218; tr=191; bl=192; br=217; lr=196; tb=179;
    }
    else{                /* Double Border */
        tl=201; tr=187; bl=200; br=188; lr=205; tb=186;
    }
    pca(tl,attr,trow,lcol);
    pca(bl,attr,brow,lcol);
    pca(tr,attr,trow,rcol);
    pca(br,attr,brow,rcol);
    for(i=lcol+1; i<rcol; i++){
        pca(lr,attr,trow,i);
        pca(lr,attr,brow,i);
    }
    for(i=trow+1; i<brow; i++){
        pca(tb,attr,i,lcol);
        pca(tb,attr,i,rcol);
    }
    i=0;
    while(title[i++]!='\0');
    length=i-1;
    DespliegaCadena(title,attr,trow,(rcol+lcol)/2-length/2);
}

/*-----*/
void cls(int attr)
/*-----*/
{
    /* Clears Screen w/Attribute, Cursor to Origin */
    union REGS rin,rout;
    rin.h.ah=6;
    rin.h.al=0;
    rin.h.bh=attr;
    rin.h.ch=0; /* top row */
    rin.h.cl=0; /* top column */
    rin.h.dh=24; /* bottom row */
    rin.h.dl=79; /* bottom column */
    int86(0x10,&rin,&rout);
    Cursor(0,0);
}

/*-----*/
int getvmode(int *iscolor)
/*-----*/
{
    union REGS rin,rout;
    int mode;
    rin.h.ah=15;          /* get display mode */
    int86(0x10,&rin,&rout);
    mode=(int)rout.h.al;
}

```

```

*iscolor=1;
if(mode==0 || mode==2 || mode==7)
  *iscolor=0;
return(mode);    /* sets color flag, returns video mode */
}

/* ----- */
int CreaVentanaMensaje(int Unico, int NumeroPalabras, ...)
/* ----- */
{
  int Bandera = 1;
  int Longitud;
  char *Rotulo;

  va_list parameters;
  va_start(parameters, NumeroPalabras);

  strcpy(temp, "");
  BorraBloque(22,1,22,78,attr2);
  if (Unico)
  {
    Rotulo = va_arg(parameters, char *);
    DespliegaCadena(Rotulo,attr1,22,3);
    return(0);
  }
  Longitud = 3;
  for (i=0; i<NumeroPalabras; i++)
  {
    Rotulo = va_arg(parameters, char *);
    if (Bandera)
    {
      DespliegaCadena(Rotulo,attr3,22,3+Longitud);
      Bandera = 0;
    }
    else
    {
      DespliegaCadena(Rotulo,attr1,22,3+Longitud);
      Bandera = 1;
    }
    Longitud = Longitud + strlen(Rotulo) + 2;
  }
  return(0);
}

/* ----- Funciones Propias ----- */

/* ----- */
int InicializaMatrizAnionica(void)
/* ----- */
{
  int i,j;

  /* Reservaci3n en Memoria */
  if (malloc(160 * sizeof(double)) == NULL) return (-1);

  /* Acido Carb3nico */
  TabCRAnionica[0][0] = 0;
  TabCRAnionica[1][0] = 2;
  TabCRAnionica[2][0] = 4;
  TabCRAnionica[3][0] = 6;
  TabCRAnionica[4][0] = 8;
  TabCRAnionica[5][0] = 10;
  TabCRAnionica[6][0] = 13;
  TabCRAnionica[7][0] = 17;
  TabCRAnionica[8][0] = 20;
  TabCRAnionica[9][0] = 25;
  TabCRAnionica[10][0] = 30;
  TabCRAnionica[11][0] = 36;
  TabCRAnionica[12][0] = 44;
  TabCRAnionica[13][0] = 54;
}

```

```

TabCRAnionica[14][0] = 65;
TabCRAnionica[15][0] = 78;
TabCRAnionica[16][0] = 100;
TabCRAnionica[0][1] = 1;
for (j = 1; j < 16; j++)
    TabCRAnionica[j][1] = TabCRAnionica[j-1][1] + 0.01;
TabCRAnionica[16][1] = 1.15;

```

/ Iones Monovalentes */*

```

TabCRAnionica[0][2] = 0;
TabCRAnionica[1][2] = 6;
TabCRAnionica[2][2] = 12;
TabCRAnionica[3][2] = 20;
TabCRAnionica[4][2] = 30;
TabCRAnionica[5][2] = 40;
TabCRAnionica[6][2] = 53;
TabCRAnionica[7][2] = 65;
TabCRAnionica[8][2] = 80;
TabCRAnionica[9][2] = 100;
TabCRAnionica[0][3] = 0.99;
for (j = 1; j < 9; j++)
    TabCRAnionica[j][3] = TabCRAnionica[j-1][3] - 0.01;
TabCRAnionica[9][3] = 0.91;

```

/ Silice */*

/ 4lbs NaOH */*

```

TabCRAnionica[0][4] = 0;
TabCRAnionica[1][4] = 1.3;
TabCRAnionica[2][4] = 2.5;
TabCRAnionica[3][4] = 4;
TabCRAnionica[4][4] = 5;
TabCRAnionica[5][4] = 7;
TabCRAnionica[6][4] = 9;
TabCRAnionica[7][4] = 10.5;
TabCRAnionica[8][4] = 12;
TabCRAnionica[9][4] = 14;
TabCRAnionica[10][4] = 17;
TabCRAnionica[11][4] = 19;
TabCRAnionica[12][4] = 21;
TabCRAnionica[13][4] = 24;
TabCRAnionica[14][4] = 26;
TabCRAnionica[15][4] = 29;
TabCRAnionica[16][4] = 32;
TabCRAnionica[17][4] = 35;
TabCRAnionica[18][4] = 38;
TabCRAnionica[19][4] = 42;
TabCRAnionica[20][4] = 44;

```

/ 6lbs de NaOH */*

```

TabCRAnionica[0][5] = 0;
TabCRAnionica[1][5] = 4;
TabCRAnionica[2][5] = 8;
TabCRAnionica[3][5] = 11;
TabCRAnionica[4][5] = 15;
TabCRAnionica[5][5] = 19;
TabCRAnionica[6][5] = 22;
TabCRAnionica[7][5] = 25;
TabCRAnionica[8][5] = 29;
TabCRAnionica[9][5] = 32;
TabCRAnionica[10][5] = 35;
TabCRAnionica[11][5] = 39;
TabCRAnionica[12][5] = 42;
TabCRAnionica[13][5] = 45;
TabCRAnionica[14][5] = 50;

```

/ 8lbs. de NaOH */*

```

TabCRAnionica[0][6] = 0;
TabCRAnionica[1][6] = 8;
TabCRAnionica[2][6] = 14;
TabCRAnionica[3][6] = 20;
TabCRAnionica[4][6] = 25;

```

```

TabCRAnionica[5][6] = 31;
TabCRAnionica[6][6] = 36;
TabCRAnionica[7][6] = 41;
TabCRAnionica[8][6] = 45;
TabCRAnionica[9][6] = 50;
TabCRAnionica[0][7] = 0.99;
for (i = 1; i < 20; i++)
    TabCRAnionica[i][7] = TabCRAnionica[i-1][7] - 0.01;
TabCRAnionica[20][7] = 0.80;

return(0);
}

/*-----*/
int BuscaFactores(void)
/*-----*/
{
    int i=0;

    Factor1 = Factor2 = Factor3 = 0;

    for (i = 1; i <= 16; i++)
        if (i == 16)
        {
            Factor1 = TabCRAnionica[16][1];
            break;
        }
        else
            if (Factor1 == 0 && PctAcidoCarbonico > TabCRAnionica[i][0] &&
                PctAcidoCarbonico <= TabCRAnionica[i+1][0])
            {
                Factor1 = TabCRAnionica[i][1];
                break;
            }

    for (i = 1; i <= 9; i++)
        if (i == 9)
        {
            Factor2 = TabCRAnionica[9][3];
            break;
        }
        else
            if (Factor2 == 0 && PctIonesMonovalentes > TabCRAnionica[i][2] &&
                PctIonesMonovalentes <= TabCRAnionica[i+1][2])
            {
                Factor2 = TabCRAnionica[i][3];
                break;
            }

    if (NivelRegeneracion == 4)
        for (i = 1; i <= 20; i++)
            if (i == 20)
            {
                Factor3 = TabCRAnionica[20][7];
                break;
            }
            else
                if (Factor3 == 0 && PctSilice > TabCRAnionica[i][4] &&
                    PctSilice <= TabCRAnionica[i+1][4])
                {
                    Factor3 = TabCRAnionica[i][7];
                    break;
                }

    if (NivelRegeneracion == 6)
        for (i = 1; i <= 14; i++)
            if (i == 14)
            {

```

```

        Factor3 = TabCRAnionica[14][7];
        break;
    }
    else
        if (Factor3 == 0 && PctSilice > TabCRAnionica[i][5] &&
            PctSilice <= TabCRAnionica[i+1][5])
        {
            Factor3 = TabCRAnionica[i][7];
            break;
        }
}

if (NivelRegeneracion == 8)
    for (i = 1; i <= 8; i++)
        if (i == 9)
        {
            Factor3 = TabCRAnionica[9][7];
            break;
        }
    else
        if (Factor3 == 0 && PctSilice > TabCRAnionica[i][6] &&
            PctSilice <= TabCRAnionica[i+1][6])
        {
            Factor3 = TabCRAnionica[i][7];
            break;
        }
}

return(0);
}

/*-----*/
int CapturaNumero(char *fld, int row, int col, int maxlen, int attr)
/*-----*/
/* Simple Numeric Input Routine          *
 * fld  storage for number as char string *
 * row,col starting cursor position for field *
 * maxlen field length                    *
 * attr  entry field attribute            *
 * returns entry length(x), -entry length(y), 0(ESC hit) */
{
    int i,j,yflag,expflag,pass;
    char c;

    for(j=0;j<maxlen;j++) pca(32,attr,row,col+j);
    Cursor(row,col);
    i=j=0;
    while(i<maxlen)
    {
        c=getch();
        pass=0;
        if((c>='0' && c<='9') || c=='.' || pass)
        {
            pca(c,attr,row,col+i); /* print character */
            Cursor(row,col+i+1); /* advance cursor */
            i++;
            if(!pass) fld[j++]=c;
        }
        else
            if((c==10 || c==13 || i==maxlen) && i!=0) break;
            else
                if(c=='\b' && i>0)
                {
                    i--;
                    pca(32,attr,row,col+i);
                    if(j!=0)
                    {
                        j--;
                        fld[j]=0;
                    }
                }
            }
        }
    }
    else if(c==27) return(0);
}

```

```

}
fid[j]='\0';
return(j);
}
/*-----*/
double RetomaDbl(int NumeroEval,int fil,int col,int cifras)
/*-----*/
{
double NumeroDbl;
while (NumeroCorrecto < 0)
{
NumeroCorrecto = CapturaNumero(NumeroEnCadena,fil,col,cifras,attr0);
if (NumeroCorrecto >= 0) NumeroDbl = atof(NumeroEnCadena);
switch (NumeroEval)
{
case 1:
{
if (NumeroDbl < 0)
{
CreaVentanaMensaje(1,1,MENSAJE_ERR02);
while (lkbhit());
CreaVentanaMensaje(0,2,"ESC","Salir");
NumeroCorrecto = -1;
}
break;
}
case 2:
{
if ((NumeroDbl < 0) || (NumeroDbl > 100))
{
CreaVentanaMensaje(1,1,MENSAJE_ERR01);
while (lkbhit());
CreaVentanaMensaje(0,2,"ESC","Salir");
NumeroCorrecto = -1;
}
break;
}
}
}
return(NumeroDbl);
}
/*-----*/
double CapacidadResinaPct0(double x)
/*-----*/
Funci3n Utilizada si 24 <= %Sodio < 60
X= Conc. de Na
Y= Cap. en kg/pie^3
Eqn# 2098 y=a+bx+cx3+d/xy
r2=0.9999836187255227
r2adj=0.9999789383613863
StdErr=0.002257628619663405
Fstat=305221.5565123782
a= 25.22378598210664
b= 0.0281369306561957
c= 4.564448110167486E-06
d= 251.9147073500656
-----
Funci3n utilizada si 60 <= %Sodio <= 100
X= Conc. de Na
Y= Cap. en kg/pie^3
Eqn# 4453 y=a+bx+cx3+d/x+e/xy
r2=0.9999856635847796
r2adj=0.9999805434364866
StdErr=0.005018150742428631
Fstat=261567.9150468504
a= -71.82229503033703
b= 0.8868352123260799
c= -0.002414113510489229
d= 4752.650715931214

```

```

e= -86180.76143131175
*/
{
double y;
double x1,x2,x3,x4;
if (x < 24)
y = 26.4;
else
{
if ((x >= 24) && (x < 60))
{
x1 = x;
x2 = x * x * x;
x3 = 1.0 / (x * x);
y = 25.22378598210664 + 0.02813693065619570 * x1
+ 4.564448110167486E-06 * x2 + 251.9147073500656 * x3;
}
else
if ((x >= 60) && (x <= 100))
{
x1 = x;
x2 = x * x;
x3 = 1.0 / x;
x4 = 1.0 / (x * x);
y = - 71.82229503033703 + 0.8868352123260799 * x1
- 0.002414113510489229 * x2 + 4752.650715931214 * x3
- 86180.76143131175*x4;
}
}
return(y);
}

/*-----*/
double CapacidadResinaPct25(double x)
/*-----*/
{
/*-----*/
Función utilizada cuando el 24 <= %Sodio <= 100
X= Conc. de Na
Y= Cap. en kg/pie^3
Eqn# 2050 y=a+bx+cx^2+d/x
r2=0.9999956185685658
r2adj=0.9999942704358168
StdErr=0.003697899830189761
Fstat=1065096.256797876
a= 24.49623135295213
b= 0.03437891782871094
c= 0.0003918916709476862
d= 40.17298637222633
/*-----*/

double y;
double x1,x2,x3;
if (x < 24)
y = 27.2;
else
{
x1 = x;
x2 = x * x;
x3 = 1.0 / x;
y = 24.49623135295213 + 0.03437891782871094 * x1
+ 0.0003918916709476862 * x2 + 40.17298637222633 * x3;
}
return(y);
}

/*-----*/
double CapacidadResinaPct50(double x)
/*-----*/
{
/*-----*/

```

```

Funci3n utilizada para 20 <= %Sodio <= 100
X= Conc. de Na
Y= Cap. en kg/pie^3
Eqn# 2036 y=a+bx+cx0x+d/x3
r2=0.9999912949369951
r2adj=0.9999878129117932
StdErr=0.005846191541548298
Fstat=421207.1464668689
a= 28.36890087982335
b= -0.1000853616423475
c= 0.01474512647777663
d= -34.42210777276447
*/
double y;
double x1,x2,x3;
if (x < 20)
{
y = 27.6;
}
else
{
x1 = x;
x2 = x * sqrt(x);
x3 = 1.0 / (x * x);
y = 28.36890087982335 - 0.1000853616423475 * x1
+ 0.01474512647777663 * x2 - 34.42210777276447 * x3;
}
return(y);
}
/*-----*/
double CapacidadResinaPct75(double x)
/*-----*/
{
/*-----*/
Funci3n Utilizada si 20 <= %Sodio <= 100
X= Conc. de Na
Y= Cap. en kg/pie^3
Eqn# 2783 y=a+bx0x+cx3+d/x3
r2=0.9999936012337406
r2adj=0.9999922541250543
StdErr=0.004879650775426145
Fstat=1041860.842841998
a= 27.54811494901291
b= 0.0001443976412620096
c= -8.374725359554259E-06
d= 103.8796246159244
*/
double y;
double x1,x2,x3;
if (x < 20)
y = 28;
else
{
x1 = x * x * sqrt(x);
x2 = x * x * x;
x3 = 1.0 / (x * x);
y = 27.54811494901291 + 0.0001443976412620096 * x1
- 8.374725359554259E-06 * x2 + 103.8796246159244 * x3;
}
return(y);
}
/*-----*/
double CapacidadResinaPct100(double x)
/*-----*/
{
/*-----*/
Funci3n Utilizada si 20 <= %Sodio <= 100
X= Conc. de Na

```

```

Y= Cap. en kg/pie^3
Eqn# 2040 y=a+bx+cx^2+dx^3
r2=0.9999960469417735
r2adj=0.999994992792913
StdErr=0.00417154065571589
Fstat=1349161.065876155
a= 28.79843562806118
b= -0.04218744901429886
c= 0.001380817728405144
d= -4.380787253748669E-06
*/
double y;
if (x < 20)
  y = 28.46;
else
  y = 28.79843562806118 + x * (-0.04218744901429886 +
    x * (0.001380817728405144 + x * -4.380787253748669E-06));
return(y);
}

/*-----*/
double PctCationesPct0(double x)
/*-----*/
{
  double y;
  double x1,x2,x3,x4;
/*-----*/
  Funci3n utilizada para 0 <= %Sodio < 60
  X= %Concentraci3n de Na
  Y= Fuga(%cationes Tot.)
  Eqn# 2040 y=a+bx+cx^2+dx^3
  r2=0.9999788449604494
  r2adj=0.9999682674406741
  StdErr=0.003787574489359296
  Fstat=141807.181580239
  a= 0.3012379900978633
  b= 0.02542108106893767
  c= 0.0001162022763532784
  d= 1.179896001401903E-06
/*-----*/
  Funci3n utilizada para 60 <= %Sodio <= 100
  X= % C. de sodio
  Y= Fuga (%de cat. tot)
  Eqn# 4453 y=a+bx+cx^2+dx+e/x^2
  r2=0.9999795156289969
  r2adj=0.9999702045512683
  StdErr=0.004350895316635101
  Fstat=146450.1178208135
  a= 52.73752275603943
  b= -0.488526795178844
  c= 0.002105711293174955
  d= -2394.588305031971
  e= 41049.51925513228
/*-----*/
  if ((x >= 0) && (x < 60))
    y=0.3012379900978633+x*(0.02542108106893767+
      x*(0.0001162022763532784+x*1.179896001401903E-06));
  else
  {
    x1=x;
    x2=x*x;
    x3=1.0/x;
    x4=1.0/(x*x);
    y=52.73752275603943-0.488526795178844*x1
      +0.002105711293174955*x2-2394.588305031971*x3
      +41049.51925513228*x4;
  }
  return(y);
}

```

```

/*-----*/
double PctCationesPct25(double x)
/*-----*/
{
    double y;
    double x1,x2,x3;
/*-----*/
    Funci3n utilizada para 0 <= %Sodio <= 100
    X= % C. de sodio
    Y= Fuga (%de cat. tot)
    Eqn# 2090 y=a+bx+cx3+d0x
    r2=0.9999618361416295
    r2adj=0.9999533552842138
    StdErr=0.003799294234439329
    Fstat=165944.7419442169
    a= 0.2012136621713123
    b= 0.01532298251866035
    c= 5.958078240105029E-07
    d= -0.02373395065774799
/*-----*/

    x1=x;
    x2=x*x*x;
    x3=sqrt(x);
    y=0.2012136621713123+0.01532298251866035*x1
      +5.958078240105029E-07*x2-0.02373395065774799*x3;
    return(y);
}

/*-----*/
double PctCationesPct50(double x)
/*-----*/
{
    double y;
    double x1,x2,x3,x4;
/*-----*/
    Funci3n utilizada para 10 <= %Sodio <= 100
    X= % C de sodio
    Y= Fuga(% Cat. tot.)
    Eqn# 4453 y=a+bx+cx3+d/x+e/x3
    r2=0.9999799162351978
    r2adj=0.999973640058697
    StdErr=0.001142302067397471
    Fstat=211609.4609675242
    a= 0.3389744665548092
    b= -0.001210663006307065
    c= 7.146703087430309E-05
    d= -7.075826736305031
    e= 71.96988531709509
/*-----*/

    if ((x >= 0) && (x < 10))
        y = 0.15;
    else
    {
        x1=x;
        x2=x*x;
        x3=1.0/x;
        x4=1.0/(x*x);
        y=0.3389744665548092-0.001210663006307065*x1
          +7.146703087430309E-05*x2-7.075826736305031*x3
          +71.96988531709509*x4;
    }
    return(y);
}

/*-----*/
double PctCationesPct75(double x)
/*-----*/
{
    double y;

```

```

double x1,x2,x3;
/*
Función utilizada 40 <= %Sodio <= 100
X= % C. de sodio
Y= fuga (% de cat.tot.)
Eqn# 2650 y=a+bxy+c/xy+dexp-x
r2=0.9999862093098513
r2adj=0.9999806930337918
StdErr=0.000243884531102648
Fstat=265876.186127015
a= 0.1301923026403671
b= 1.686336365935503E-05
c= -85.90872318986446
d= 372548226092323.8
*/
if ((x >= 0) && (x < 40))
  y = 0.11;
else
  {
  x1=x*x;
  x2=1.0/(x*x);
  x3=exp(-x);
  y=0.1301923026403671+1.686336365935503E-05*x1
    -85.90872318986446*x2+372548226092323.8*x3;
  }
return(y);
}
/*
double FPS_Nivel4(double x)
/*
{
double y;
/*
Función Utilizada para 4.5 <= %Silice <= 40
X= % de silice
Y= Fuga de silice
Eqn# 6001 y=a+bx+cxy+dx3+ex4
r2=0.9999950603505923
r2adj=0.9999931604854355
StdErr=0.0004050115489482236
Fstat=708548.8103223906
a= 0.002678793949497979
b= 0.01003235298470319
c= 0.0001095853383755426
d= -5.901723893238849E-07
e= 4.946860271686856E-08
*/
if ((x >= 0) && (x < 4.5))
  y = 0;
else
  y=0.002678793949497979+x*(0.01003235298470319+
    x*(0.0001095853383755426+x*(-5.901723893238849E-07+
    x*4.946860271686856E-08)));
return(y);
}
/*
double FPS_Nivel6(double x)
/*
{
double y;
double x1,x2,x3;
/*
Función Utilizada para 10.5 <= %Silice <= 40
X= % de silice
Y= Fuga de silice
Eqn# 2879 y=a+bx3+cexpx+d0x
r2=0.999987573632049
r2adj=0.9999844670400613

```

```

StdErr=0.0002906812901526522
Fstat=456013.8789496029
a= -0.05691580224353769
b= 1.970126445645327E-06
c= 7.030267472610914E-20
d= 0.03199736359495247
*/
if ((x >= 0) && (x < 10.5))
  y = 0;
else
  {
    x1=x*x*x;
    x2=exp(x);
    x3=sqrt(x);
    y=-0.05691580224353769+1.970126445645327E-06*x1
      +7.030267472610914E-20*x2+0.03199736359495247*x3;
  }
return(y);
}

/*-----*/
double FPS_Nivel8(double x)
/*-----*/
{
  double y;
/*-----*/
  Funci3n Utilizada cuando 0 <= %Silice <= 13
  X= % de silice
  Y= Fuga de silice
  Eqn# 6001 y=a+bx+cx2+dx3+ex4
  r2=0.9999815658867146
  r2adj=0.9999744758431433
  StdErr=0.0002628035242583279
  Fstat=189861.8841285138
  a= 0.0005815587204846901
  b= 0.003391995711406611
  c= 5.762171388405068E-05
  d= -2.231784251646343E-06
  e= 5.047735641866082E-08
/*-----*/
  if ((x >= 0) && (x <= 13))
    y = 0;
  else
    y=0.0005815587204846901+x*(0.003391995711406611+
      x*(5.762171388405068E-05+x*(-2.231784251646343E-06+
        x*5.047735641866082E-08)));
  return(y);
}

/*-----*/
double FC_Nivel4(double x)
/*-----*/
{
  double y;
/*-----*/
  Funci3n utilizada en 0 <= Fuga de Sodio <= 8
  X= Fuga de Sodio
  Y= Factor de correcci3n
  Eqn# 2040 y=a+bx+cx2+dx3
  r2=0.9999317216619673
  r2adj=0.99992121730227
  StdErr=0.001290033660096261
  Fstat=131804.4017218286
  a= 0.9965788451060245
  b= -0.1082724377002421
  c= 0.008594324728682474
  d= -0.0003666819288225302
/*-----*/
  y=0.9965788451060245+x*(-0.1082724377002421+
    x*(0.008594324728682474+x*-0.0003666819288225302));
}

```

```

return(y);
}

/*-----*/
double FC_Nivel6(double x)
/*-----*/
{
double y;
double x1,x2,x3,x4;
/*-----*/
Funci3n utilizada 0 <= Fuga de Sodio <= 4
X= Fuga de Sodio
Y= Factor de correcci3n
Eqn# 6001 y=a+bx+cx2+dx3+ex4
r2=0.9999746306175387
r2adj=0.9999630990800562
StdErr=0.0004267162439621338
Fstat=118249.7798840063
a= 1.000283537050325
b= -0.1075370910973749
c= 0.02659514232448773
d= -0.0058550309152226
e= 0.0005076645084344239
/*-----*/
Funci3n utilizada 4 <= Fuga de Sodio <= 8
X= Fuga de Sodio
Y= Factor de correcci3n
Eqn# 4453 y=a+bx+cx2+dx+e/x2
r2=0.9999788402576962
r2adj=0.9999707018952718
StdErr=0.0002546788019933867
Fstat=165404.9416417848
a= 4.150089508137071
b= -0.4492580212321198
c= 0.01941687856270382
d= -11.52707443126621
e= 15.50218588635162
/*-----*/
if ((x >= 0) && (x <= 4))
y=1.000283537050325+x*(-0.1075370910973749+
x*(0.02659514232448773+x*(-0.005855030915222600+
x*0.0005076645084344239)));
else
{
x1=x;
x2=x*x;
x3=1.0/x;
x4=1.0/(x*x);
y=4.150089508137071-0.4492580212321198*x1
+0.01941687856270382*x2-11.52707443126621*x3
+15.50218588635162*x4;
}
return(y);
}

/*-----*/
double FC_Nivel8(double x)
/*-----*/
{
double y;
/*-----*/
Funci3n Utilizada para 0 <= Fuga de Sodio <= 4
X= fuga de sodio
Y= factor correcion
Eqn# 2040 y=a+bx+cx2+dx3
r2=0.9999833041590704
r2adj=0.9999772329441869
StdErr=0.0002216690420837742
Fstat=239576.6247118936
a= 0.999962002352913

```

```

b= -0.05672030244773998
c= 0.00617527620726104
d= -0.0004851421538920275

```

```

Funci3n utilizada para 4 <= Fuga de Sodio <= 8
X= fuga de sodio

```

```

Y= factor correcion

```

```

Eqn# 6001 y=a+bx+cx2+dx3+ex4

```

```

r2=0.9999814343151208

```

```

r2adj=0.9999729953674484

```

```

StdErr=0.0001951247531051155

```

```

Fstat=161585.4369208678

```

```

a= 0.8071387112938201

```

```

b= 0.08623267728088095

```

```

c= -0.03236318636882039

```

```

d= 0.003930055815244988

```

```

e= -0.0001752970366055273

```

```

if ((x >= 0) && (x <= 4))

```

```

    y=0.9999620023529130+x*(-0.05672030244773998+
        x*(0.006175276207261040+x*-0.0004851421538920275));

```

```

else

```

```

    y=0.8071387112938201+x*(0.08623267728088095+
        x*(-0.03236318636882039+x*(0.003930055815244988+
            x*-0.0001752970366055273)));

```

```

return(y);

```

```

}

```

```

/*-----*/

```

```

int PrmtrsCationAnion(void)

```

```

/*-----*/

```

```

{

```

```

    strcpy(temp,"Cationes");

```

```

    Ventana(1,10,8,69,atr2,2,temp);

```

```

    CreaVentanaMensaje(0,2,"ESC", "Salir");

```

```

    DespliegaCadena("ppm Sodio :",atr1,2,12);

```

```

    DespliegaCadena("ppm Potasio :",atr1,3,12);

```

```

    DespliegaCadena("ppm Calcio :",atr1,4,12);

```

```

    DespliegaCadena("ppm Magnesio :",atr1,5,12);

```

```

    DespliegaCadena("ppm Manganeso :",atr1,6,12);

```

```

    DespliegaCadena("ppm Hierro :",atr1,7,12);

```

```

    strcpy(temp,"Aniones");

```

```

    Ventana(9,10,16,69,atr2,2,temp);

```

```

    DespliegaCadena("ppm CO3 :",atr1,10,12);

```

```

    DespliegaCadena("ppm HCO3 :",atr1,11,12);

```

```

    DespliegaCadena("ppm Cloruro :",atr1,12,12);

```

```

    DespliegaCadena("ppm Nitrato :",atr1,13,12);

```

```

    DespliegaCadena("ppm Hidr3xido :",atr1,14,12);

```

```

    DespliegaCadena("ppm Sulfato :",atr1,15,12);

```

```

    strcpy(temp,"");

```

```

    Ventana(17,10,20,69,atr2,2,temp);

```

```

    DespliegaCadena("ppm Silice :",atr1,18,12);

```

```

    DespliegaCadena("ppm CO2 :",atr1,19,12);

```

```

    NumeroCorrecto = -1;

```

```

    PPMsodio = RetomaDb(1,2,28,8);

```

```

    if (!NumeroCorrecto) return(-1); /* presion3 ESC */

```

```

    NumeroCorrecto = -1;

```

```

    PPMPotasio = RetomaDb(1,3,28,8);

```

```

    if (!NumeroCorrecto) return(-1); /* presion3 ESC */

```

```

    NumeroCorrecto = -1;

```

```

    PPMCalcio = RetomaDb(1,4,28,8);

```

```

    if (!NumeroCorrecto) return(-1); /* Presion3 ESC */

```

```

    NumeroCorrecto = -1;

```

```

    PPMMagnesio = RetomaDb(1,5,28,8);

```

```

if (!NumeroCorrecto) return(-1); /* Presion ESC */

NumeroCorrecto = -1;
PPMManganeso = RetomaDb(1,6,28,8);
if (!NumeroCorrecto) return(-1); /* Presion ESC */

NumeroCorrecto = -1;
PPMHierro = RetomaDb(1,7,28,8);
if (!NumeroCorrecto) return(-1); /* Presion ESC */

NumeroCorrecto = -1;
PPMCO3 = RetomaDb(1,10,28,8);
if (!NumeroCorrecto) return(-1); /* presion ESC */

NumeroCorrecto = -1;
PPMHCO3 = RetomaDb(1,11,28,8);
if (!NumeroCorrecto) return(-1); /* presion ESC */

NumeroCorrecto = -1;
PPMCloruro = RetomaDb(1,12,28,8);
if (!NumeroCorrecto) return(-1); /* presion ESC */

NumeroCorrecto = -1;
PPMNitrato = RetomaDb(1,13,28,8);
if (!NumeroCorrecto) return(-1); /* presion ESC */

NumeroCorrecto = -1;
PPMHidroxido = RetomaDb(1,14,28,8);
if (!NumeroCorrecto) return(-1); /* Presion ESC */

NumeroCorrecto = -1;
PPMSulfato = RetomaDb(1,15,28,8);
if (!NumeroCorrecto) return(-1); /* Presion ESC */

NumeroCorrecto = -1;
PPMSilice = RetomaDb(1,18,28,8);
if (!NumeroCorrecto) return(-1); /* presion ESC */

NumeroCorrecto = -1;
PPMCO2 = RetomaDb(1,19,28,8);
if (!NumeroCorrecto) return(-1); /* presion ESC */

FactorY = PPMMagnesio + PPMCalcio + PMPotasio + PPMSodio
          + PPMManganeso + PPMHierro;
PctSodio = (PPMSodio + PMPotasio) / FactorY * 100;
PctAlcalinidad = (PPMCO3 + PPMHCO3) / FactorY * 100;
FactorZ = PPMCloruro + PPMSulfato + PPMNitrato
          + PPMSilice + PPMCO2 + PPMHidroxido + PPMCO3
          + PPMHCO3;
PctAcidoCarbonico = (PPMCO3 + PPMHCO3 + PPMCO2) / FactorZ * 100;
PctIonesMonovalentes = (PPMCloruro + PPMNitrato) / FactorZ * 100;
PctSilice = (PPMSilice / FactorZ) * 100;

FactorY = FactorY / 17.1;
FactorZ = FactorZ / 17.1;

return(0);
}

/*-----*/
int UnidadCationica()
/*-----*/
{
    int i;
    char Respuesta;

    BorraBloque(0,0,18,79,attr2);
    Ventana(0,0,5,79,attr2,1,temp);

```

```

sprintf(NumeroEnCadena,"ppm Ca : %5.3f",PPMCalcio);
DespliegaCadena(NumeroEnCadena,attr2,1,2);
sprintf(NumeroEnCadena,"ppm Mg : %5.3f",PPMMagnesio);
DespliegaCadena(NumeroEnCadena,attr2,1,57);
sprintf(NumeroEnCadena,"ppm Na : %5.3f",PPMSodio);
DespliegaCadena(NumeroEnCadena,attr2,2,2);
sprintf(NumeroEnCadena,"ppm K : %5.3f",PPMPotasio);
DespliegaCadena(NumeroEnCadena,attr2,2,57);
sprintf(NumeroEnCadena,"% Na : %5.3f",PctSodio);
DespliegaCadena(NumeroEnCadena,attr2,3,2);
sprintf(NumeroEnCadena,"% Alc. : %5.3f",PctAlcalinidad);
DespliegaCadena(NumeroEnCadena,attr2,3,57);
sprintf(NumeroEnCadena,"Factor Y: %5.3f",FactorY);
DespliegaCadena(NumeroEnCadena,attr2,4,2);

/* ----- Pct's Exactos para calcular la Capacidad de Resina ----- */
if (PctAlcalinidad == 0)
  CapacidadDeResina = CapacidadResinaPct0(PctSodio);
else
  if (PctAlcalinidad == 25)
    CapacidadDeResina = CapacidadResinaPct25(PctSodio);
  else
    if (PctAlcalinidad == 50)
      CapacidadDeResina = CapacidadResinaPct50(PctSodio);
    else
      if (PctAlcalinidad == 75)
        CapacidadDeResina = CapacidadResinaPct75(PctSodio);
      else
        if (PctAlcalinidad == 100)
          CapacidadDeResina = CapacidadResinaPct100(PctSodio);

/* ----- Interpolaci#n para calcular la Capacidad de Resina ----- */
if ((PctAlcalinidad > 0) && (PctAlcalinidad < 25))
{
  DiferenciaDeCapacidad = CapacidadResinaPct25(PctSodio)
    - CapacidadResinaPct0(PctSodio);
  CapacidadDeResina = CapacidadResinaPct0(PctSodio) +
    (PctAlcalinidad * DiferenciaDeCapacidad / 25);
}
else
  if ((PctAlcalinidad > 25) && (PctAlcalinidad < 50))
  {
    DiferenciaDeCapacidad = CapacidadResinaPct50(PctSodio)
      - CapacidadResinaPct25(PctSodio);
    CapacidadDeResina = CapacidadResinaPct25(PctSodio) +
      ((PctAlcalinidad-25)*DiferenciaDeCapacidad/25);
  }
else
  if ((PctAlcalinidad > 50) && (PctAlcalinidad < 75))
  {
    DiferenciaDeCapacidad = CapacidadResinaPct75(PctSodio)
      - CapacidadResinaPct50(PctSodio);
    CapacidadDeResina = CapacidadResinaPct50(PctSodio) +
      ((PctAlcalinidad-50)*DiferenciaDeCapacidad/25);
  }
else
  if ((PctAlcalinidad > 75) && (PctAlcalinidad < 100))
  {
    DiferenciaDeCapacidad = CapacidadResinaPct100(PctSodio)
      - CapacidadResinaPct75(PctSodio);
    CapacidadDeResina = CapacidadResinaPct75(PctSodio) +
      ((PctAlcalinidad-75)*DiferenciaDeCapacidad/25);
  }

CapacidadDeResina = CapacidadDeResina * 35.32;
Ventana(6,1,8,39,attr2,2,temp);
sprintf(NumeroEnCadena,"Capacidad de Resina %5.3f Kgr/m3",CapacidadDeResina);
DespliegaCadena(NumeroEnCadena,attr2,7,2);

CapacidadPorUnidad = CapacidadDeResina * 1000;

```

```

Ventana(6,41,8,79,atr2,2,temp);
sprintf(NumeroEnCadena,"Capacidad(gr/m3): %5.3f",CapacidadPorUnidad);
DespliegaCadena(NumeroEnCadena,atr2,7,43);

Ventana(9,1,11,39,atr2,2,temp);
DespliegaCadena("Galones de Agua a Tratar:",atr1,10,2);
NumeroCorrecto = -1;
GalonesH2O = RetornaDb(1,10,28,8);
if (!NumeroCorrecto) return(-1);

VolumenResina = GalonesH2O * FactorY / (CapacidadPorUnidad * 35.32);
Ventana(9,41,11,79,atr4,2,temp);
sprintf(NumeroEnCadena,"Volumen de Resina: %5.3f m3",VolumenResina);
DespliegaCadena(NumeroEnCadena,atr4,10,43);

BorraBloque(21,0,23,79,atr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,atr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",atr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getch();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(19,3,19,70,atr2);
        DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",atr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5,"***** Unidad Cationica *****\n");
    fprintf(pfile5,"Volumen de Resina Cationica: %5.3f m3 \n\n",VolumenResina);
    fclose(pfile5);
}

CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
HacerFugaSodio = 1;
HacerUniAnionica = 1;
return(0);
}

/*-----*/
int UnidadAnionica()
/*-----*/
{
    int i;
    char Respuesta,MensajeAnionico[80];

    if (!HacerUniAnionica)
    {
        CreaVentanaMensaje(1,1,"No se ha efectuado la Unidad Catiónica. Presione ESC.");
        while(getch() != ESCAPE);
        return(0);
    }

    if (InicializaMatrizAnionica())
    {
        CreaVentanaMensaje(1,1,MENSAJE_ERR04);
        while(getch() != ESCAPE);
        return(0);
    }
}

BorraBloque(0,0,18,79,atr2);
Ventana(0,0,7,79,atr2,1,temp);

```

```

sprintf(NumeroEnCadena,"ppm Cloruro      : %5.3f",PPMCloruro);
DespliegaCadena(NumeroEnCadena,attr2,1,2);
sprintf(NumeroEnCadena,"ppm Sulfato      : %5.3f",PPMSulfato);
DespliegaCadena(NumeroEnCadena,attr2,1,45);
sprintf(NumeroEnCadena,"ppm Nitrato      : %5.3f",PPMNitrato);
DespliegaCadena(NumeroEnCadena,attr2,2,2);
sprintf(NumeroEnCadena,"ppm Hidróxido    : %5.3f",PPMHidroxido);
DespliegaCadena(NumeroEnCadena,attr2,2,45);
sprintf(NumeroEnCadena,"ppm Carbonato    : %5.3f",PPMCO3);
DespliegaCadena(NumeroEnCadena,attr2,3,2);
sprintf(NumeroEnCadena,"ppm Bicarbonato  : %5.3f",PPMHCO3);
DespliegaCadena(NumeroEnCadena,attr2,3,45);
sprintf(NumeroEnCadena,"ppm Silice      : %5.3f",PPMSilice);
DespliegaCadena(NumeroEnCadena,attr2,4,2);
sprintf(NumeroEnCadena,"ppm CO2        : %5.3f",PPMCO2);
DespliegaCadena(NumeroEnCadena,attr2,4,45);
sprintf(NumeroEnCadena,"Factor Z        : %5.3f",FactorZ);
DespliegaCadena(NumeroEnCadena,attr2,5,2);
sprintf(NumeroEnCadena,"%% Acido Carbonico: %5.3f",PctAcidoCarbonico);
DespliegaCadena(NumeroEnCadena,attr2,5,45);
sprintf(NumeroEnCadena,"%% Iones Monoval.: %5.3f",PctIonesMonovalentes);
DespliegaCadena(NumeroEnCadena,attr2,6,2);
sprintf(NumeroEnCadena,"%% Silice        : %5.3f",PctSilice);
DespliegaCadena(NumeroEnCadena,attr2,6,45);

Ventana(8,1,10,79,attr2,2,temp);
strcpy(MensajeAnionico,"Nivel de Regeneración(lbs NaOH/pie3):");
DespliegaCadena(MensajeAnionico,attr2,9,3);
NumeroCorrecto = -1;
NivelRegeneracion = 0;
CreaVentanaMensaje(1,1,"Valores Permitidos para el Nivel de Regeneración: 4,6,8.");
while (NumeroCorrecto < 0)
{
    NumeroCorrecto = CapturaNumero(NumeroEnCadena,9,41,1,attr0);
    if (NumeroCorrecto>0) NivelRegeneracion = atoi(NumeroEnCadena);
    if ((NivelRegeneracion != 4) && (NivelRegeneracion != 6)
        && (NivelRegeneracion != 8))
    {
        CreaVentanaMensaje(1,1,MENSAJE_ERR03);
        while (!kbhit());
        CreaVentanaMensaje(0,2,"ESC", "Salir");
        NumeroCorrecto = -1;
    }
}

CreaVentanaMensaje(0,2,"ESC", "Salir");

BuscaFactores();

if (NivelRegeneracion == 4) CapacidadDelIntercambio = 7800;
else
    if (NivelRegeneracion == 6) CapacidadDelIntercambio = 15200;
    else
        if (NivelRegeneracion == 8) CapacidadDelIntercambio = 16100;

CapacidadCorregida = CapacidadDelIntercambio * Factor1 * Factor2 * Factor3 * 35.32;

Ventana(11,1,13,39,attr2,2,temp);
DespliegaCadena("Galones de Agua a tratar:",attr1,12,2);
NumeroCorrecto = -1;
GalonesH2O = RetornaDbl(1,12,28,8);
if (!NumeroCorrecto) return(-1);

VolumenResina = GalonesH2O * FactorZ / CapacidadCorregida;

Ventana(11,40,13,79,attr4,2,temp);
VolumenResina = VolumenResina / 35.32;
sprintf(NumeroEnCadena,"Volumen de Resina Aniónica: %5.3f m3",VolumenResina);
DespliegaCadena(NumeroEnCadena,attr4,12,41);

```

```

Ventana(14,1,16,39,attr2,2,temp);
DespliegaCadena("Volumen del Tanque m3:",attr1,15,2);
NumeroCorrecto = -1;
VolumenTanque = RetornaDb(1,15,27,8);
if (!NumeroCorrecto) return(-1);

GalH2ODecat = CapacidadPorUnidad * VolumenTanque / FactorY;
Ventana(14,40,16,79,attr4,2,temp);
sprintf(NumeroEnCadena,"Gal. de H2O Decationizada: %5.3f",GalH2ODecat);
DespliegaCadena(NumeroEnCadena,attr4,15,41);

GalNeto = GalH2ODecat - (GalonesH2O * VolumenTanque);
Ventana(17,1,19,39,attr4,2,temp);
sprintf(NumeroEnCadena,"Galonaje Neto: %5.3f",GalNeto);
DespliegaCadena(NumeroEnCadena,attr4,18,2);

GalUANionico = CapacidadCorregida * VolumenTanque / FactorZ;
Ventana(17,40,19,79,attr4,2,temp);
sprintf(NumeroEnCadena,"Galones Unidad Anionica: %5.3f",GalUANionico);
DespliegaCadena(NumeroEnCadena,attr4,18,41);

BorraBloque(21,0,23,79,attr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,attr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getch();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(19,3,19,70,attr2);
        DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5,"\n\n***** Unidad Anionica *****\n");
    fprintf(pfile5,"Volumen de Resina Anionica : %5.3f m3 \n",VolumenResina);
    fprintf(pfile5,"Galones de Agua Decationizada: %5.3f \n",GalH2ODecat);
    fprintf(pfile5,"Galonaje Neto : %5.3f \n",GalNeto);
    fprintf(pfile5,"Galones de Unidad Anionica : %5.3f \n",GalUANionico);
    fclose(pfile5);
}

CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
HacerFugaSilice = 1;
return(0);
}

/*-----*/
int FugaSodio(void)
/*-----*/
{
    if (!HacerFugaSodio)
    {
        CreaVentanaMensaje(1,1,"No se ha efectuado la Unidad Catiónica. Presione ESC.");
        while(getch() != ESCAPE);
        return(0);
    }
}

BorraBloque(0,0,18,79,attr2);
Ventana(0,0,3,79,attr2,1,temp);

```

```

sprintf(NumeroEnCadena,"%% Na : %5.3f",PctSodio);
DespliegaCadena(NumeroEnCadena,atr2,1,2);
sprintf(NumeroEnCadena,"%% Alc.: %5.3f",PctAlcalinidad);
DespliegaCadena(NumeroEnCadena,atr2,1,57);
sprintf(NumeroEnCadena,"Factor Y: %5.3f",FactorY);
DespliegaCadena(NumeroEnCadena,atr2,2,2);

/*----- Calculo con Porcentajes Exactos -----*/
if (PctAlcalinidad == 0)
  PctCationes = PctCationesPct0(PctSodio);
else
  if (PctAlcalinidad == 25)
    PctCationes = PctCationesPct25(PctSodio);
  else
    if (PctAlcalinidad == 50)
      PctCationes = PctCationesPct50(PctSodio);
    else
      if (PctAlcalinidad == 75)
        PctCationes = PctCationesPct75(PctSodio);

/*----- Calculo con Interpolacion -----*/
if ((PctAlcalinidad > 0) && (PctAlcalinidad < 25))
{
  DiferenciaDeCapacidad = PctCationesPct25(PctSodio)
    - PctCationesPct0(PctSodio);
  PctCationes = PctCationesPct0(PctSodio) +
    (PctAlcalinidad * DiferenciaDeCapacidad / 25);
}
else
  if ((PctAlcalinidad > 25) && (PctAlcalinidad < 50))
  {
    DiferenciaDeCapacidad = PctCationesPct50(PctSodio)
      - PctCationesPct25(PctSodio);
    PctCationes = PctCationesPct25(PctSodio) +
      ((PctAlcalinidad-25)*DiferenciaDeCapacidad/25);
  }
  else
    if ((PctAlcalinidad > 50) && (PctAlcalinidad < 75))
    {
      DiferenciaDeCapacidad = PctCationesPct75(PctSodio)
        - PctCationesPct50(PctSodio);
      PctCationes = PctCationesPct50(PctSodio) +
        ((PctAlcalinidad-50)*DiferenciaDeCapacidad/25);
    }

if (PctCationes < 0)
  FugaDeSodio = 0;
else
  FugaDeSodio = PctCationes * FactorY * 17.1 / 100;
  FugaDeSodioEntera = ceil(FugaDeSodio);

Ventana(5,20,8,59,atr4,2,temp);
sprintf(NumeroEnCadena,"Fuga de Sodio (ppm Na): %5.3f",FugaDeSodio);
DespliegaCadena(NumeroEnCadena,atr4,6,26);
sprintf(NumeroEnCadena,"Fuga de Sodio (ppm CaCO3): %5.0f",FugaDeSodioEntera);
DespliegaCadena(NumeroEnCadena,atr4,7,26);

BorraBloque(21,0,23,79,atr2);
strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
  Ventana(21,0,23,79,atr2,1,temp);
  DespliegaCadena("Desea Imprimir [S/N]? ",atr1,22,3);
  Cursor(22,25);
  Imprimir[0] = getche();
  if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
  // impresion de encabezados en el impresor

```

```

if ((pfile5=fopen("LPT1","w"))==NULL)
{
    BorraBloque(19,3,19,70,attr2);
    DespliegaCadena("ERROR!: No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
    while(1kbhit());
    return(0);
}
fprintf(pfile5,"n\n***** Fuga de Sodio *****n\n");
fprintf(pfile5,"Fuga de Sodio : %5.3f (ppm Na) n",FugaDeSodio);
fprintf(pfile5,"Fuga de Sodio : %5.3f (ppm CaCO3) n\n",FugaDeSodioEntera);
fclose(pfile5);
}
CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
return(0);
}

/*-----*/
int FugaSilice(void)
/*-----*/
{
    BorraBloque(0,0,18,79,attr2);
    Ventana(0,0,4,79,attr2,1,temp);

    if (!HacerFugaSilice)
    {
        CreaVentanaMensaje(1,1,"No se ha Efectuado la Unidad Aniónica. Presione ESC.");
        while(getch() != ESCAPE);
        return(0);
    }

    sprintf(NumeroEnCadena,"%% S;lice      : %5.3f",PctSilice);
    DespliegaCadena(NumeroEnCadena,attr2,1,2);
    sprintf(NumeroEnCadena,"Fuga de Sodio      : %5.3f",FugaDeSodio);
    DespliegaCadena(NumeroEnCadena,attr2,2,2);
    sprintf(NumeroEnCadena,"Nivel de Regeneraci#n: %1d",NivelRegeneracion);
    DespliegaCadena(NumeroEnCadena,attr2,3,2);

    switch (NivelRegeneracion)
    {
        case 4:
        {
            FugaPromedioSilice = FPS_Nivel4(PctSilice);
            FactorCorreccion = FC_Nivel4(FugaDeSodio);
            break;
        }
        case 6:
        {
            FugaPromedioSilice = FPS_Nivel6(PctSilice);
            FactorCorreccion = FC_Nivel6(FugaDeSodio);
            break;
        }
        case 8:
        {
            FugaPromedioSilice = FPS_Nivel8(PctSilice);
            FactorCorreccion = FC_Nivel8(FugaDeSodio);
            break;
        }
    }

    Ventana(5,0,9,79,attr4,1,temp);
    FugaSiliceCorregida = FugaPromedioSilice / FactorCorreccion;
    sprintf(NumeroEnCadena,"Fuga Prom. de S;lice      : %5.3f ppm",FugaPromedioSilice);
    DespliegaCadena(NumeroEnCadena,attr4,6,2);
    sprintf(NumeroEnCadena,"Factor de Correcci#n      : %5.3f ",FactorCorreccion);
    DespliegaCadena(NumeroEnCadena,attr4,7,2);
    sprintf(NumeroEnCadena,"Fuga de S;lice Corregida: %5.3f ppm",FugaSiliceCorregida);
    DespliegaCadena(NumeroEnCadena,attr4,8,2);

    BorraBloque(21,0,23,79,attr2);

```

```

strcpy(Imprimir,"Q");
while(Imprimir[0] != 'S' && Imprimir[0] != 'N')
{
    Ventana(21,0,23,79,attr2,1,temp);
    DespliegaCadena("Desea Imprimir [S/N]? ",attr1,22,3);
    Cursor(22,25);
    Imprimir[0] = getche();
    if (Imprimir[0] == 27) return(-1);
}
if (Imprimir[0] == 'S')
{
    // impresion de encabezados en el impresor
    if ((pfile5=fopen("LPT1","w"))==NULL)
    {
        BorraBloque(19,3,19,70,attr2);
        DespliegaCadena("ERROR! No se puede acceder a impresora. Presione cualquier tecla.",attr1,19,3);
        while(!kbhit());
        return(0);
    }
    fprintf(pfile5,"\n\n***** Fuga de Silice *****\n");
    fprintf(pfile5,"Fuga Promedio de Silice : %5.3f (ppm) \n",FugaPromedioSilice);
    fprintf(pfile5,"Factor de Correccion : %5.3f \n",FactorCorreccion);
    fprintf(pfile5,"Fuga de Silice Corregida: %5.3f (ppm) \n\r",FugaSiliceCorregida);
    fclose(pfile5);
}
CreaVentanaMensaje(0,2,"ESC","Salir");
while(getch() != ESCAPE);
return(0);
}
/*-----*/
/*-----*/
void Intercambiolonico()
/*-----*/
{
    /*-- Definicion temporal de atributos --*/
    attr3=WHITE(5<<4); /* Atributos del Mensaje abajo en la Pantalla */
    attr2=WHITE(BLACK<<4); /* Atributos del Marco de Ventana Blanco/Negro */
    attr1=WHITE(BLACK<<4); /* Atributos del Texto Desactivado Blanco/Negro */
    attr0=WHITE(BLUE<<4); /* Atributos del Texto Activado Blanco/Azul */
    BorraBloque(0,0,23,79,attr1);
    /* --Este atributo es propio --*/
    attr4=WHITE(MAGENTA<<4); /* Atributos del Texto para respuestas Blanco/Cyan */
    /*-----*/
    while(!VarRetorno)
    {
        Reconstruye = 1;
        HacerFugaSilice = 0;
        HacerFugaSodio = 0;
        Ventana(21,0,23,79,attr1,1,temp);
        VarRetorno = PrmtrsCationAnion();
        if (VarRetorno)
        {
            BorraBloque(0,0,23,79,attr2);
            SalirOpcion_12 = 1;
        }
        else SalirOpcion_12 = 0;
        while(!SalirOpcion_12)
        {
            if (Reconstruye)
            {
                BorraBloque(0,0,20,79,attr2);
                SalirOpcion_12 = 0;
                PosicionMenu = 121;
                strcpy(temp,"Intercambio Ionico");
                CreaVentanaMensaje(1,1,MENSAJE_121);
                Ventana(8,21,13,56,attr2,2,temp);
                DespliegaCadena(OPCION_121,attr0,9,22);
                DespliegaCadena(OPCION_122,attr1,10,22);
                DespliegaCadena(OPCION_123,attr1,11,22);
            }
        }
    }
}

```



```

        DespliegaCadena(OPCION_124,attr0,12,22);
        PosicionMenu = 124;
        break;
    }
    case 124:
    {
        DespliegaCadena(OPCION_124,attr1,12,22);
        CreaVentanaMensaje(1,1,MENSAJE_121);
        DespliegaCadena(OPCION_121,attr0,9,22);
        PosicionMenu = 121;
        break;
    }
}
break;
}
case ENTER:
{
    switch(PosicionMenu)
    {
        case 121:
        {
            UnidadCationica();
            BorraBloque(0,0,20,79,attr2);
            Reconstruye = 1;
            break;
        }
        case 122:
        {
            FugaSodio();
            BorraBloque(0,0,20,79,attr2);
            Reconstruye = 1;
            break;
        }
        case 123:
        {
            UnidadAnionica();
            BorraBloque(0,0,20,79,attr2);
            Reconstruye = 1;
            break;
        }
        case 124:
        {
            FugaSilice();
            BorraBloque(0,0,23,79,attr2);
            Reconstruye = 1;
            break;
        }
    }
}
break;
}
case ESCAPE:
{
    BorraBloque(0,0,20,79,attr2);
    SalirOpcion_12 = 1;
}
}
}
}

void main(void)
{
    Intercambiolonico();
}

```

CAPITULO V

MANUAL DEL USUARIO

El presente manual ofrece los procedimientos necesarios para el uso adecuado del programa elaborado en lenguaje C que permite estimar los parámetros de diseño de los sistemas de intercambio iónico y lagunas de estabilización; así como también el cálculo de las constantes para una isoterma de adsorción y el área transversal y profundidad de un sedimentador.

Los programas utilizan todo el ingreso de datos en el sistema internacional de unidades. Asegurarse que todos sus cálculos ocupen dicho sistema.

Consideraciones adicionales para el uso del programa

1. Para las decisiones de Si ó No (S/N) se oprime la letra en mayúscula respectiva con el propósito que el usuario analice la pregunta.
2. Oprimir la tecla Enter después de cada orden o ingreso de datos.
3. Para salir de cualquier pantalla, presionar la tecla escape (Esc).

PASOS PARA OPERAR LOS PROGRAMAS

- A. Encender la computadora, y esperar que aparezca el prompt (C:\ >).
- B. Introducir el disco que contiene los programas de los sistemas de tratamiento de aguas, en la unidad lectora (A o B).
- C. Digitar la letra de la unidad lectora (A: o B:) . Aparecerá A:\ > ó B:\ >.
- D. Escribir la palabra "menu". Presionar la tecla Enter para continuar.
- E. Se despliega una pantalla de presentación del programa. Pulsar la tecla Enter para continuar.
- F. Aparecerá la pantalla :

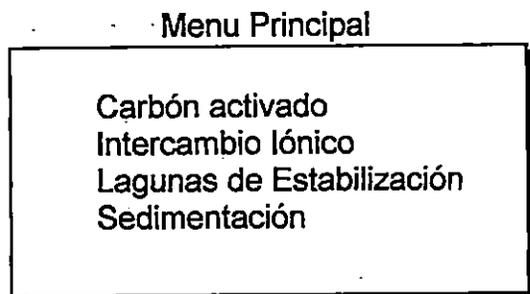


Figura 1. Pantalla Principal

Mediante las teclas de dirección (flecha arriba y flecha abajo) seleccionar el sistema de interés.

Cuando el cursor esta en la opción (cursor activado) en la parte inferior de la pantalla aparece un mensaje explicativo de dicha opción.

Presionar la tecla Enter para aceptar una opción.

OPCIONES DEL MENU PRINCIPAL

CARBON ACTIVADO

Esta opción permite evaluar las constantes empíricas de las ecuaciones de Freundlich y Langmuir para la adsorción en carbón activado. El programa esta diseñado para utilizar un número máximo de 15 pares de datos.

PROCEDIMIENTO

1. Seleccionar la Opción: Carbón Activado

Aparece una pantalla de presentación con los diferentes métodos de solución y las variables de contexto.

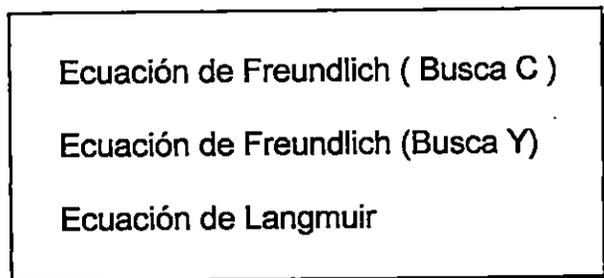


Figura 2. Pantalla del Menu Carbón Activado

En la parte inferior aparece un mensaje explicativo de la opción en la que se encuentra el cursor.

Para salir de esta pantalla, presionar la tecla Escape. Vuelve al menu principal.

EJEMPLO DE USO DE LA OPCIÓN CARBÓN ACTIVADO (CASO I):

Se desea conocer las constantes empíricas de la ecuación de Freundlich a partir de los datos de laboratorio siguientes:

M	X/M
0.000	9.6
0.001	8.6
0.004	6.3
0.008	4.3
0.02	1.7
0.04	0.7

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Carbón Activado.

Seleccionar: Ecuación de Freundlich (Busca C)

La letra C que muestra la pantalla quiere decir que se encontrará la concentración (C) a partir de los datos de dosis de Carbón (M) y de la cantidad adsorbida por dosis de carbón (X/M).

Aparece una pantalla con M, Y, C y sus respectivos logaritmos.

2. Introducir el número total de pares datos -1. Esto excluye el par de datos en los cuales la dosis de carbón es cero. Número de datos : 5
3. Introducir la Concentración inicial, es decir el dato (X/M) cuando la dosis de carbón es cero. Valor inicial (Co): 9.6
4. Ingresar los datos experimentales restantes. No incluir el par donde la concentración es cero.

Después de ingresar cada par de datos aparecerán los cálculos de concentración y sus respectivos logaritmos.

5. Al finalizar el ingreso de los datos aparece en la parte inferior: Datos Procesados, presionar cualquier tecla para continuar.
6. Aparece la solución a la ecuación con los valores de las constantes empíricas. (K, 1/n).
7. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel.

8. Presionar la tecla Escape para salir. Se retorna al menu de opciones de Carbón Activado (Ver Figura 2).
9. Pulsar la tecla Escape para volver al menu principal (Ver Figura 1).

EJEMPLO DE USO DE LA OPCIÓN CARBÓN ACTIVADO (CASO 2):

Se desea conocer las constantes empíricas de la ecuacion de Freundlich a partir de los datos de laboratorio siguientes:

M	C
0.00	7.70
0.05	3.67
0.1	2.20
0.3	0.87
1.0	0.25

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Carbón Activado

Seleccionar: Ecuación de Freundlich (Busca Y)

La letra Y significa que se encontrará la cantidad adsorbida por dosis de carbón (X/M) a partir de la dosis de carbón (M) y la concentración (C).

Aparece una pantalla con M, C, Y y sus respectivos logaritmos.

2. Introducir el número total de pares de datos -1. Esto excluye el par de datos en los cuales la dosis de carbón es cero. Número de datos : 4

3. Introducir la Concentración inicial, es decir el dato cuando la dosis de carbón es cero. Valor inicial (C_0): 7.7
4. Ingresar los datos experimentales restantes. No incluir el par donde la concentración es cero.

Después de ingresar cada par de datos aparecerán los cálculos de concentración y sus respectivos logaritmos.

5. Al finalizar el ingreso de los datos aparece en la parte inferior: Datos Procesados, presionar cualquier tecla para continuar.
6. Aparece la solución de la ecuación con los valores de las constantes empíricas. (K , $1/n$).
7. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel.

8. Presionar la tecla Escape para salir. Se retorna al menu de opciones de Carbón Activado (Ver Figura 2).
9. Pulsar la tecla Escape para volver al menu principal (Ver Figura 1.).

EJEMPLO DE USO DE LA OPCIÓN CARBÓN ACTIVADO (CASO 3):

Se desea conocer las constantes empíricas de la ecuación de Langmuir a partir de los datos de laboratorio siguientes:

X/M	C
0.133	10
0.187	20
0.220	30

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Carbón Activado
 Seleccionar: Ecuación de Langmuir

Aparece una pantalla con las variables X/M , C , C/Y , C^2/Y , C^2

2. Introducir el número de pares de datos. Número de datos: 3
3. Ingresar los datos experimentales de la tabla.
4. Al finalizar el ingreso de los datos aparece en la parte inferior: Datos Procesados, presione cualquier tecla para continuar.
5. Aparece la solución a la ecuación con los valores de las constantes empírica (a,b).
6. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel.

7. Presionar la tecla Escape para salir. Se retorna al menu de opciones de Carbón Activado.(Ver Figura 2).
8. Pulsar la tecla Escape para volver al menu principal (Ver Figura 1).

INTERCAMBIO IONICO

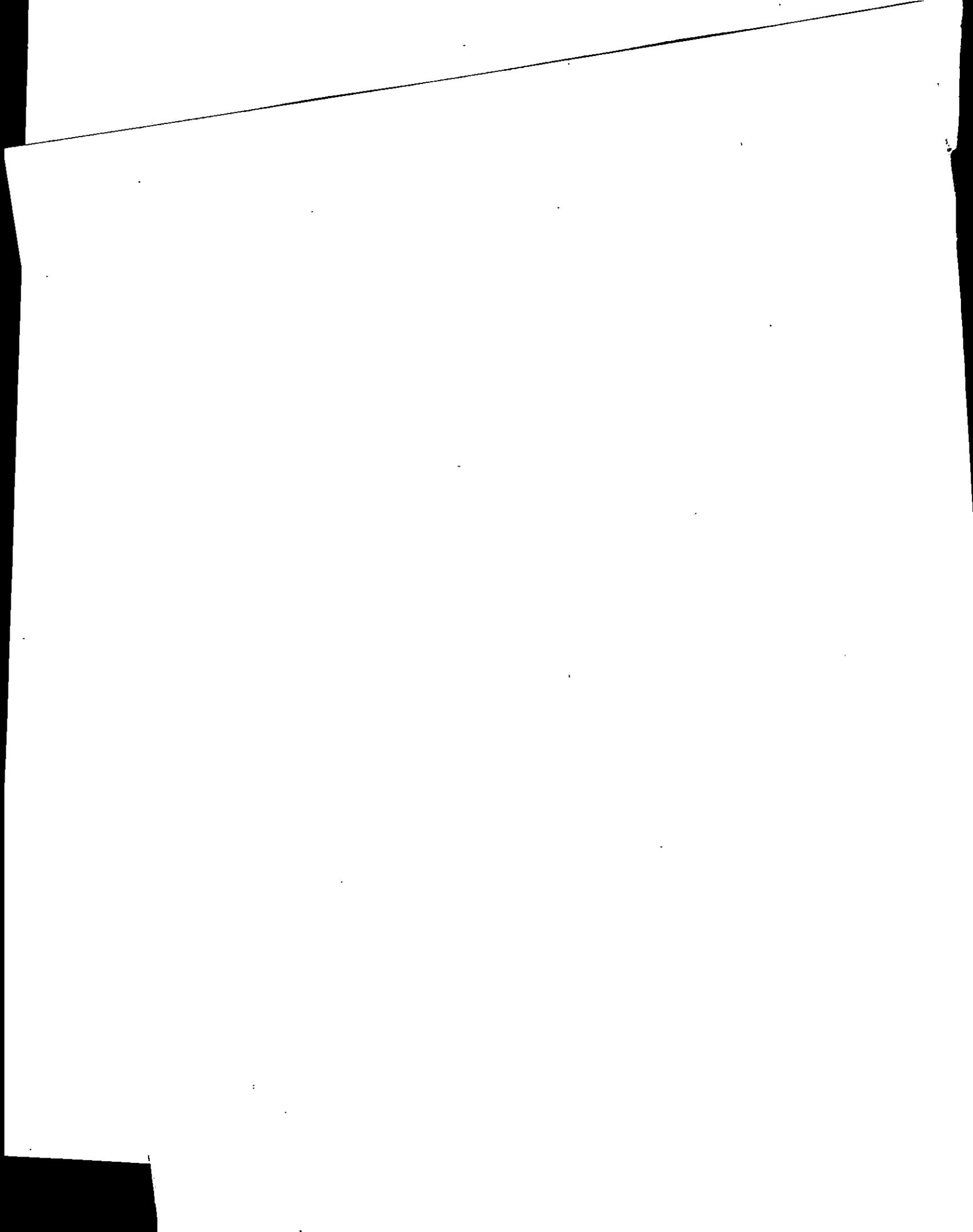
Esta opción permite evaluar los parámetros de diseño de un desmineralizador tales como los volúmenes de la resina catiónica y aniónica y las fugas de sodio y sílice.

EJEMPLO DE LA OPCIÓN: INTERCAMBIO IÓNICO

Determinar el volumen o cantidad de resina (aniónica y catiónica) necesaria para producir 3,532 galones de agua desmineralizada de un agua suministrada. Además hacer la corrección de este volumen (si fuese necesario) con el objeto de adecuarlo al tamaño de tanques para resina disponibles en el mercado.

Finalmente debe calcularse el nuevo galonaje que teóricamente se obtendría con el nuevo volumen de resina.

Los datos necesarios para el cálculo se presentan a continuación:



En caso que un parámetro no se haya analizado, introducir el valor de cero.

2. Posterior a la introducción de los datos, aparece un sub-menu que muestra las opciones:

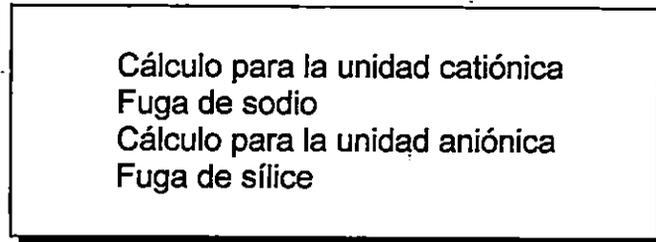


Figura 3. Pantalla de opciones para Intercambio Iónico

3. Seleccionar: Cálculo para la unidad catiónica.
Aparece un cuadro resumen de resultados.
4. Introducir los galones de agua a tratar

Galones de agua a tratar: 3532

Se evalúa el volumen de resina .

5. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel.

6. Pulsar la tecla Escape para volver al menu de Intercambio Iónico.

7. Seleccionar : Fuga de Sodio

Se evalúa la fuga de sodio para la unidad catiónica.

8. Pulsar la tecla Escape para volver al menu de Intercambio Iónico.

9. Seleccionar : Cálculo para la Unidad Aniónica

Aparece cuadro resumen de resultados y se solicita el nivel de regeneración para la unidad aniónica.

En la parte inferior se observan los valores permitidos para el nivel de regeneración 4,6,8

10. Introduzca el nivel de regeneración deseado

Nivel de Regeneración: 8

11. Introducir los galones de agua a tratar: 3532

Se evalúa el volumen de resina aniónica y se solicita el volumen del tanque (usar el anexo 1 para intercambio iónico) :0.0142

El valor leído debe ser el próximo superior al valor calculado por el programa.

12. Aparece un cuadro final con resultados.

13. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel.

14. Pulsar la tecla Escape para volver al menu de opciones de intercambio iónico.

15. Seleccione : Fuga de Sílice

Aparece un cuadro resumen y se evalúa la fuga promedio de sílice, factor de corrección , fuga de sílice corregida.

16. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel.

17. Presionar la tecla Escape tres veces para salir de la pantalla y volver al menu principal (Ver Figura 1).

LAGUNAS DE ESTABILIZACION

Esta opción permite evaluar los parámetros de diseño para lagunas de tipo aerobias, facultativas, anaerobias y con aireación mecánica.

PROCEDIMIENTO

1. Seleccionar la Opción: Lagunas de Estabilización

Aparece una pantalla de presentación con los diferentes métodos de solución.

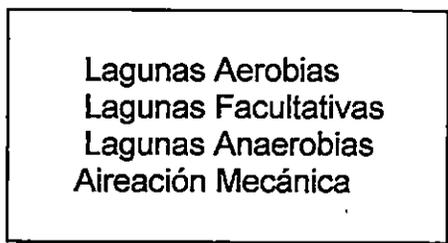


Figura 4. Pantalla de opciones para Lagunas de Estabilización

En la parte inferior aparece un mensaje explicativo de la opción en la que se encuentra el cursor.

Para salir de esta pantalla, presionar la tecla Escape. Vuelve al menu principal.

EJEMPLO DE USO DE LA OPCION LAGUNAS AEROBIAS

Diseñese una laguna de estabilización aerobia para tratar un caudal residual de 3,785 m³/ día. La laguna se construirá en Ahuachapán y se estima su funcionamiento a partir del mes de mayo. La profundidad será de 0.9 m.

Encontrar: a) Superficie, b) Carga Hidráulica, c) Carga Superficial, d) Tiempo de detención. Otros datos a considerar son los siguientes:

DBO a ser removida: 0.09 (Kg/ m³)

Temperatura del Estanque: 32°C

Porcentaje de Sustrato Remanente: 10

Factor de dispersión: 1

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Lagunas de Estabilización

Seleccionar : Lagunas Aerobias

Aparece una pantalla con el nombre Producción de Oxígeno.

2. Si desea encontrar el caudal, presionar la tecla S (mayúscula) e ingrese los datos de número de personas que aportan agua y la producción residual por persona. Si no se quiere encontrar el caudal, presionar la tecla N (mayúscula) e ingrese el valor, para este caso presionar N y el caudal (m³ / día) es 3785.
3. Introducir la DBO (Kg/m³) a ser removida, 0.09.

13. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel

14. Presionar la tecla Escape para salir de la pantalla. Se retorna al menu de opciones de Lagunas de Estabilización (Ver Figura 4).
15. pulsar la tecla Escape para volver al menu principal (Ver Figura 1).

EJEMPLO DE USO DE LA OPCION LAGUNAS FACULTATIVAS

Calcúlese lo siguiente para una Laguna facultativa: a) Carga Orgánica, b) Volumen, c) Superficie, d) tiempo de detención, e) Carga Superficial

Datos disponibles:

Temperatura del estanque: 32°C

Tiempo de detención a 35°C: 6 días

DBO del agua residual : 0.300 Kg/m³

Habitantes que producen residuos: 20,000

Volumen del agua residual por habitante: 0.15 m³/día

Caudal : 3,000 m³/ día

Profundidad : 1.75 m.

Temperatura media del agua en el mes más frío :10°C

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Lagunas de Estabilización

Seleccionar : Lagunas Facultativas

Aparece una pantalla en la que pide la temperatura del estanque (°C)

2. Introducir el dato de temperatura del estanque (°C), 32
3. Si conoce la velocidad de descomposición presionar la tecla S (mayúscula) e ingrese el dato. Sino, presionar la letra N (mayúscula) y aparecerá el resultado. En este caso presionar la tecla N.
4. Introducir el tiempo de detención (días) a 35°C, para este caso 6.
5. Introducir el valor de la DBO₅ del afluente, 0.3

6. Si conoce la DBO_5 del efluente presionar la letra S (mayúscula) e introducir el dato. Sino, presionar la letra N (mayúscula) y aparece el resultado de la DBO_5 del efluente; para este caso presionar S e ingrese el valor de 0.013.
7. Introducir el dato de caudal ($m^3/día$), 3000.
8. Introducir el dato del número de habitantes, 20000.
9. Para la producción de agua residual presionar S (mayúscula) si la conoce e ingrese el dato. Sino, presionar N y aparece el resultado; en este caso S y el dato es 0.15.
10. Ingresar el dato de la temperatura del mes más frío ($^{\circ}C$), 10.
11. Ingresar el dato de la DBO total (Kg/m^3), 0.3
Aparece el resultado de volumen de la laguna.
12. Ingresar el dato de la profundidad de la laguna (m), 1.75
Aparecen los resultados de Superficie, Carga Orgánica, Carga Superficial, Carga Hidráulica y tiempo de detención.
13. Si desea imprimir, pulsar la letra S en mayúscula.
Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel
14. Presionar la tecla Escape para salir de la pantalla. Se retorna al menu de opciones de Lagunas de Estabilización (Ver Figura 4).
15. Pulsar la tecla Escape para volver al menu principal (Ver Figura 1)

EJEMPLO DE USO DE LA OPCION LAGUNAS ANAEROBIAS

Los datos para el proceso de cálculo de un proyecto de estanque anaerobio de zona subtropical son:

Población= 8,000 habitantes

DBO_5 del agua afluente= $0.400 Kg/m^3$

DBO_5 del agua efluente= $0.240 Kg/m^3$

Caudal de agua residual= $0.07 m^3/día$

Profundidad= 2m

Exponente determinado por experimentación=4.8

Coefficiente de proyecto=6.0

Calcular: a) el tiempo de detención, b) Volumen del estanque, c) Superficie del estanque

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Lagunas de Estabilización

Seleccionar : Lagunas Anaerobias

Aparece una pantalla en la que se pide la Demanda Bioquímica de Oxígeno inicial.

2. Introducir el valor de la Demanda Bioquímica de Oxígeno inicial (Kg/m^3), 0.4.
3. Introducir el valor de la Demanda Bioquímica de Oxígeno final (Kg/m^3), 0.240.
4. Si conoce el caudal presionar la tecla S (mayúscula) e ingrese el dato. Sino, presionar la letra N (mayúscula) e ingrese los datos de número de personas que aportan agua y la producción residual por persona ($\text{m}^3/\text{día}$) para el caso presionar S y 560.

Aparece el resultado de volumen.

5. Introducir el dato de la profundidad (m) de la laguna, 2.

Aparecen los resultados de Superficie de la laguna, Carga Hidráulica y tiempo de detención.

6. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel

7. Presionar la tecla Escape para salir de la pantalla. Se retorna al menu de opciones de Lagunas de Estabilización (Ver Figura 4).
8. Pulsar la tecla Escape para volver al menu principal (Ver Figura 1).

EJEMPLO DE USO DE LA OPCION AIERACION MECANICA

Calcular el tamaño y número de aireadores mecánicos necesarios para reducir la DBO del agua residual afluyente en un 50%

Datos disponibles:

Caudal = 7,570 m³/ día

DBO₅ del afluente = 0.250 Kg/ m³

Nivel de oxígeno disuelto en el estanque = 0.0015 Kg/ m³

Nivel de saturación de oxígeno en el agua destilada a 20°C = 0.0092 Kg/ m³

Nivel de saturación de oxígeno en el estanque a temperatura T = 0.0102 Kg/ m³

Temperatura del agua a tratar = 15°C

$\alpha = 0.7$

Coefficiente de transferencia, $\alpha = 0.90$

Velocidad de reacción = 0.35 por día

Rendimiento del aireador por el fabricante = 1.6

Profundidad = 3m.

USO DEL PROGRAMA

1. Seleccionar Método de Solución: Lagunas de Estabilización

Seleccionar : Aireación Mecánica

Aparece una pantalla con el nombre de Aireación Mecánica y se solicita el valor del rendimiento del aireador por el fabricante.

2. Introducir los siguientes datos:

Rendimiento del aireador por el fabricante, 1.6.

Nivel de saturación de oxígeno en agua destilada a 20°C (Kg/m³), 0.0092

Nivel de oxígeno disuelto en el estanque (Kg/ m³), 0.0015

Nivel de saturación de oxígeno a temperatura°C (Kg/ m³), 0.0102

Coefficiente de transferencia de agua residual/ agua de grifo, 0.9

Temperatura del agua a tratar (°C), 15

Demanda bioquímica de oxígeno del afluente (Kg/ m³), 0.250

Demanda bioquímica de oxígeno del efluente (Kg/ m³), 0.125

Coefficiente α , 0.7

Caudal (m³/ día), 7570

Aparece el resultado de Carga Orgánica.

3. Introducir el dato del porcentaje de DBO a reducir, 50.
4. Si se conoce la velocidad de descomposición presionar la tecla S (mayúscula) e ingrese el valor; sino se conoce el valor presionar N (mayúscula) y aparecerá el resultado; para este caso presionar S y 0.35.

Aparece los resultados de tiempo de detención y de volumen.

5. Introducir la profundidad del estanque (m), 3.

Se despliega una pantalla con los siguientes resultados: Superficie, Aireación Mecánica, Aireación Suplementaria y las Necesidades de Energía que es el dato que sirve para conocer el número de aireadores.

6. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel

7. Presionar la tecla Escape para salir de la pantalla. Se retorna al menu de opciones de Lagunas de Estabilización (Ver Figura 4).
8. Pulsar la tecla Escape para volver al menu principal (Ver Figura 1).

SEDIMENTACION

Esta opción permite evaluar los parámetros de diseño para un sedimentador. El programa está elaborado para utilizar un número máximo de 50 pares de datos. Se recomienda ingresar como mínimo 10 pares de datos para obtener mejores resultados.

EJEMPLO DE USO DE LA OPCION SEDIMENTACION

Se desea diseñar un tanque de sedimentación para una suspensión con una concentración inicial $C_0 = 4 \text{ Kg/ m}^3$. En una probeta de laboratorio se ha estudiado la sedimentación discontinua de esta suspensión, obteniéndose los resultados de la tabla (tiempo, altura). Calcular el área y la altura necesaria para conseguir una concentración de sólido espesado $C_u = 24 \text{ Kg / m}^3$ para un caudal de $16.67 \text{ m}^3/ \text{ h}$. La densidad es de $1,200 \text{ Kg/ m}^3$.

TIEMPO (h)	ALTURA (m)
0.00	0.39
5.00	0.32
10.00	0.246
15.00	0.180
20.00	0.132
25.00	0.096
30.00	0.071
35.00	0.055
40.00	0.042
45.00	0.034
50.00	0.026
55.00	0.020
60.00	0.018
65.00	0.016

USO DEL PROGRAMA

1. Seleccionar la Opción: Sedimentación

Aparece una pantalla de advertencia, donde se indica que hay que graficar previamente en papel milimetrado los datos obtenidos en el laboratorio. Queda a opción del usuario graficar los datos con ayuda de una computadora.

Presionar la letra S (mayúscula) para continuar.

2. Introducir el número de pares de datos, 14.
3. Hacer el ingreso de los datos tiempo y altura. Un ejemplo es el que sigue:

TIEMPO (h)	ALTURA(m)
0.00	0.39
5.00	0.32
10.00	0.246
15.00	0.180
.	.
.	.

4. Ingresar el número de pares de datos que se aproximan a la linealidad al inicio y al final según su gráfico; en este caso 3 y 4.

- Se despliega una pantalla que muestra los diferentes cálculos que realiza el programa. Si los datos que se han ingresado al principio son pocos aparece una pantalla que pide que se reinicie el ingreso de datos.
5. Ingresar el dato de la altura inicial de la interfase (éste dato es la altura cuando el tiempo es 0.00 en metros), 0.39.
 6. Introducir los siguientes datos:

Concentración inicial de la solución (Kg/ m^3), 4

Concentración final de la solución (Kg/ m^3), 24

Caudal ($\text{m}^3/\text{ h}$), 16.67.

Aparece el resultado del área transversal y profundidad del sedimentador.
 7. Si desea imprimir, pulsar la letra S en mayúscula.

Antes de utilizar la opción de impresión asegurarse que la impresora este conectada y tenga papel
 8. Presionar la tecla Escape para salir de la pantalla. Se retorna al menu principal.

**ANEXOS PARA
INTERCAMBIO IONICO**

**GUIA DE DIMENSIONAMIENTO DE SISTEMAS DE INTERCAMBIO IONICO DE LA
CULLIGAN COMPANY DE LOS ESTADOS UNIDOS**

FRP RESIDENTIAL/LIGHT COMMERCIAL TANKS

TANK SELECTOR GUIDE

FRP tanks are constructed of a fiberglass reinforced polyester resin for standard water conditioning use and of fiberglass reinforced vinylester resin for DI applications. Unlike other tanks and metal vessels, FRP tanks are made with one-piece construction. That means there are no seams, welds, or joints which can corrode, electrolyze, give way, or leak. Finished tanks are about one-third the weight of comparable steel tanks, so they are easier and more economical to handle.

Features:

- Standard 2 1/2" top thread (other sizes available)
- NSF approved
- Available in natural or blue color
- Operating pressure: 150 psi
- Max. operating temp.: 120°F polyester
150°F vinylester

Applications:

This economical and durable tank is a popular choice for standard water softener applications. Vinylester tanks are ideal for use as DI exchange tanks.

CAPACITIES AND DIMENSIONS

US STANDARD											METRIC									
TANK SIZE	GAL.	CU. FT.	NOMINAL SIZE		DIMENSIONS IN INCHES						TANK SIZE	LTR.	NOMINAL SIZE		DIMENSIONS IN MILLIMETERS					
			A	C1	STD. BASE L1	EXT. BASE L2	D	H1	H2	A			C1	STD. BASE L1	EXT. BASE L2	D	H1	H2		
6 x 18	2.3	0.3	6.0	18.0	18.1	24.0	9.4	4.1	4.5	6 x 18	8.7	152	457	460	610	239	104	114		
6 x 35	4.4	0.6	6.0	35.0	35.1	41.0	26.4	4.1	4.5	6 x 35	16.6	152	889	892	1041	671	104	114		
7 x 24	3.8	0.5	7.0	24.0	24.1	30.0	15.1	4.9	4.1	7 x 24	14.4	178	610	612	762	384	124	104		
7 x 35	5.7	0.8	7.0	35.0	35.1	40.9	26.0	4.9	4.1	7 x 35	21.6	178	889	892	1039	660	124	104		
7 x 40	6.6	0.9	7.0	40.0	40.2	46.0	31.1	4.9	4.1	7 x 40	25.0	178	1016	1021	1168	790	124	104		
7 x 44	7.3	1.0	7.0	44.0	44.4	50.2	35.3	4.9	4.1	7 x 44	27.6	178	1118	1128	1275	897	124	104		
8 x 17	3.1	0.4	8.0	17.0	17.2	23.0	7.2	5.3	4.4	8 x 17	11.7	203	432	437	584	183	135	112		
8 x 24	4.6	0.6	8.0	24.0	24.3	30.1	14.3	5.3	4.4	8 x 24	17.4	203	610	617	764	363	135	112		
8 x 30	6.0	0.8	8.0	30.0	30.2	36.0	20.2	5.3	4.4	8 x 30	22.7	203	762	767	914	513	135	112		
8 x 35	7.1	0.9	8.0	35.0	35.2	41.0	25.2	5.3	4.4	8 x 35	26.9	203	889	894	1041	640	135	112		
8 x 40	8.2	1.1	8.0	40.0	40.3	46.1	30.3	5.3	4.4	8 x 40	31.0	203	1016	1024	1171	770	135	112		
8 x 44	9.1	1.2	8.0	44.0	44.5	50.2	34.4	5.3	4.4	8 x 44	34.4	203	1118	1130	1275	874	135	112		
9 x 35	8.8	1.2	9.0	35.0	35.3	41.0	24.5	5.3	5.2	9 x 35	33.3	299	889	897	1041	622	135	132		
9 x 40	10.3	1.4	9.0	40.0	40.3	46.0	29.5	5.3	5.2	9 x 40	39.0	229	1016	1024	1168	749	135	132		
9 x 48	12.5	1.7	9.0	48.0	48.3	54.0	37.5	5.3	5.2	9 x 48	47.3	229	1219	1227	1372	952	135	132		
10 x 19	5.1	0.7	10.0	19.0	19.3	24.8	8.0	5.8	5.2	10 x 19	19.3	254	483	490	630	203	147	132		
10 x 30	8.8	1.2	10.0	30.0	30.4	36.0	19.3	5.8	5.2	10 x 30	33.3	254	762	772	914	490	147	132		
10 x 35	10.4	1.4	10.0	35.0	35.2	40.7	23.8	5.9	5.2	10 x 35	30.4	254	889	894	1034	605	150	132		
10 x 40	12.1	1.6	10.0	40.0	40.3	45.8	28.9	5.9	5.2	10 x 40	45.9	254	1016	1024	1163	734	150	132		
10 x 42	12.7	1.7	10.0	42.0	42.3	47.8	31.0	5.8	5.2	10 x 42	48.1	254	1067	1074	1214	787	147	132		
10 x 44	13.4	1.8	10.0	44.0	44.3	49.9	33.1	5.8	5.2	10 x 44	50.7	254	1118	1125	1267	841	147	132		
10 x 47	14.4	1.9	10.0	47.0	47.3	52.8	36.0	5.8	5.2	10 x 47	54.5	254	1194	1201	1341	914	147	132		
10 x 54	16.7	2.2	10.0	54.0	54.3	59.8	43.0	5.8	5.2	10 x 54	63.2	254	1372	1379	1519	1092	147	132		
12 x 48	23.4	3.1	12.0	48.0	48.9	53.8	35.0	6.6	6.4	12 x 48	88.6	305	1219	1242	1366	889	168	162		
13 x 54	27.6	3.7	13.0	54.0	54.9	59.8	35.6	9.1	9.3	13 x 54	104.5	330	1372	1394	1519	904	231	236		
14 x 65	39.7	5.3	14.0	65.0	65.7	70.5	43.0	10.8	11.1	14 x 65	150.3	356	1651	1669	1791	1092	274	282		
16 x 65	48.3	6.4	16.0	65.0	65.9	71.1	41.7	11.5	11.9	16 x 65	182.8	406	1651	1674	1806	1052	292	302		



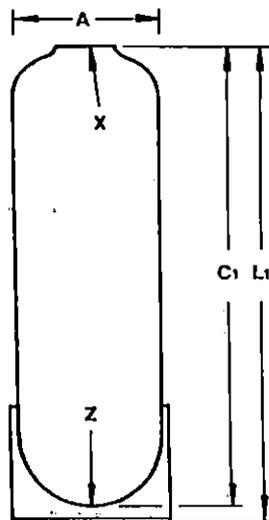
MATT-SON, INC. • 28W005 Industrial Ave. • Barrington, IL 60010 USA
 TEL 708 MATTSON (628 8766) FAX 708 382 5814 (Illinois and International)
 TEL 708 382 7810 FAX 800 833 5593 (US and Canada Tollfree)

TERMS AND CONDITIONS: MINIMUM ORDER \$100.00 • ALL PRICES F.O.B. SHIPPING POINT
 NET 30 DAYS ON APPROVED ACCOUNTS • PRICES SUBJECT TO CHANGE WITHOUT NOTICE

CAPACITIES AND DIMENSIONS

US STANDARD							METRIC					
TANK SIZE	US GAL.	CU. FT.	NOMINAL DIMENSIONS IN INCHES				NOMINAL DIMENSIONS IN CENTIMETERS					
			A	C ₁	L ₁	X	TANK SIZE	LITER	A	C ₁	L ₁	X
6 x 13	1.1	0.15	6.3	12.6	12.7	2.5	6 x 13	4.2	16.0	32.0	32.3	6.35
6 x 18	1.9	0.25	6.3	18.0	18.1	2.5	6 x 18	7.2	16.0	45.7	46.0	6.35
6 x 35	3.7	0.51	6.3	34.8	35.0	2.5	6 x 35	14.0	16.0	88.4	88.9	6.35
7 x 35	5.0	0.67	7.3	34.8	35.0	2.5	7 x 35	18.9	18.5	88.4	88.9	6.35
7 x 44	7.3	1.0	7.3	44.0	44.2	2.5	7 x 44	27.6	18.5	111.8	112.3	6.35
8 x 35	6.5	0.86	8.3	34.8	35.0	2.5	8 x 35	24.6	21.1	88.4	88.9	6.35
8 x 40	7.5	1.01	8.3	40.0	40.2	2.5	8 x 40	28.4	21.1	101.6	102.1	6.35
8 x 44	8.4	1.13	8.3	44.2	44.4	2.5	8 x 44	31.8	21.1	112.8	112.8	6.35
9 x 12	2.4	0.32	9.3	12.1	12.6	2.5	9 x 12	9.09	23.6	30.7	32.0	6.35
9 x 18	3.9	0.52	9.3	18.0	18.2	2.5	9 x 18	14.7	23.6	45.7	46.2	6.35
9 x 35	8.2	1.09	9.3	34.8	35.0	2.5	9 x 35	31.0	23.6	88.4	88.9	6.35
9 x 40	9.9	1.33	9.3	40.0	40.3	2.5	9 x 40	37.5	23.6	101.6	102.4	6.35
9 x 48	11.7	1.56	9.3	48.0	48.3	2.5	9 x 48	44.3	23.6	121.9	122.7	6.35
10 x 35	9.9	1.33	10.1	34.8	35.0	2.5	10 x 35	37.5	25.7	88.4	88.9	6.35
10 x 40	11.4	1.53	10.1	39.8	40.0	2.5	10 x 40	43.1	25.7	101.1	101.6	6.35
10 x 44	13.4	1.8	10.1	43.8	44.0	2.5	10 x 44	50.7	25.7	111.3	111.8	6.35
10 x 54	16.0	2.14	10.1	54.3	54.5	2.5	10 x 54	60.6	25.7	137.9	138.4	6.35
12 x 48	17.2	2.3	12.0	48.0	48.2	2.5	12 x 48	65.0	30.5	121.9	122.4	6.35
12 x 52	21.8	2.9	12.0	52.0	52.2	2.5	12 x 52	82.4	30.5	132.1	132.6	6.35
13 x 54	27.5	3.7	13.0	54.0	54.3	2.5*	13 x 54	104.0	33.0	137.2	137.9	6.35*
14 x 47	26.4	3.53	14.3	47.4	47.8	4.5	14 x 47	99.9	36.3	120.4	121.4	11.43
14 x 65	41.0	5.5	14.3	65.0	65.4	2.5*	14 x 65	155.0	36.3	165.1	166.1	6.35*
16 x 28	19.8	2.64	16.0	29.0	30.3	2.5*	16 x 28	74.9	40.6	73.6	76.9	6.35*
16 x 40	29.5	3.93	16.0	41.2	42.3	2.5*	16 x 40	111.6	40.6	104.6	107.4	6.35*
16 x 53	40.3	5.37	16.0	54.0	55.2	2.5*	16 x 53	152.5	40.6	137.1	140.2	6.35*
16 x 65	49.0	6.5	16.0	64.9	66.2	4.0	16 x 65	185.5	40.6	164.8	168.1	10.16
18 x 65	62.0	8.29	18.7	66.4	67	4.0	18 x 65	234.7	47.5	168.6	170.2	10.16
**21 x 36	47.1	6.28	21.0	37.7	41.2	2.5	**21 x 36	178.2	53.3	95.7	104.6	6.35
**24 x 38	61.2	8.16	24.0	39.7	50.0	2.5*	**24 x 38	231.6	60.9	100.8	126.9	6.35*
**25 x 50	86.4	11.52	24.0	51.0	55.6	2.5	**24 x 50	327.0	60.9	129.5	141.2	6.35

* Also available in 4.0 inch (10.16 cm)
** See Note 3 listed below.



Notes:

1. Optional "Z" bottom hole available only in 3/4" NPT on limited sizes.
2. Larger Polyglass pressure vessels available up to 48" diameter with ASME code stamp optional.
3. Tripod base comes standard on 21" - 48" diameter tanks ordered with a bottom opening: The 24 x 38 will only come with a tripod base if dual inlets are ordered. The 24 x 38 single 4" top will have a standard base.

Warranty:

A limited warranty applies to each Polyglass pressure vessel. Vessels used according to design for water conditioning applications from 6" through 16" diameters are warranted to be free from defects for 5 years. Warranty specifically does not cover fractures caused by external impact or exposure to vacuum.



MATT-SON, INC. • 28W005 Industrial Ave. • Barrington, IL 60010 USA
TEL 708 MATTSON (628 8766) FAX 708 382 5814 (Illinois and International)
TEL 708 382 7810 FAX 800 833 5593 (US and Canada Tollfree)

TERMS AND CONDITIONS: MINIMUM ORDER \$100.00 • ALL PRICES F.O.B. SHIPPING POINT
NET 30 DAYS ON APPROVED ACCOUNTS • PRICES SUBJECT TO CHANGE WITHOUT NOTICE

CAPACITIES AND DIMENSIONS

US STANDARD (INCHES)

TANK SIZE	TANK		OVERALL HEIGHT		TOP OPENING		BOTTOM OPENING (OPTIONAL)		TOTAL VOLUME		DOME VOLUME		SHIP WT. LBS.
	DIA.	HT.	STD. BASE L1	EXT. BASE L2	FLANGE 6"	FLANGE 6"	GRAVEL LBS.	RESIN CU.FT.	CU. FT.	GAL.	CU. FT.	GAL.	
24 x 72	24	75.0	81.75	81.75	X	X	175	10	15.9	119	1.3	9.7	195
30 x 72	30	75.5	83.5	86.75	X	X	250	15	23.8	185	2.5	18.7	272
36 x 72	36	76.5	83.5	88.0	X	X	350	22	33.1	247	4.2	31.4	335
42 x 72	42	73.0	N/A	84.5	X	Std.	500	30	45.4	340	6.4	47.9	660
48 x 72	48	77.75	N/A	89.0	X	Std.	600	40	61.9	463	9.9	74.1	760

METRIC (CM)

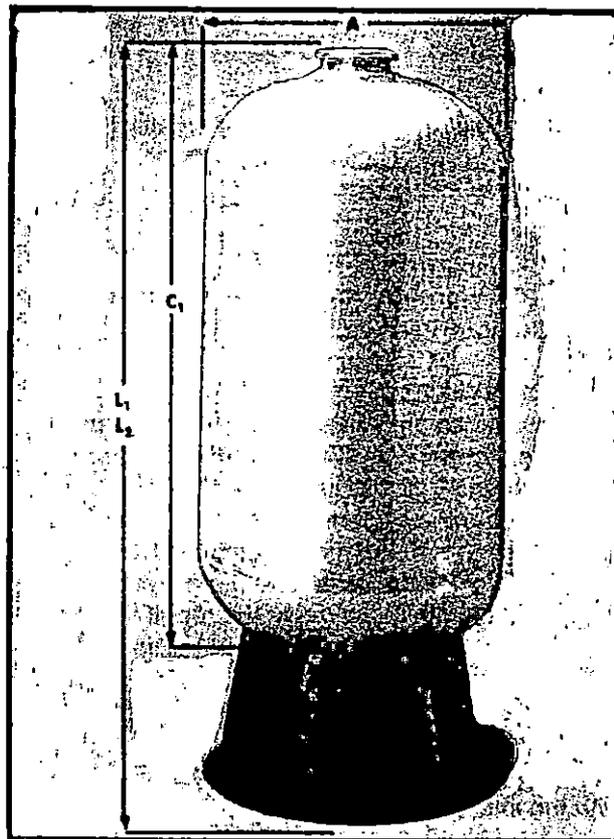
TANK SIZE	TANK		OVERALL HEIGHT		TOP OPENING		BOTTOM OPENING (OPTIONAL)		TOTAL VOLUME		DOME VOLUME		SHIP WT. KGS.
	DIA.	HT.	STD. BASE L1	EXT. BASE L2	FLANGE 6"	FLANGE 6"	GRAVEL KGS.	RESIN M ³	LITERS	LITERS	LITERS		
24 x 72	61	183	207.6	207.6	X	X	79.4	283.0	450.0	36.8	88.5		
30 x 72	76	183	212	220.7	X	X	113.4	424.5	673.5	70.8	123.4		
36 x 72	91	183	212	223.5	X	X	158.8	622.6	936.7	118.9	152.0		
42 x 72	107	183	N/A	214.5	X	Std.	226.8	849.0	1284.8	181.1	299.4		
48 x 72	122	183	N/A	226.0	X	Std.	272.2	1132.0	1751.8	280.2	344.7		

Notes:

- 150 psi operating pressure
120°F operating temperature on threaded tanks
150°F operating temperature on flanged tanks
- Single 6" top flange standard on 24", 30", and 36" diameter vessels. 42" and 48" require top and bottom openings. 6" top and bottom flange standard. Optional 6" bottom flange available on 24", 30", and 36" vessels.
- ASME Section X code stamp optional for 24" to 48" diameter vessels.
- Smaller Polyglass pressure vessels available down to 6" diameter. Call Matt-Son for information.

Warranty:

A limited warranty applies to each Polyglass pressure vessel. Those vessels used for commercial or industrial water softeners, filters or deionizers, and operating within the specified physical properties of the vessel, will be warranted for five years from the date of manufacture. Any vessel deemed by the manufacturer to have failed within the warranty period will be replaced without charge. Application other than those described herein must be approved by the manufacturer prior to installation to provide warranty coverage. Warranty specifically does not cover sandblasting of tank caused by faulty distribution system, damage caused by exposure to vacuum, external attack by corrosive liquids or gases, and damage as a result of external impacts.



MATT-SON, INC. • 28W005 Industrial Ave. • Barrington, IL 60010 USA
 TEL 708 MATTSON (628 8766) FAX 708 382 5814 (Illinois and International)
 TEL 708 382 7810 FAX 800 833 5593 (US and Canada Tollfree)

TERMS AND CONDITIONS: MINIMUM ORDER \$100.00 • ALL PRICES F.O.B. SHIPPING POINT
 NET 30 DAYS ON APPROVED ACCOUNTS • PRICES SUBJECT TO CHANGE WITHOUT NOTICE

CONCLUSIONES Y RECOMENDACIONES

1. Al proyectar unidades de sedimentación es necesario realizar ensayos de laboratorio para obtener los datos experimentales que se transforman en parámetros de diseño, tales como el área de la sección transversal y profundidad del sedimentador.

El procedimiento de cálculo tradicional requiere de graficar los datos experimentales de tiempo en el eje de las abscisas y la altura de la interfase en el eje de las ordenadas. Una vez construída la gráfica se trazan líneas tangentes al principio y final de la curva y se localiza el ángulo de bisección.

Luego, con los resultados de tiempo y altura en la zona de compresión y con los balances de masa se estiman el área y la profundidad del sedimentador.

El presente documento cuenta con un informe de los procedimientos matemáticos involucrados en el diseño de unidades de sedimentación (Cap. I); así como de los diferentes modelos matemáticos y métodos numéricos utilizados para la resolución del problema.

Finalmente, el programa elaborado permite de una manera fácil, obtener el área transversal y profundidad del sedimentador a partir de datos experimentales.

2. El uso de lagunas de estabilización para el tratamiento de aguas residuales está limitado por las características del agua en cuestión y puede ser utilizado solamente para aquella agua en la que se ha comprobado que la vía biológica es adecuada para su descontaminación en cierto grado. El uso de lagunas de estabilización, ya sean del tipo aerobias, anaerobias o facultativas dependerá del tipo de agua a tratar, por lo que es necesario realizar ensayos biológicos de degradación de los contaminantes del agua residual para garantizar la eficiencia del tratamiento por lagunaje.

Otros factores a considerar son la disponibilidad de terreno a bajo costo y con características favorables, tales como: no estar muy cerca de las comunidades, topografía y geografía del lugar, así como también de los factores meteorológicos de la región.

El programa para computadora de lagunas de estabilización ha sido preparado en base a modelos matemáticos que consideran la cinética de degradación, condiciones topográficas, meteorológicas y la población a servir permitiendo evaluar los parámetros de diseño del sistema para un proyecto específico, tales como profundidad, superficie, volumen y tiempo de detención.

Se recomienda el uso de lagunas cuadradas o rectangulares para aquellas de tipo primario debido a que en ellas ocurre una acumulación grande de sólidos. Para lagunas secundarias es recomendable el uso de lagunas de forma alargada.

3. La adsorción en carbón activo es un sistema de tratamiento que puede ser utilizado para la remoción de contaminantes de tipo orgánico en agua residual tratada previamente por otro sistema. Por otra parte, puede ser utilizado en procesos industriales que requieran la eliminación de sustancias coloreadas, generadoras de sabor desagradable en aguas, eliminación de componentes orgánicos en agua para bebida o para agua de uso alimenticio o farmacéutico.

El diseño de la unidad de adsorción en carbón activado se basa en la isoterma de adsorción de Freundlich y Langmuir, siendo necesario la obtención de las constantes empíricas de dichos modelos a partir de datos experimentales proporcionados por los ensayos de adsorción con el agua a tratar.

El programa para adsorción en carbón activado se fundamenta en los modelos matemáticos de Freundlich y Langmuir y facilita la evaluación de las constantes de cada una de las isotermas.

4. Para diseñar un adecuado sistema de tratamiento de agua utilizando resina de intercambio iónico, debe contarse con un análisis fisicoquímico del agua, el cual permitirá evaluar las concentraciones de las especies presentes y determinar si es adecuado el uso de dicha tecnología. Además, permite realizar una adecuada selección del tipo de resina para el sistema.

El programa elaborado sobre intercambio iónico permite estudiar los volúmenes de resina aniónica y catiónica requeridos para la desmineralización de un agua de calidad específica; esta basado en los parámetros de diseño propuestos en la guía de dimensionamiento de sistemas intercambio iónico de la Culligan Company de los Estados Unidos.

Para ello se procedió a elaborar modelos de regresión de las diversas curvas características y establecer los planteamientos lógicos y matemáticos para la resolución de la problemática.

Como resultado, el programa solicita mediante un fácil sistema de entrada de datos, las características fisicoquímicos del agua a tratar, el volumen ó caudal requerido, además de otros factores y finalmente, se evalúan la cantidad de resina catiónica y aniónica necesarias para el tratamiento.

5. Para el estudio de los programas de tratamiento de aguas expuestos en éste trabajo se sugiere profundizar en el estudio de métodos numéricos e inferencia estadística, específicamente en las áreas de regresión y correlación de datos.

Los programas de computadora formulados en lenguaje C para el diseño de unidades de tratamiento expuestos en el presente trabajo de graduación, utilizan el sistema internacional (SI) para las variables involucradas y permiten evaluar con facilidad mediante el ingreso de datos los parámetros de diseño.

REFERENCIAS BIBLIOGRAFICAS

1. Almanaque, Salvadoreño, Ministerio de Agricultura y Ganadería , Centro de Metereología e Hidrología, República de el Salvador, C.A, 1993.
2. Austin, George T. "Manual de Procesos Químicos en la Industria", Primera Edición en Español, Mc. Graw-Hill, 1994.
3. Aitken, Peter-Jones Bradley, "Aprendiendo C en 21 días " , edición en español, Pretice Hall Hispanoamericana, S.A.,,impreso en Mexico, 1994.
4. Babbit, Harold E.-Donald James J., "Water Supply Engineering " , sixth edition, Mc-Graw-Hill Book Company, Inc. ,United States of America, 1962.
5. Bonilla, Gildaberto, "Estadística II Métodos Prácticos de Inferencia Estadística", Tercera Edición, UCA Editores, El Salvador, 1995.
6. Castillo del Valle, José Ernesto, "Aplicación de las Computadoras al Diseño de Columnas de Platos", Trabajo de Graduación para optar al título de Ingeniero Químico, Universidad de El Salvador, Facultad de Ingeniería y Arquitectura, noviembre, 1985.
7. CEPIS, OPS, OMS, "Manual de Disposición de Aguas Residuales", Tomo II, Lima, Perú, 1991.
8. CEPIS, "Reuso en Acuicultura de las Aguas Residuales Tratadas en Lagunas de Estabilización de San Juan", Tratamiento de las Aguas Residuales y Aspectos Sanitarios, Sección II, OMS, OPS, Lima, Perú, 1992.
9. CEPIS, "Seminario Regional de Investigación sobre Lagunas de Estabilización", Programa de Salud Ambiental, OMS, OPS, Lima, Perú, 1986.
10. Chapra Steven C.-Canale Raymond P. "Métodos Numéricos para Ingenieros con Aplicaciones en Computadoras Personales ", Primera edición, impreso en Mexico, 1988.

11. Cox, Charles R., "Práctica y Vigilancia de las Operaciones de Tratamiento del Agua", Organización Mundial de la Salud, Ginebra, 1966.
12. Devore, Jay, Peck Roxy, "Statistics", The Exploration and Analysis of Data, West Publishing Company, MN, USA, 1986.
13. Fair, Gordon Maskew-Geyer John Charles, "Purificación de Aguas y Tratamiento y Remoción de Aguas Residuales", Volumen 2, Octava Reimpresión, Editorial Limusa, Grupo Noriega Editores, México D.F, 1993.
14. Faust, Samuel D-Aly Osman M., "Adsorption Processes for Water Treatment", Butterwoth Publishers, United States of America, 1987.
15. FIA, UES, OPS-MSP Curso de "Lagunas de Estabilización", San Salvador, 1968
16. Foust, Alan S. y otros, "Principios de Operaciones Unitarias", Tercera impresión, Compañía Editorial Continental, S.A de C.V, México, 1990.
17. Freund John E. "Modern Elementary Statistics", Seventh Edition Prentice-Hall, United States of America, 1988.
18. Fuentes T. , Alicia et al. "Diseño y Construcción de un Sistema de Desmineralización de Agua para uso del Laboratorio de Ingeniería Química", Trabajo de Graduación para optar al Título de Ingeniero Químico, Facultad de Ingeniería y Arquitectura, Universidad de El Salvador, 1994.
19. Gloyna, Earnest F., "Estanques de Estabilización de Aguas Residuales", Edición en inglés, publicada en 1971, Organización Mundial de la Salud, Impreso en Suiza, 1973.
20. Kernigham, Brian W.- Ritchie Dennis M., "El Lenguaje de Programación C", quinta reimpresión, Programas Educativos, S.A de C.V., Prentice-Hall Hispanoamericana, S.A., Mexico, 1988.

32. Sanchez Hernández, Salvador Ernesto- Merino Garay, Jairo Orlando, "Lagunas de Estabilización en El Salvador", Tesis para optar al grado de Ingeniero Civil, Universidad Centroamericana José Simeón Cañas, El Salvador, 1983.
33. Swokowski-Earl W., "Matrices y Determinantes", Grupo Editorial Iberoamérica, S.A. de C.V., Mexico, 1986.
34. Tejada Guerra, Manuel- Figueroa Campos, Silvia, "Geometría Analítica", para Bachillerato, Mc Graw-Hill Interamericana de Mexico, S.A. de C.V., Mexico, 1992.
35. Treybal, Robert E., "Operaciones de Transferencia de Masa", Segunda Edición Mc Graw Hill, Impreso en México, 1988.
36. Yanez Cossío Fabian, "Lagunas de Estabilización", teoría, diseño, evaluación y mantenimiento, Imprenta Monsalve, Cuenca, Ecuador, 1993.