

1. Análisis – Laberinto 40%

Julio César Echavarría Noreña 000419350

¿Cómo se está representando cada nodo del terreno a recorrer en la aplicación interactiva?

Se representa por medio de un grafo.

b. ¿Cuál es la estructura de datos que se está usando para almacenar cada nodo del terreno?

Se está utilizando un diccionario.

c. ¿Cómo se definen el punto de inicio y el punto final del recorrido?

Se definen por medio de un [SerializedField] en el que se elige el gameobject del nodo correspondiente.

d. ¿Qué ventaja tiene usar la estructura de datos anterior para este problema?

En el diccionario se pueden guardar los padres, por lo tanto, nos sirve para devolvernos al momento de saber cual es el camino que se debe tomar.

e. Escribe los pasos que componen el algoritmo BFS en la solución del problema.

El algoritmo BFS funciona, primero cogiendo el nodo raíz y agregándolo a la cola, luego mira sus hijos y se añade este al diccionario, luego añade a la cola sus hijos y repite la operación, indexando los padres de los hijos en el diccionario, esto hasta lograr el resultado deseado. En el momento que llega a este resultado, mira el diccionario y se devuelve en los nodos que le resultan favorables para lograr el objetivo (cabe agregar que los nodos visitados no se vuelven a revisar, para eso se agregan al diccionario), los agrega en una lista y termina el proceso, además cabe agregar que, para poder imprimir los resultados en orden, debe ponerlos en el sentido inverso al que los puso en la lista, así quedarán en orden.

f. ¿Para qué se usa una cola en el algoritmo BFS de la aplicación interactiva?

Para poner en la cola a los nodos que se tiene alrededor, así ir pasando entre ellos y solo dejar los favorables en la lista.

g. ¿Cómo se determina si un nodo ya fue explorado?

Se determina si un nodo ya fue explorado, si el nodo entra al diccionario.

h. ¿Cómo se determina qué nodos serán explorados?

Los nodos que están en la cola son los que serán explorados, ya que teniendo en cuenta el grafo, son los que están “al lado”.

i. ¿Qué estructura de datos se está utilizando para almacenar los nodos que componen la ruta calculada desde el punto de inicio al punto final?

Una lista.

j. ¿Qué ventaja tiene usar la estructura de datos de la pregunta anterior?

La ventaja de tenerlos en una lista es que esta permite “voltearlos” rápidamente, permitiéndonos llegar sin mucha complejidad al path deseado.

k. Al crear el path ¿Por qué es necesario usar el método Reverse()?

El Reverse() es necesario, ya que, como ya expliqué en el punto pasado, el camino quedaría al revés, por lo tanto toca voltearlo para poder usarlo.

l. Explica cómo funciona la aplicación interactiva luego de dar click en play.

Primero la aplicación crea y toma los datos de los nodos, creando así rápidamente un grafo. Luego, por medio del jugador, toma los nodos iniciales y finales y los transforma, activando así la búsqueda del path. En la búsqueda del path se empieza a ejecutar todo el proceso anteriormente explicado de BFS y muestra de forma visual el camino.

m. ¿Qué es una Corutina y cómo y para qué se está usando en la aplicación interactiva?

La corrutina es por así decirlo, una forma de poner nuestro código de forma que haga algo similar a una multitarea, muy parecido a el como funcionan los hilos. En el proyecto, esta se usa para poder iterar la búsqueda del camino más corto.