División de Ingeniería
Departamento de Tecnologías de Información y Electrónica
Estructuras de Datos (TC1018)
Profesor: Dr. Vicente Cubells Nonell

Tecnológico de Monterrey

Tarea 1. Recursividad

Orientaciones sobre la entrega de la tarea

La respuesta a cada ejercicio es un programa funcionando que modele el problema planteado. Como respuesta de la tarea debe entregarse el código fuente de cada ejercicio por separado (cada ejercicio es un proyecto diferente). Para la evaluación, el profesor compilará el proyecto de cada ejercicio, el cual debe funcionar sin errores y analizará el código fuente.

La respuesta de la tarea debe subirse a GitHub y en BB debe subirse la liga de acceso al repositorio. La tarea estará activa en BB hasta el 29 de enero de 2015 a las 23:55 horas.

No se aceptan trabajos fuera de fecha ni por correo, en ambos casos la calificación de la tarea será 0 puntos.

Ejercicio 1.

Programe un algoritmo recursivo que permita resolver una matriz de N x N con la siguiente forma (ejemplo para una matriz de 5 x 5). EL algoritmo debe funcionar para cualquier valor de N.

1 1 1 1 1
1 2 2 2 2
1 2 4 4 4
1 2 4 8 8
1 2 4 8 16

Ejercicio 2.

Realice un programa que resuelva el problema de un laberinto: Se trata de encontrar un camino que nos permita salir de un laberinto definido en una matriz N x N. Para movernos por el laberinto, sólo podemos pasar de una casilla a otra que sea adyacente a la primera y no esté marcada como una casilla prohibida (esto es, las casillas prohibidas determinan las paredes que forman el laberinto y pueden marcarse con un número, por ejemplo -1).

Algoritmo:

Se comienza en la casilla (0,0) y se termina en la casilla (n-1,n-1)

Nos movemos a una celda adyacente si esto es posible.

Cuando llegamos a una situación en la que no podemos realizar ningún movimiento que nos lleve a una celda que no hayamos visitado ya, retrocedemos sobre nuestros pasos y buscamos un camino alternativo.

Ejercicio 3.

Si se tiene el siguiente problema:

Se tiene un tablero de n × m y un robot parado en la casilla superior izquierda (posición 0, 0).

El robot sólo puede moverse hacia la derecha o hacia abajo.

Si se mueve a la derecha avanza 3 casilleros.

Si se mueve hacia abajo avanza 2 casilleros.

Realice un programa que utilizando recursividad permita determinar cuántos caminos distintos posibles puede seguir el robot para llegar a la casilla inferior derecha (posición n-1, m-1) del tablero.