

INTELIGENCIA ARTIFICIAL

CURSO 2021-22

PRACTICA 2: Repertorio de preguntas para la autoevaluación de la práctica 2.

APELLIDOS Y NOMBRE	Torres Ramos, Juan Luis		
GRUPO TEORÍA	C	GRUPO PRÁCTICAS	2C

Instrucciones iniciales

En este formulario se proponen preguntas que tienen que ver con ejecuciones concretas del software desarrollado por los estudiantes. También aparecen preguntas que requieren breves explicaciones relativas a como el estudiante ha hecho algunas partes de esa implementación y que cosas ha tenido en cuenta.

En las preguntas relativas al funcionamiento del software del alumno, estas se expresan haciendo uso de la versión de invocación en línea de comandos cuya sintaxis se puede consultar en el guion de la práctica.

El estudiante debe poner en los recuadros la información que se solicita.

En los casos que se solicita una captura de pantalla (**ScreenShot**), extraer la imagen de la ejecución concreta pedida donde aparezca la línea de puntos que marca el camino (justo en el instante en el que se construye obtiene el plan). Además, en dicha captura debe aparecer al menos el nombre del alumno. Ejemplos de imágenes se pueden encontrar en [Imagen1](#) y en [Imagen2](#).

Enumera los niveles presentados en su práctica (Nivel 0, Nivel 1, Nivel 2, Nivel 3, Nivel 4):

Nivel 0, Nivel 1, Nivel 2, Nivel 3, Nivel 4.
--

Nivel 0-Demo

- (a) Rellena los datos de la tabla con el resultado de aplicar
`./practica2 mapas/mapa30.map 1 0 18 13 3 13 26`

ScreenShot

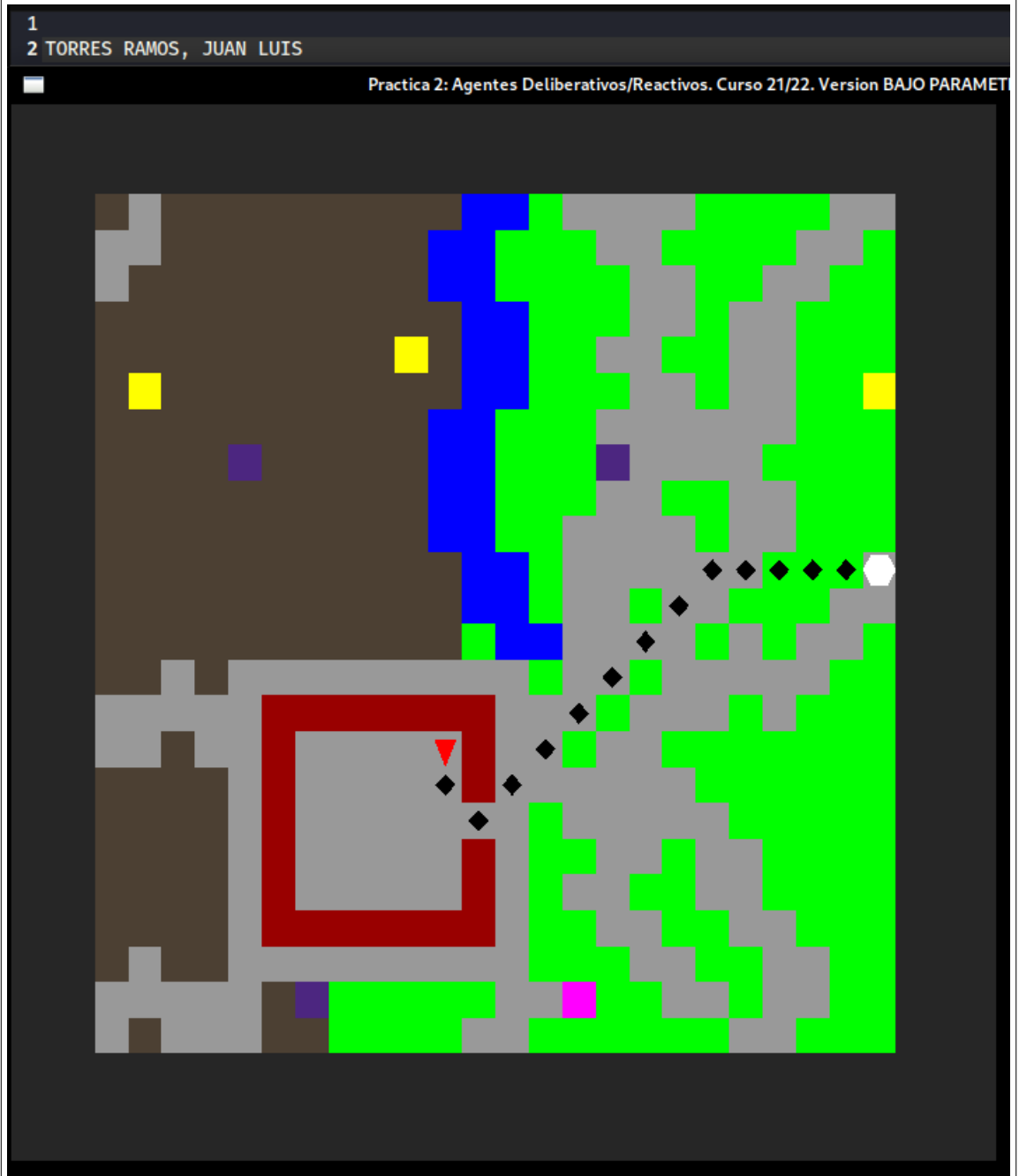


Instantes de simulación no consumidos	2930
Tiempo Consumido	0.007703
Nivel Final de batería	948
Plan	iiiiA A A A i i A A A A i i A A A A i i A A A A i A I A A A I A A A d d A A A A A A A i A A A A A A i i A A A A A A A A A A A A

Nivel 1-Óptimo en número de pasos

- (a) Rellena los datos de la tabla con el resultado de aplicar
`./practica2 mapas/mapa30.map 1 1 18 13 3 13 26`

ScreenShot



Instantes de simulación no consumidos

2981

Tiempo Consumido

0.356442

Nivel Final de Batería

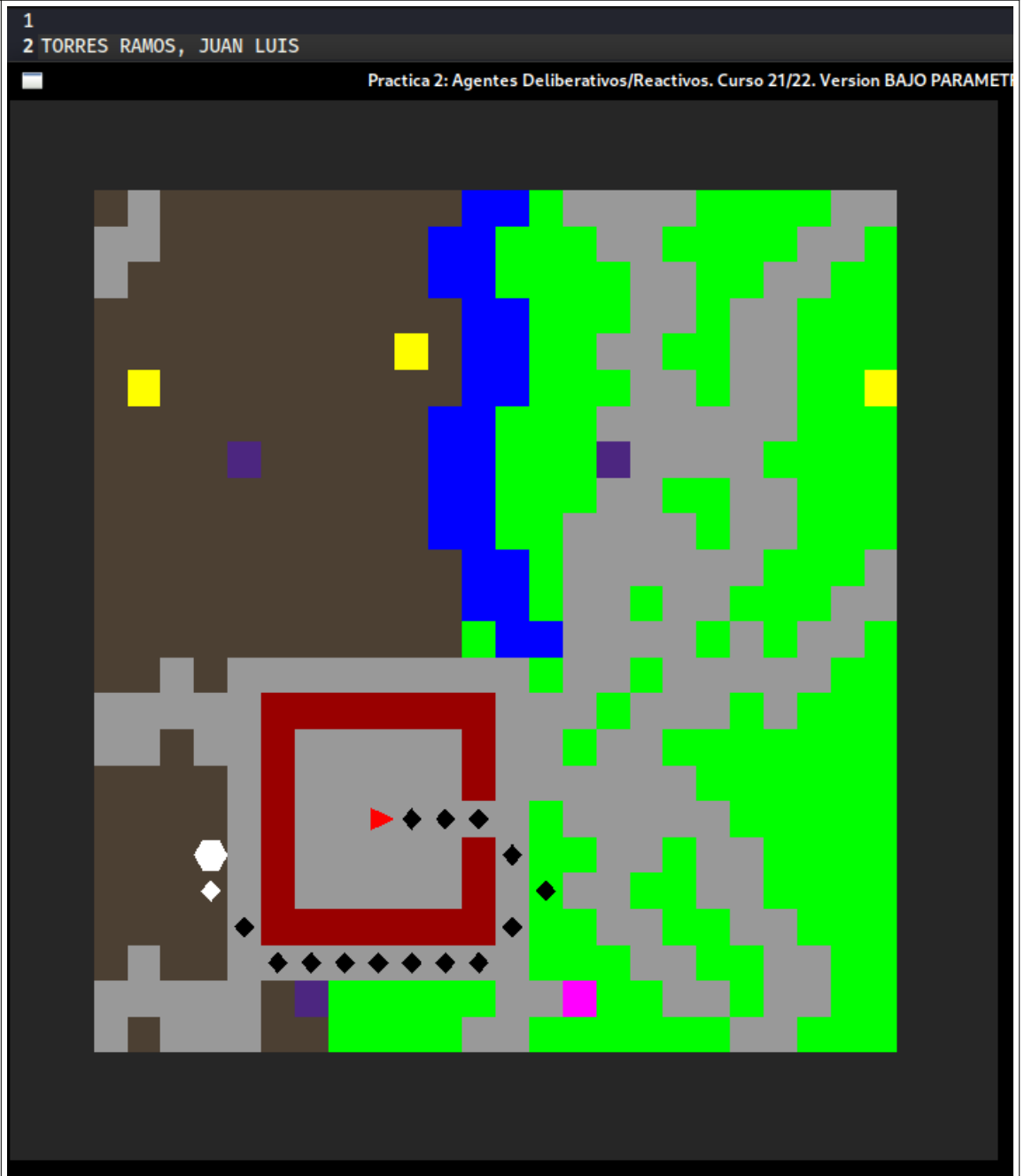
2685

Plan

d A i A I A A A A A A d A A A A A

(b) Rellena los datos de la tabla con el resultado de aplicar
./practica2 mapas/mapa30.map 1 1 20 11 3 21 6

ScreenShot



Instantes de simulación no consumidos

2977

Tiempo Consumido

0.922101

Nivel Final de Batería

2876

Plan

i A A A d A A D A A d A A A A A A d A A d A

Nivel 2-Óptimo en coste (un objetivo)

(a) Indica el algoritmo implementado para realizar este nivel

He implementado tanto el Algoritmo de Costo Uniforme con el de A*. CU, a pesar de tenerlo, no lo utilizo ya que es demasiado lento.

(b) Si usaste A*, describe la heurística utilizada

Si ya que $F(n) = g(n)$, siendo g el calculo del coste minimo. Es admisible ya que siempre la heurística $h(n)$ va a ser menor o igual (en el mejor caso) que la distancia real, ademas encuentra un camino del nodo origen O a nodo destino G cuyo coste es óptimo.

(c) ¿cambió la definición de concepto de estado con respecto al usado en los niveles 0 y 1?

Si ahora, en el renombrado ComprarEstadosCoste o ComparaEstadosEstrella comparo si el nodo tiene zapatillas o tiene bikini, además en el NodoCoste añado el valor de coste y en el NodoEstrella añado los valores f (el coste total) y g (coste del nodo origen hasta el actual). Tambien hay un valor h que calcularia la DM, PERO NO LA UTILIZO, si la utilizase la heurística no sera andimible cuando belkan se puede mover en 8 direcciones.

(d) ¿Has incluido dentro del algoritmo de búsqueda usado en este nivel que si pasas por una casilla que da las zapatillas o el bikini, considere en todos los estados descendientes de él que tiene las zapatillas y/o el bikini? En caso afirmativo, explicar brevemente cómo.

Si lo considero. Mientras creamos el arbol, veo priemro si el nbodo tiene zapatillas o bikini, y al atributo del nodo `current.st.TieneZapatillas/TieneBikini` → asigno a true. Esa compribacion ya que en `ComparaEstadosEstrella / ComparaEstadosCoste` comparo dichas variables y tengo que tenerlas en cuanta en cada nodo para coger la optima.

(e) Rellena los datos de la tabla con el resultado de aplicar
./practica2 mapas/mapa30.map 1 2 20 11 3 21 6

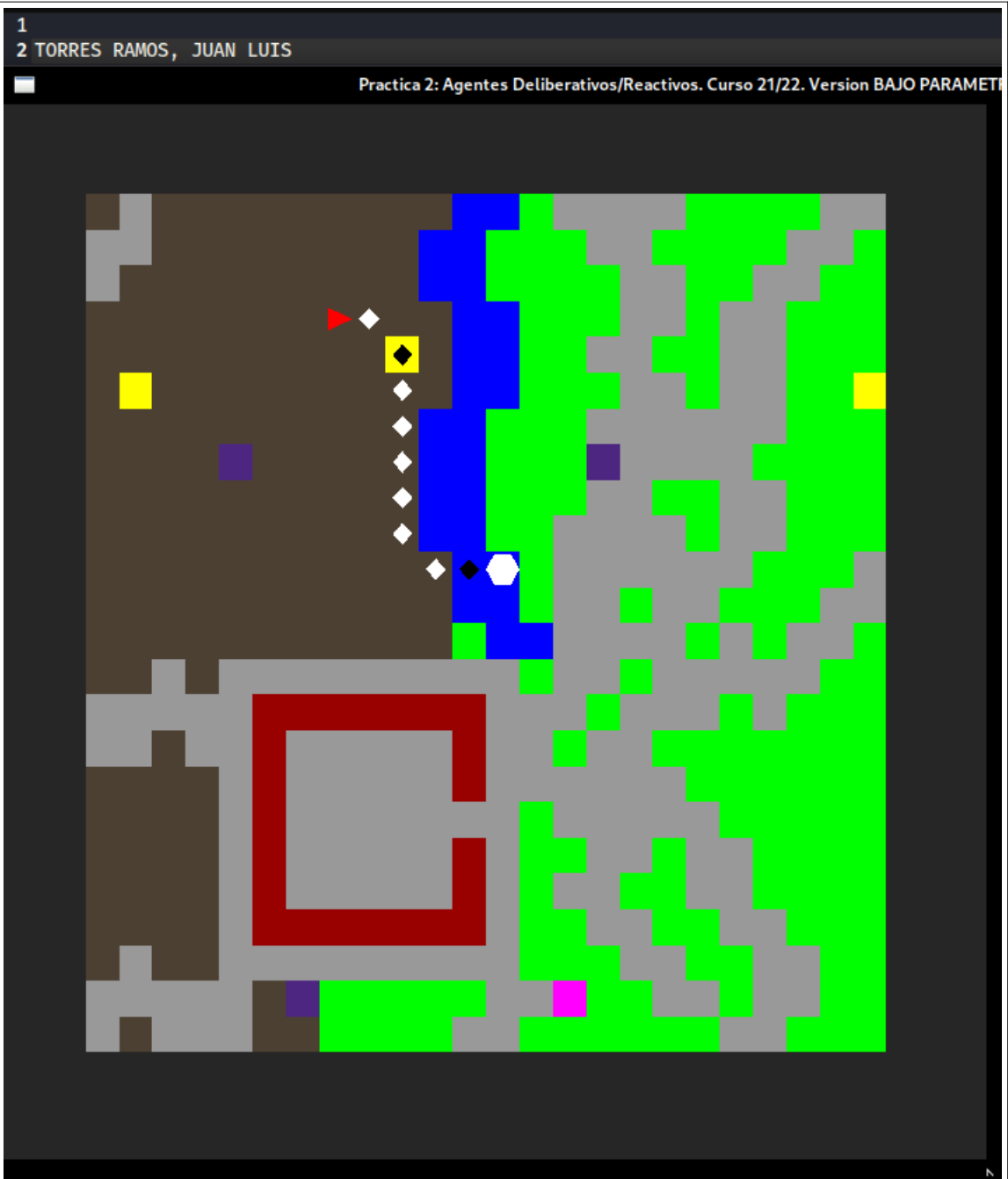
ScreenShot



Instantes de simulación no consumidos	2975
Tiempo Consumido	0.069451
Nivel Final de Batería	2976
Plan	i A A A d A d A A A D A A A A A A A D A A i A

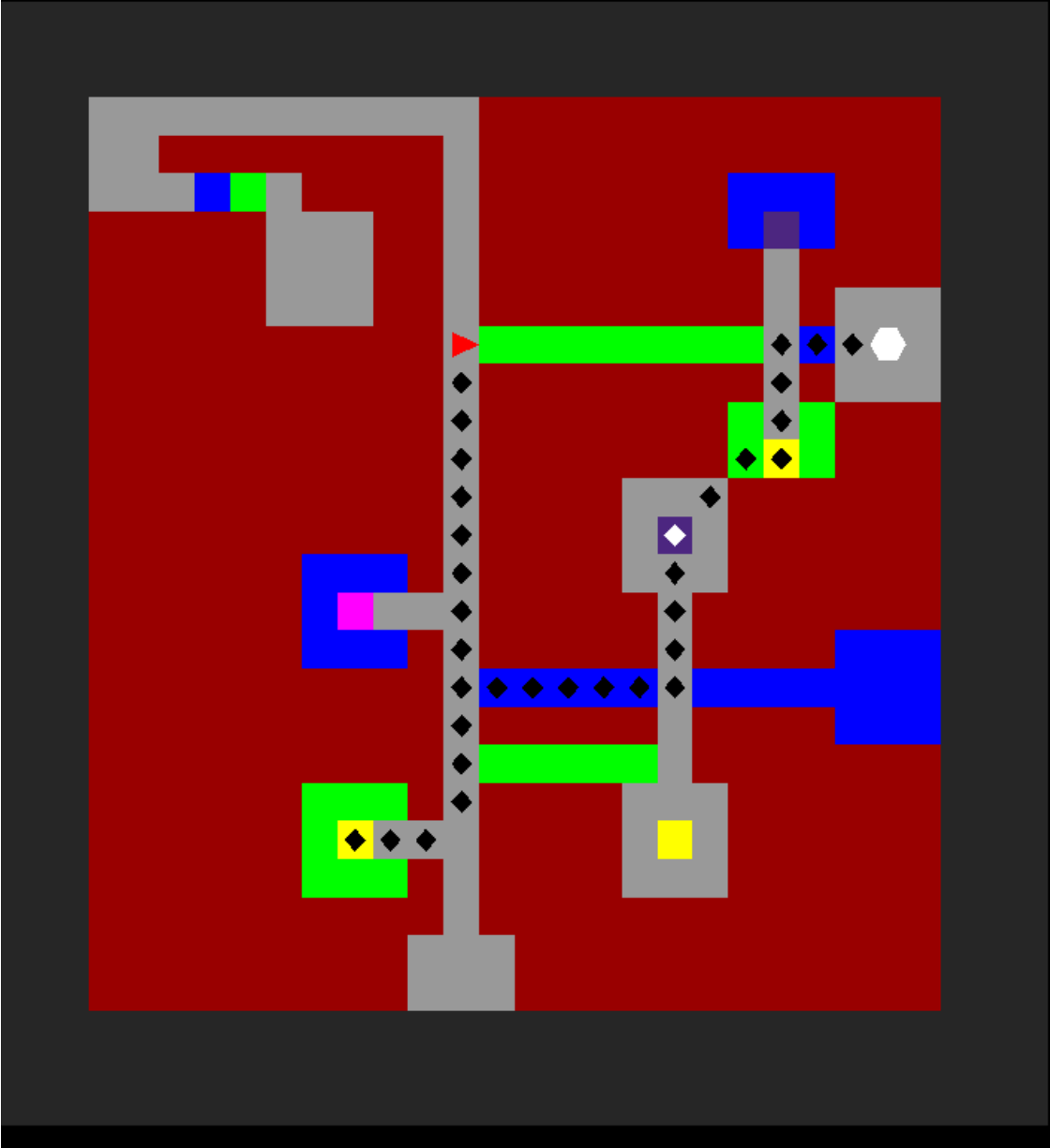
(f) Rellena los datos de la tabla con el resultado de aplicar
./practica2 mapas/mapa30.map 1 2 6 10 1 13 15

ScreenShot



Instantes de simulación no consumidos		2984
Tiempo Consumido		0.097239
Nivel Final de Batería		2968
Plan	d A d A d A A A A i A i A A	

(g) Rellena los datos de la tabla con el resultado de aplicar
./practica2 mapas/scape.map 1 2 9 13 1 9 25

ScreenShot	<div> <div>1</div> <div>2 TORRES RAMOS, JUAN LUIS</div> <div> <div></div> <div>Practica 2: Agentes Deliberativos/Reactivos. Curso 21/22. Version BAJO PARAMETRIZACION</div> </div> </div> 	
	Instantes de simulación no consumidos	2945
	Tiempo Consumido	0.05993
	Nivel Final de Batería	2878
	Plan	d D A A A A A A A A A A A A A d A d A A I I A A i A i A A A D A A A A A A I A A A A d A A d A I A A A D A A A

Nivel 3-Reto 1 (Max. Descubrir mapa)

- (a) Indica el/los algoritmos de búsqueda implementados para realizar este nivel

A*

- (b) Si usaste A*, describe la heurística utilizada. ¿Es admisible?

Lo uso , esta explicao para el nivel 2, al fin de cuentas, sigo la misma filosofia de CU, tengo en cuenta el costo del camino en $g(n)$.

- (c) Describe de forma simple la estrategia que has implementado para maximizar el descubrimiento del mapa. Qué papel juego el algoritmo de búsqueda dentro de tu estrategia.

La estrategia a seguir es la siguiente. En un principio belkan tiene un comportamiento reactivo, se va moviendo por el mapa descubriendo todo lo posible, cuando, en el rango de sus sensores encuentra una casilla de interes, unas zapatillas o un bikini , si no las tiene va a por ella y durante un intervalo las usa, tambien durante en ese intervalo que usa dicho objeto, si encuentra una casilla de recarga va hacia ella. Cuando acaba el intervalo, si en sus sensores encuentra el otro objeto que no ha cogido va hacia el, lo que significa que primero usa zapatillas cuando la encuentra, luego bikini , y asi sucesivamente

- (d) Si exploraste varias estrategias indica aquí una descripción general y la razón por las que las descartaste

Uso solo una estrategia.

- (e) Rellena los datos de la tabla con el resultado de aplicar
./practica2SG mapas/mapa75.map 1 3 11 47 0

Porcentaje de mapa descubierto	38.1511
--------------------------------	---------

- (f) Rellena los datos de la tabla con el resultado de aplicar
./practica2SG mapas/mapa100.map 1 3 63 31 3

Porcentaje de mapa descubierto	18.49
--------------------------------	-------

(g) Rellena los datos de la tabla con el resultado de aplicar
./practica2SG mapas/medieval.map 1 3 3 3 2

Porcentaje de mapa descubierto	0
---------------------------------------	---

Se cae por un precipicio ya que al calcular el camino no tiene en cuenta la casilla de precipicio. Facilmente se podria haber resuelto replanificando el camino como en el nivel 4. Teniendo en cuenta la casilla precipicio y si la encuentro delante pues borro y creo un nuevo plan

Nivel 4-Reto 2 (Max. Número de misiones)

- (a) ¿Qué algoritmo o algoritmos de búsqueda usas en el nivel 4? Explica brevemente la razón de tu elección.

A*

- (b) Explica brevemente qué política has seguido para abordar el problema de tener 3 objetivos.

Primero lo que hacemos es buscar 3 objetivos, belkan calcula el plan para ir al primer objetivo de la lista, y asi con los siguientes hasta acabar con todos. Cuando la lista de objetivos este vacia se vuelve a buscar nuevos objetivos.

- (c) ¿Bajo qué condiciones replanifica tu agente?

Mi agente replanifica cuando se ha chocado (sensores.colision a true, tengo que borrar el plan ya que estoy en una posicion distinta a la del plan), cuando no puedo avanzar dentor de un plan ya que hay algun elemento como un muro o no tener zapatillas para un bosque y cuando llega a su objetivo busca un nuevo plan.

- (d) Explica el valor que le has dado a la casilla desconocida en la construcción de planes cuando el mapa contiene casillas aún sin conocer. Justifica ese valor.

El valor de la casilla es como si fuera un terreno, por eso , cuando empezamos y calcula el primer plan, por lo general lo calculara sin tener en cuante los obstaculos del camino ya que no tiene info del mapa, cuando vaya caminando y se encuentre con un muro replanificara.

- (e) ¿Has tenido en cuenta la recarga de batería? En caso afirmativo, describe la política usada por tu agente para proceder a recargar.

La recarga de bateria la tuve en cuenten un principio pero vi que para mi no era más eficiente, no se pero yo lo que hacia era mientras esta haciendo los planes si en sus sensores encuentra una casilla recarga guardaba el punto. Si tenia poca bateria calculaba un plan para ir a esa casilla y ya y luego de recargar hasta que la bateria estuviese llena pasaba al siguiente punto. En teoria deberia calcular más objetivos, pero en mi caso no se dio ya que la recarga era fija, recargaba 1000 y al final cuando encontraba el siguiente objetivo volvía a la recarga por que se pasaba el limite, es decir no me era rentable con mi funcion recarga, que deberia haberla hecho hasta que la bateria estuviese llena.

En resumen lo hice pero lo deseche ya que sin bateria me calculaba más puntos.

- (f) ¿Has tenido en cuenta la existencia de aldeanos y lobos para definir el comportamiento del agente? En caso afirmativo, describe en qué sentido los has tenido en cuenta.

Los he tenido en cuenta cuando me chocaba. Cuando me chocaba no podía hacer interacciones, seguir con el plan o escribir en el mapa ya que el plan se estropeaba. Los considero como obstáculos y cuando se ponen en medio, belkan espera hasta que se muevan por así decirlo.

- (g) Añade aquí todos los comentarios que desees sobre el trabajo que has desarrollado sobre este nivel, qué consideras con son importantes para evaluar el grado en el que te has implicado en la práctica y que no se puede deducir de la contestación a las preguntas anteriores.

Lo dicho previamente sobre la batería, pero en verdad no tiene relevancia ya que lo borre, pero por lo menos lo planteé y lo hice.

(h) Rellena los datos de la tabla con el resultado de aplicar

**./practica2SG mapas/mapa50.map 1 4 28 25 3 39 45 9 16 38 13 27 23 31 18
45 31 35 7 12 6 40 7 20 6 10 25 41 30 14 31 26 24 38 26 38 20 44 14 17 40 45
3 4 9 33 44 17 3 3 11 42 13 26 18 38 25 33 26 46 46 36 14 36 31 17 34 8 22 44
41 16 11 44 17 29 32 42 21 46 19 40 34 45 24 46 7 44 32 21 30 14 39 15 22 11
9 13 27 20 8 45 5 6**

Instantes de simulación no consumidos	2572
Tiempo Consumido	11.8895
Nivel Final de Batería	533
Porcentaje de mapa descubierto	50.64
Objetivos	8

(i) Rellena los datos de la tabla con el resultado de aplicar

**./practica2SG mapas/mapa75.map 1 4 11 47 0 59 68 3 32 14 25 57 20 41 35
47 27 38 71 68 47 53 10 18 10 60 11 16 38 62 45 22 47 39 37 57 39 57 30 66
21 26 60 68 5 6 14 49 66 26 4 4 17 63 19 40 28 57 38 50 39 69 69 54 21 55 47
26 51 12 33 66 62 24 17 67 26 51 71 44 49 64 32 69 29**

Instantes de simulación no consumidos	2627
Tiempo Consumido	21.8277
Nivel Final de Batería	345
Porcentaje de mapa descubierto	27.4311
Objetivos	1

(j) Rellena los datos de la tabla con el resultado de aplicar

**./practica2SG mapas/marymonte.map 1 4 66 38 0 59 64 41 35 12 65 71 68
47 53 10 45 69 8 6 37 68 59 36 48 11 13 70 70 8 67 59 10 34 70 33 4 66 70 37
22 46 71 70 18**

Instantes de simulación no consumidos	2839
Tiempo Consumido	5.82544
Nivel Final de Batería	0
Porcentaje de mapa descubierto	10.4533
Objetivos	1

Comentario final

Consigna aquí cualquier tema que creas que es de relevancia para la evaluación de tu práctica o que quieras hacer saber al profesor.

LO PRIMERO Y MAS IMPORTANTE ES PARA EL NIVEL 3

No se, esto no es muy profesional pero tengo que comentartelo

No se porque pero a la hora de modularizarlo lo hice mal, el problema es en la recarga que no sale, es solo en vez de actualizar con `actual = SensorCasilla(sensores, actual);`

al principio se ha de hacer al final, si no no sale de la recarga .Abajo seria el codigo correcto. Las funciones y todo esta bien.

De todos modos las busquedas y eso me da porcentajes bajos para mapas grandes, por lo menos la hace, pero ten en cuenta eso, es solo cambiar una linea. Ha sido un descuido y por lo menos quiero que se valore algo de esta parte, aunque si no se valora lo entiendo.

```
if (sensores.nivel == 3) {  
  
    EmergenciaBelkan();  
  
    if (comienzo){  
  
        ActualizarMapa(sensores);  
        VistaAgente(sensores);  
        IntervaloBusqueda();  
  
        // modo busqueda = cuando hay plan  
        if (hayPlan and plan.size() > 0) { // hay un plan no vacio  
            accion = plan.front(); // tomo la siguiente accion del Plan  
            plan.erase(plan.begin()); // eliminamos la accion del plan  
        } else { // si no hay plan el robot se mueve aleatoriamente  
            accion = MoverAleatorio(sensores);  
        }  
  
    }  
  
    actual = SensorCasilla(sensores, actual);  
    MovimientoRepetido(accion, ultimaAccion);  
    comienzo = true;  
}
```

en A* NO UTILIZO MANHATTAN, es una funcion que devuelve 0 solo, puedes verlo en el codigo, solo estuve haciendo pruebas para 4 direcciones, pero como los resultado no eran exactos a los de InformacionAdicional ya que no se movia en 8 direcciones, lo comente solo y deberia haberlo borrarlo, no solo comentarlo.

Tambien me parece raro durante las prueba de autoevaluacion que en mapas con precipicios (medieval) en medio aunque calcule el plan por el y luego quiera replanificar, se acaba cayendo por que pasa por ahi. En teoria belkan no deberia pasar por ahi ya que lo interpreta como muro, se puede ver en casos en los que planifica por fuera y luego replanifica ya que no puede pasar por ahi, si no hubiese tenido en cuenta eso, cuando hiciese un plan por fuera el mapa se caeria. Es el unico caso en el que se ha caido que me ha parecido peculiar.

Tambien pasa en el nivel 3 , se cae pero en ese caso no replanifico. Deberia haber probado en todos los mapas pero en el caso del medieval no ha sido un mapa de prueba que haya usado, por lo que lo del precipicio falla.

Por lo general en mis pruebas belkan iba hacia los objetivos bien, y cuando chocaba recalcula bien la posicion y sigue con el objetivo, por lo general, me ha pasado el caso anterior con la evaluacion y tenia que comentartelo

Para el mapa 30 por lo menos a veces me ha salido mejores resultado para los puntos, 35-30 objetivos mas o menos me calculaba.

Funcionar funciona todo correctamente, y sobre todo he probado en los mapa de 30 y 50 teniendo en cuenta lo anterior, he implementado todos los algoritmos y para el reto 1 y 2 por lo menos hace lo pedido. Me he centrado en que sea funciona en esta practica.