

ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL
FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN
FUNDAMENTOS DE PROGRAMACIÓN
SEGUNDA EVALUACIÓN
II TÉRMINO 2011-2012

Nombre: _____ **Matrícula:** _____

TEMA 1 (20 puntos)

Sea la siguiente operación, aplicable a cualquier número entero positivo:

- Si el número es par, se divide entre 2.
- Si el número es impar, se multiplica por 3 y se suma 1.

Dado cualquier número, si realizamos sucesivamente esta operación sobre los resultados, tendremos una serie que converge hacia 1. Esta es llamada la **conjetura de Collatz**. Por ejemplo, si comenzamos con el número 13, la serie a obtener sería:

13, 40, 20, 10, 5, 16, 8, 4, 2, 1

Usted debe: Escribir una función recursiva **int collatz(int n)** que, dado un número entero positivo, indique cuantos términos existen en la serie antes de llegar a 1. Para el ejemplo anterior, **collatz(13)** retorna el valor de **10**.

TEMA 2 (35 puntos)

Se desea implementar un sistema para la administración del cine de FIEC. Los asientos de la sala de cine están representados por una matriz de M filas y N columnas. La posición de cada asiento se definirá con una letra para la fila y un número para la columna. En la matriz los asientos libres están representados por el 0 y los ocupados por el 1.

	1	2	3	4	5	6	7	8	9	10	..
A					*	*					
B							*	*	*	*	
C	*	*	*	*	*	*	*	*	*	*	
D	*	*	*	*	*	*	*	*	*	*	*
E	*	*	*	*	*	*	*	*	*	*	*
F	*	*	*	*	*	*	*	*	*	*	*
G	*	*	*	*	*	*	*	*	*	*	*
H	*	*	*	*	*	*	*	*	*	*	*
.											
.											

A Ud. Se le ha solicitado implementar:


- a) El procedimiento **visualizar** (int cine[M][N]), que permita visualizar como en el siguiente gráfico qué sitios están disponibles para que el cliente pueda decidir dónde sentarse. Los asteriscos indican los asientos libres.
- b) La función int **reserva**(int cine[M][N], char fila, int columna), la cual deberá comprobar que la posición del asiento que recibe está libre, devolver 1 y actualizar la matriz, caso contrario devolverá -1.
- c) La función bool **película**(int cine [M][N]), la cual devuelve true si la función tiene suficientes asientos reservados para proyectar la película, de lo contrario devuelve false. Sólo se podría proyectar una función de la película si está lleno como mínimo el 20% del auditorio.

TEMA 3 (45 puntos)

Un crucigrama es un pasatiempo escrito que consiste en escribir en una plantilla una serie de palabras en orden vertical y horizontal que se cruzan entre sí. Para el desarrollo del juego, en la plantilla se proponen listas de definiciones, horizontal y vertical. La plantilla está dividida en casillas blancas que corresponden a letras individuales y negras que sirven para separar palabras.

Una empresa está implementando la versión electrónica del juego. El sistema debe mostrar un menú con las siguientes opciones:

1. Nuevo Juego
 1. Ingresar Palabra
 2. Cancelar Juego
2. Salir

	crucigrama.dat	definiciones.dat
	<pre>1,2,h,false 1,2,v,fila 1,6,v,else 2,8,v,ctype 3,6,h,setup 3,10,v,punt 4,1,h,math 6,1,h,h 6,3,h,fuente</pre>	<pre>1,h,valor booleano de negación 1,v, Conjunto de celdas contenidas de forma horizontal 2,v, estructura de control que modifica el flujo de ejecución 3,v,librería de manejo de caracteres 4,h,proyecto de instalación en Visual Studio 5,v,4 letras de variable que referencia región de memoria 6,h,librería de operaciones matemáticas básicas 7,h,extensión de librería que guarda la interfaz 8,h,código de instrucciones que sigue la computadora</pre>

A partir de los archivos `crucigrama.dat` y `definiciones.dat` implemente:

a) El procedimiento **mostrarCrucigrama**(FILE *df, char C[M][N]), que recibe un puntero al archivo "definiciones.dat" y muestra el contenido de la matriz C de MxN que representa el crucigrama y bajo la matriz las definiciones de las palabras. El archivo contiene el número de la palabra, su posición horizontal o vertical y su definición.

b) La función **int colocarPalabra**(char C[M][N], char *palabra, int f, int c, char o, FILE *cf), que recibe la matriz C del crucigrama, la palabra con la fila, columna y orientación donde desea colocarla, y un puntero al archivo "crucigrama.dat" para comprobar su validez. El archivo contiene la fila, columna y orientación de la respuesta. La función retorna 1 si la palabra en esa posición es correcta y la coloca de lo contrario retorna 0.

c) El programa principal, que muestre el menú del sistema. En cada paso del juego el programa debe de mostrar el crucigrama y las definiciones abajo del mismo. El juego termina cuando el usuario llena el crucigrama correctamente con todas las palabras o escoge la opción Cancelar Juego del menú interno. Luego de esto el programa deberá regresar al menú principal. Use los procedimientos y funciones implementados en los literales anteriores.

Nota: Asuma que tiene disponible la siguiente función y procedimiento:

- **void inicializarCrucigrama** (FILE *cf, char C[M][N]), que recibe un puntero al archivo "crucigrama.dat" y la matriz C vacía. El procedimiento llena con '0' los espacios negros y con '1' los espacios de las palabras.
- **int contarPalabras**(FILE *cf) que recibe un puntero al archivo "crucigrama.dat" y retorna la cantidad de palabras que se deben llenar en el crucigrama.