

PRACTICA 2

Nivel 1

- **Anchura:** He usado el algoritmo de profundidad que se nos da en el código y el cambiado la estructura de datos de pila a cola para que al recorrer el árbol por definición cuando llegué al destino la solución dada sea la mas optima en cuanto a espacio recorrido.
- **Coste Uniforme:** He modificado el algoritmo que nos da en el código cambiando la estructura de datos a un multimap, usando como clave el coste total creando un struct nodo coste que incluya una variable coste y un par de booleanas para bikini y zapatillas, He creado un método CosteAccion que calcula el coste de la casilla en la que está el personaje teniendo en cuenta las booleanas de cada nodo.

Nivel 2

He usado el algoritmo de coste uniforme y he modificado el método think añadiendo código para pintar el mapa según se indica en la guía de la práctica.

Para su correcto funcionamiento he añadido que si el jugador tiene un aldeano delante y tiene intención de avanzar su acción será quedarse quieto hasta que el aldeano no esté delante y así el jugador pueda avanzar. Para tratar los muros y precipicios, si se detecta uno de estos delante y su intención es ir en esa dirección se cambia la variable hayPlan a false para que calcule un nuevo plan al entrar en el if para así evitar caer por el precipicio o colisionar con el muro.

En cuanto a la optimización de su funcionamiento he implementado distintas medidas:

- Cuando el jugador esta explorando el mapa y se encuentra que su siguiente acción implica ir por el agua o por el bosque se recalcula el plan para ver si con el mapa descubierto hay alguna opción mejor.
- En cuanto a las estaciones de carga el código a la hora de explorar el mapa si en una de las casillas que esta descubriendo se encuentra la estación de carga guarda su posición en una variable de tipo estado.
 - Si pasa por encima de una estación de carga solo parará a cargar la batería si la batería del jugador es menor a 1300 y el tiempo restante mayor a 1000 (números elegido ejecutando el código y viendo que funcionaba bien en la mayoría de mapas) y la cargará hasta 1300.
 - Si la batería del jugador esta por debajo de 450 y el tiempo restante es mayor a 1000 en cuanto alcance su objetivo o tenga que recalcular la ruta el jugador irá a la estación de carga y cargará hasta 1300.
- A la hora de explorar el mapa si en una de las casillas que esta descubriendo se encuentra el bikini o las zapatillas el jugador dará prioridad a estas antes que al objetivo e irá a obtenerlas.
- He modificado el código del coste uniforme añadiendo un bucle while que se encarga de eliminar los elementos del mapa que ya se encuentran en generados para así agilizar

el funcionamiento del algoritmo a la hora de recorrer los elementos del mapa y evitando problemas de asignación de memoria que surgían en los mapas grandes si no se aplica esta mejora.