

INTRODUCCIÓN A LA PROGRAMACIÓN ORIENTADA A OBJETOS

750081M

EISC - Universidad del Valle

Ángel E. García Baños

2018-10-05

TALLER 3: Matrices, bucles y condicionales

Objetivo: Diseñar un algoritmo para fabricar laberintos. Este algoritmo se podrá usar para el proyecto, aunque habrá que hacer algunas modificaciones en los tipos de datos.

Se proporcionan los siguientes archivos:

Laberinto.h

Laberinto.cpp

Test/Test_Laberinto.cpp aquí está el main()

Usted debe escribir la función que falta en el archivo Laberinto.cpp, para fabricarlo.

En el constructor de Laberinto ya hay una parte realizada:

- Rellenar todas las casillas con pared **[1]**.
- Poner la puerta de entrada **[5]** en la columna que está más a la izquierda (con la fila elegida al azar) y la salida **[6]** en la columna que está más a la derecha (con la fila elegida al azar).

Se sugiere fabricar el laberinto implementando estos pasos:

- Trazar una línea recta entre la entrada y la salida, vaciando las casillas **[0]**.
- Trazar unas cuantas líneas rectas más para despistar, entre dos puntos elegidos al azar. Conviene que escriba una función **protected** para este punto y el anterior, que reciba la fila y columna de dos puntos, y que dibuje una recta (poniendo ceros **[0]**) entre ellos.
- Situar los tesoros **[2]**, gnomos **[3]** y dragones **[4]** en casillas elegidas al azar que previamente no tuvieran nada **[0]**.

Usted debe correr algunas pruebas que están en Test/Test_Laberinto.cpp. Para ello, ejecute estos 3 comandos:

```
cd Test
```

```
g++ -std=c++11 ../Laberinto.cpp Test_Laberinto.cpp -o Test_Laberinto
```

```
./Test_Laberinto
```

Metodología: PAIR PROGRAMMING en grupos de 2 personas, alternándose cada 10 minutos en el uso del computador. Si terminan con todo correcto en el tiempo de la clase, tendrán 0.1 de PREMIO en la nota final de la asignatura.