Tarea 02: Pilas y colas

Profesor: Alejandro Hernández Mora *
Ayudante: Pablo Camacho González †
Ayudante Lab: Luis Manuel Martínez Dámaso ‡

Lunes 13 de Abril de 2020

1 Motivación

Una imprenta se dedica a hacer libros con actividades para niños, entre éstas se encuentran libros con actividades lúdicas, que ayudan a su desarrollo. La imprenta desea publicar laberintos en sus libros, así que te contrata a ti, que sabes estructuras de datos y un poco de algoritmos para que hagas un programa que automáticamente genere laberintos.

2 Algoritmos

Creación del laberinto: El laberinto será modelado con un arreglo bidimensional, en este arreglo estarán las casillas del laberinto, cada una tendrá la posibilidad de poder abrirse camino en 4 direcciones; arriba, abajo, izquierda y derecha; si el paso está abierto para alguna, entonces quitamos la línea(pared) que divide a ambas casillas.

```
1 P = new Pila();
2 Elegir una casilla aleatoriamente. Marcarla como visitada y agregarla a la pila.
3 while !pila.es Vacia() do
4 Elegir aleatoriamente una casilla adyacente no visitada, marcarla como visitada, borrar la pared correspondiente, se agrega a la pila. if noHay Vecinos Sin Visitar() then
5 | P.pop();
6 | end
7 end
```

Solución al laberinto: Una vez creado el laberinto, se hace un algoritmo prácticamente igual para encontrarlo. Deseamos encontrar el camino más corto del inicio al fin, así que en lugar de utilizar una pila, utilizaremos una cola.

Para las casillas de inicio y fin hay varias opciones, pedirselas al usuario, mantenerlas aleatorias o fijar un inicio y un fin para todos los laberintos (Por ejemplo dos esquinas).

^{*}alejandrohmora@ciencias.unam.mx

[†]pablopcg1@ciencias.unam.mx

 $^{^{\}ddagger}$ luismanuelyes@ciencias.unam.mx

3 Requerimientos

- Deberás escribir un programa en Java que te permita crear un laberinto dados dos parámetros m y
 n, que son el largo y ancho del laberinto (casillas de una matríz).
- El programa recibe los parámetros conforme a la entrada estándar de linux (parámetros dados en la terminal).
- La salida deberá ser el laberinto dibujado en la terminal.y deberás mostrar la trayectoria del inicio al fin mediante un camino con el caracter 'x'.
- Únicamente podrás utilizar las estructuras de datos que programamos en el laboratorio y arreglos.
- Es importante que tu programa garantice que cualquier laberinto tiene solución. Es decir que para cualquier par de casillas elegidas arbitrariamente, exista un camino entre éstas.

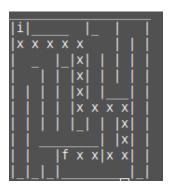


Figure 1: Una salida similar a esta es suficiente para cumplir con el programa.

4 Punto extra

En esta ocasión el punto extra no será únicamente uno, serán tres puntos extra. Con la finalidad de que recuperen puntos si no entregaron la primer tarea. No se acreditan menos de los tres puntos, es decir se ganan todos, o no se gana ninguno, incluso aunque estén parcialmente hechos.

- Deberás crear una interfaz gráfica donde se manipulará por completo el programa.
- En lugar de dibujar en la terminal el laberinto tu programa deberá crear y guardar tantas imagenes como laberintos se generen y se deseen guardar, durante la ejecución del programa. Las imágenes deben guardarse en una localidad donde el usuario elija. Deberás marcar con alguna figura el inicio y con alguna otra el final, el camino estará representado por casillas de color rojo, que conecten únicamente al inicio con el final.

5 Entrega

La entrega será conforme a los lineamientos de entrega de las prácticas, únicamente a través de la plataforma de trabajo *Google Classroom* correspondiente al curso.

Deberá ser entregada antes del día martes 28 de abril de 2018 a las 23:59, por cada día de retraso se penalizará con un punto menos, con un retraso mayor a tres días, la tarea no será calificada.