

PROYECTO PARTE 1. ELEMENTOS BASE PARA TRABAJAR CON MAPAS

En este proyecto se desarrollarán los elementos básicos para trabajar con laberintos como:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1															
2															
3															
4															
5															
6															
7															
8															
9															
10															
11															
12															
13															
14															
15															

Esta clase de mapa de laberinto puede estar en un archivo de texto plano con extensión .txt que contenga los valores del laberinto separados por coma. En este caso, los posibles valores son:

Color	Representa	Valor codificado
	Pared	0
	Camino	1

Así, el archivo de texto de este laberinto sería:

0,0,0,0,0,0,0,0,1,0,0,0,0,0,0
0,1,1,1,1,0,1,1,1,1,1,1,1,1,1
0,1,0,1,0,0,1,0,1,0,0,0,0,0,0
0,0,1,1,1,1,1,0,1,0,0,0,0,0,0
0,0,0,1,0,1,0,0,1,0,0,0,0,0,0
0,0,0,1,0,0,0,0,1,0,0,0,0,0,0
0,1,1,1,1,1,1,0,1,0,0,0,0,0,0
0,1,0,1,0,0,0,0,0,0,0,0,0,0,0
0,1,0,1,1,1,1,1,1,0,0,0,0,0,0
1,1,0,1,0,0,0,1,0,0,0,0,0,0,0
0,0,0,1,0,0,0,1,0,0,0,0,0,0,0
0,0,0,1,0,0,0,1,0,0,0,0,0,0,0
0,0,0,1,0,0,0,1,0,0,0,0,0,0,0
0,0,0,1,0,0,0,1,0,0,0,0,0,0,0

Se pueden agregar más valores para representar diferentes terrenos, por ejemplo considerando los siguientes valores:

Color	Nombre del terreno	Código del terreno en el txt
	Montaña	0
	Camino	13
	Agua	25
	Arena	23
	Bosque	46
	Lava	58

Podemos crear mapas de terreno como:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	46	46	46	13	13	13	13	13	13	13	13	13	13	46	46
2	46	46	46	46	46	46	13	13	13	13	46	23	46	46	46
3	46	46	46	25	25	46	13	46	13	13	23	23	46	13	13
4	46	46	25	25	25	25	13	46	13	13	23	23	13	13	13
5	46	46	25	25	25	25	13	46	46	13	13	23	13	13	13
6	46	46	46	25	25	13	13	13	13	13	13	23	23	23	23
7	46	46	46	0	0	0	13	13	13	13	13	23	23	23	23
8	46	46	46	13	0	46	46	13	13	23	23	23	23	25	25
9	46	46	46	13	13	46	46	0	13	23	23	23	25	25	25
10	46	46	13	13	13	13	0	0	0	23	23	25	25	25	25
11	46	46	13	13	13	13	13	13	23	23	23	25	25	25	25
12	46	46	13	13	13	13	13	13	23	23	25	25	25	25	25
13	46	46	13	58	58	13	13	23	23	23	25	25	25	25	25
14	46	46	46	58	58	58	13	23	23	23	25	25	25	25	25
15	46	46	46	58	58	58	13	23	23	23	25	25	25	25	25

Tome en cuenta las siguientes consideraciones:

1. Se pueden recibir mapas de n columnas x m filas, donde n y m tienen un máximo de 15. Note que n puede ser diferente de m pero que siempre se debe tener la misma cantidad de columnas para todas las filas.
2. Cualquier posición fuera del mapa no debe estar considerada para jugar.

1. Cargando mapas

1.1 Cargar un mapa

Desarrolle un método para cargar un archivo .txt y mostrarlo en una interfaz gráfica. Considere que el sistema debe identificar:

- (10) Cuántas filas y columnas tiene el mapa (note que al parsear debe identificar errores en esta parte). Recuerde que se recibirán mapas de $n \times m$, en formato txt y con valores separados por coma. Tome en cuenta que el número de columnas será el mismo para todas las filas, y se tomará el número de columnas que aparezca en la primer fila. Ejemplo, si la primer línea del archivo contiene 8 ids separados por comas entonces el número de columnas es 8; si alguna fila (línea) contiene más o menos columnas se deberá marcar error. También considere que el salto de línea NO siempre contiene retorno de carro.
- (10) Cuántos tipos de terreno están especificados en el txt (note que los ids son enteros positivos, incluyendo el 0, y pueden no ser consecutivos). Deberá poder identificar como errores los ids vacíos, letras, etc.
- (10) Para cada terreno, solicitar al usuario el nombre y el color (o imagen). El sistema puede considerar las posiciones fuera del mapa como un terreno extra.

1.2 Preguntar por el tipo de terreno

- (5) Agregar a la interfaz una forma de solicitar o preguntar por una coordenada específica y que el sistema muestre información de dicha coordenada, como el nombre de terreno.

Por ejemplo, en el mapa del laberinto, la coordenada (A,7) es pared y la (B,7) es camino. En el mapa de terreno, (E,7) es montaña, (K,15) es agua y (E,6) es arena.

1.3 Agregar información a una coordenada

Además del tipo de terreno, las coordenadas deberían tener información adicional, por ejemplo: estado inicial, estado final, número de visita, etc.

- (5) Agregue una forma de elegir o especificar la coordenada inicial (estado inicial) y la coordenada final (estado final)
- (10) Agregue una forma de mostrar información adicional directamente en la casilla (específicamente: la visita y si es inicial o final).

Por ejemplo, en el siguiente mapa nosotros establecemos las siguientes marcas:

- El número en cada coordenada especifica el número de visita de los estados/nodos visitados
- I es el estado inicial
- X es el estado o posición actual
- F es el estado final

	A	B	C	D	E
6				8	
7		5	6	7,9,11	10
8		4		12	
9		3		13,X	
10	I,1	2			
11				F	

2. Creando seres que viajan en el mapa

Para diseñar un ser se necesita especificar qué puede percibir, y que puede hacer, de acuerdo a sus habilidades. Por ejemplo, podemos considerar el costo de movimiento y tomar en cuenta los siguientes seres.

Color	Representa	Costo de movimiento			
		Humano	Mono	Pulpo	Pie grande
	Montaña	N/A	N/A	N/A	15
	Camino	1	2	2	4
	Agua	2	4	1	N/A
	Arena	3	3	N/A	N/A
	Bosque	4	1	3	4
	Pantano	5	5	2	5
	Nieve	5	N/A	N/A	3

- g) (15) Agregar a la interfaz una forma de crear al menos 1 ser y máximo 5 seres, que tengan su costo de movimiento para cada tipo de terreno (Note que el costo de movimiento 0 es diferente a N/A. En el caso de N/A el ser no puede moverse en ese terreno, mientras que costo 0 significa que puede moverse y no le cuesta nada. También note que los costos no son solo enteros, pero siempre son costos positivos. Debe truncar a dos decimales.)
- h) (5) Permitir elegir un ser (de los previamente creados) y posicionarlo en el estado inicial (note que debe validar si el ser elegido puede ser puesto en el estado inicial de acuerdo con su costo de movimiento y también validara si puede moverse en el estado final)
- i) (25) El ser deberá poderse mover (arriba, abajo, izquierda y derecha) con el teclado considerando los costos de movimiento. Es decir, si quiero mover el ser hacia arriba pero tiene un terreno donde el costo para dicho ser es N/A no debo poder hacerlo. Note que el ser no puede moverse fuera del mapa. Note también que cuando el ser se va moviendo en el mapa debe mostrarse la información en las coordenadas (lo desarrollado en el punto f), es decir, el número de visita y si es inicial o final. Note que un nodo puede ser visitado varias veces. También note que las visitas comienzan a contarse a partir de 1 y desde el estado inicial.
- j) (5) Cuando el ser llegue al estado final deberá lanzar un mensaje o mostrarlo de alguna forma en la interfaz.

Notas importantes:

- Evite, en la medida de lo posible, que el usuario capture cosas. Es preferible seleccionar. En caso de que se deban capturar datos es importante utilizar expresiones regulares para validar las entradas.
- Cuando exista un error es necesario mandar mensajes de alerta que especifiquen cuál es el problema
- Se hará referencia a las casillas por x,y es decir columna-fila, donde la columna son letras y la fila son números.
- Las filas deben comenzar en 1 y las columnas comienzan en A
- En la parte superior o inferior del mapa mostrar las letras de columna para una mejor referencia
- En la parte derecha o izquierda del mapa mostrar los números de fila para una mejor referencia
- Evite “*hardcodear*” cosas, por ejemplo: la ruta de archivo de mapa.

