

## INTELIGENCIA ARTIFICIAL

CURSO 2020-21

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

APELLIDOS Y NOMBRE	Ruiz Fernandez, Juan Carlos		
GRUPO TEORÍA	B	GRUPO PRÁCTICAS	3

#### 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 han 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):**

0,1,2,3,4
-----------


Nivel 0-Demo

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

ScreenShot		
	Instantes de simulación no consumidos	2908
	Tiempo Consumido	0.015625
	Nivel Final de Batería	1818
	Plan	AAAAIAAAAAIAAAAAIAAADDDAAAAADAA AADAAAAADA AADAAAIIAAIAAIAAIAAADDADADADAAAAA AAAAAAAAIAAAAAAA

Nivel 1-Óptimo en número de pasos

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

ScreenShot		
	Instantes de simulación no consumidos	2975
	Tiempo Consumido	0.062
	Nivel Final de Batería	1884
	Plan	I A A I A A A A A A A A A A A I A A A A A A A

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

ScreenShot



Instantes de simulación no consumidos		2975
Tiempo Consumido		0.094
Nivel Final de Batería		2971
Plan	D D A A A A D A A A A D A A A A A A A A D A A A	

## Nivel 2-Óptimo en coste 1 objetivo

(a) Indica el algoritmo implementado para realizar este nivel

Costo Uniforme (en mi practica y memoria lo nombré A\* por despiste de principiante, por no acordarme lo he arrastrado hasta el final. Por lo que cada vez que ponga A\* en mi practica o memoria me refiero a Costo Uniforme)

(b) Si usaste A\*, indica la heurística utilizada

(c) ¿cambió el concepto de estado con respecto al usado en el nivel 0 y 1? Si la respuesta es afirmativa, explica brevemente en que consistió el cambio.

Si, el estado ahora tiene en cuenta bikini y zapatillas las cuales son variables booleanas que modifíco en mi clase *supernodo* cuando cojo una u otra.

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


**Belkan mapas/mapa30.map 1 2 20 11 3 21 6**

ScreenShot		
	Instantes de simulación no consumidos	2974
	Tiempo Consumido	0.016
	Nivel Final de Batería	2974
	Plan	D D A A A A D A A A A D A A A A A A A A D A A A I A

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

ScreenShot		
	Instantes de simulación no consumidos	
	2986	
	Tiempo Consumido	
	0.078	
Nivel Final de Batería		
2965		
Plan	A A D A A A A A A A I A A A	

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

ScreenShot		
	Instantes de simulación no consumidos	2909
	Tiempo Consumido	0.0625
	Nivel Final de Batería	2674
	Plan	D A A A A A A A A A A A A A A D A A A I I A A A I A A A A A D A A A A A A I A A A A A D D A A A A A A D A A A A A A A A D A D A A A D D A A A D A A A A



### Nivel 3-Óptimo en coste 3 objetivos

(a) Indica el algoritmo implementado para realizar este nivel

Mismo que el anterior, Coste Uniforme

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

(c) ¿has usado un concepto de estado diferente al utilizado en los algoritmos empleados anteriormente? Si la respuesta es "Sí", describe brevemente cómo cambió y en que afecta al algoritmo usado.

Si, un array de booleanos, de tamaño 3, que representa los objetivos visitados con un 1. Las posiciones de los objetivos en la lista corresponde al mismo orden en el que aparecen en el contenedor si iteramos normalmente sobre el.  
El algoritmo ahora tiene como condición de parada que el nodo actual debe haber conseguido los tres objetivos, todos los elementos del array de booleanos de su estado deben estar a 1.  
Al final de ese mismo bucle hago la comprobación de si el nodo actual es alguno de los objetivos de la lista, y si lo es

(d) Incluye aquí todos los comentarios sobre las dificultades que te llevó la implementación de este nivel.

Voy a extraer tal cual lo he escrito en la memoria de la practica, adjuntada en el zip, mi recorrido sobre este nivel:

*Para este nivel pensé en un principio iterar sobre el list destinos. Lo implementé, y aunque no era problema de esto, no encontraba ningún plan.*

*Pensé en implementar un contador como condición de parada del while, incrementando éste al final si la casilla actual era un destino.*

*Con eso no tenia en cuenta en la secuencia si he recorrido el objetivo, además que no reconocía si era la secuencia de objetivos óptima y se basaría en mejor camino objetivo por objetivo y no para los tres. Por recomendación del profesor, incorporé a los nodos cuantos objetivos ha recorrido. De nuevo era otro contador y actualicé ComparaEstados además de modificar super nodo para devolver si uno > otro si uno tiene menos objetivos que otro.*

*Con esto hice que el while comprobara que el nodo actual no hubiese cogido los tres objetivos, pero ello no funciona. Ni si quiera entraba en el while. Problema mío seguramente.*

*De nuevo, estaba en el problema que no tenia en cuenta que objetivo había visitado y cual no. A partir de aquí no lo pensé demasiado bien y di demasiadas vueltas a una solución. Los errores que fui cometiendo los escribo a continuación:*

1) Cambié de estrategia a un array de booleanos en estado haciendo los respectivos



cambios en el resto de programa, el cual tampoco me funcionó (problema mío de dejarme algo por el camino) ya que me hacia un bucle infinito.

- 2) Implementé un set en super nodo para los visitados y comparar en el while con su size. También, cambié el for de comprobar los destinos con un find. Además de comprobar que el actual no sea esa casilla y si no la tiene.
- 3) El set no me funcionaba por comparaestado, si hiciera set tenía que meter en hpp el comparaestados y me daba errores de undefined reference por el orden de escribirlos.
- 4) Intenté con vector y hacer find yo mismo.
- 5) Con vector funcionaba, he cambiado el > de super nodo, pero no funcionaba bien. El recorrido no es óptimo, pero si lo quitaba me hacía bucle infinito.
- 6) Cambié el ultimo for para hacer que se almacene los destinos según su posición en el vector para compararlos uno por uno en el comparaestados y no con size.
- 7) Creé otro struct para comparar solo filas y columnas para así no hacer recursividad ya que en comaparaestados tenía otro comparaestados para ver la diferencia entre los objetivos.
- 8) Cambié el compara estados porque me daba fallos para el nivel dos, culpa del vector estaba vacío pues daba segmentation fault.
- 9) El fallo a esta altura era que no igualaba bien los destinos objetivo con actual.
- 10) Lo arreglé, pero no llegaba a ningún camino.
- 11) Pensé: El operador > de supernodo solo tiene en cuenta el gasto y no los objetivos.
  - 1) Si le añado que compare nvisitados tampoco termina pero si coge nodos con más objetivos pero con más costo.

Me di cuenta de toda la vuelta que estaba dando cuando la solución ya pasó por mis manos. Implementé de nuevo el array de booleanos en estado para saber, solo por índice en éste, cual había encontrado. Actualicé Comparaestados para que comparase cada uno de los booleanos que representan los objetivos.

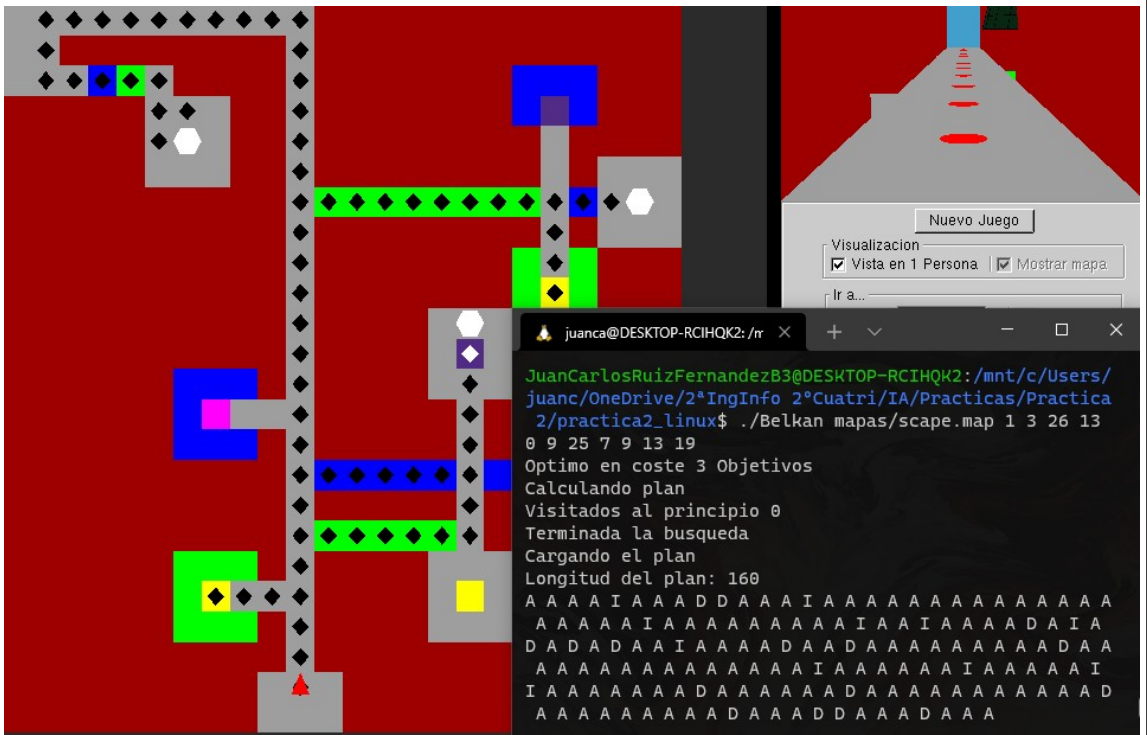
(e) Rellena los datos de la tabla con el resultado de aplicar  
**Belkan mapas/scape.map 1 3 13 19 2 8 13 21 13 11 13**

ScreenShot		
	Instantes de simulación no consumidos	
	Tiempo Consumido	
	Nivel Final de batería	
	Plan	A A A A A A A A A I I A A A A A I A A A A A A I A A A I I A A A A A A A A A A A A A

(f) Rellena los datos de la tabla con el resultado de aplicar  
**Belkan mapas/scape.map 1 3 13 19 2 8 13 21 13 16 12**

ScreenShot		
	Instantes de simulación no consumidos	2952
	Tiempo Consumido	0.03125
	Nivel Final de Batería	2908
Plan	A A A A A A A A A I I A A A A A I A A A A A A I A A A I I A A A A A I A D D A I A A A A A A A A	

(g) Rellena los datos de la tabla con el resultado de aplicar  
**Belkan mapas/scape.map 1 3 26 13 0 9 25 7 9 13 19**

ScreenShot		
Instantes de simulación no consumidos		2839
Tiempo Consumido		0.484375
Nivel Final de Batería		2388
Plan	<pre> A A A A I A A A D D A A A I A A A A A A A A A A A A A A A A A A I A A A A A A A A A I A A I A A A A D A I A D A D A D A A I A A A A D A A D A A A A A A A A A D A A A A A A A A A A A A A A A A A I A A A A A A I A A A A A I I A A A A A A A A D A A A A A A A A A A A A A A A A A D A A A A A A A A A D A A A A D D A A A D A A A A A </pre>	

## Nivel 4-Reto

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

El mismo que en el Nivel2 (costo uniforme aunque ponga Aestrella). Al calcular solo un objetivo ahorro en el tiempo de deliberación y tengo la libertad para deliberar por mi cuenta.

- (i) ¿Has incluido dentro del algoritmo de búsqueda usado en el nivel 4 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.

No, lo obvié. Ya que el algoritmo lo tiene en cuenta, pero mi fallo está en que solo lo recuerda para un recorrido y no entre los siguientes.

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

Cada vez que llego a un objetivo lo elimino de la lista, hasta que se queda vacía y es entonces cuando relleno la lista de objetivos.

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

Cuando encuentra el siguiente objetivo, y el nivel de bateria es inferior a 500, introduzco la casilla de bateria si la hay como primer objetivo.  
Si estoy en la casilla de recarga y mi nivel es menor de 2000, sigo ahi hasta que supere esa cantidad.  
Cuando se agotan los 3 objetivos los relleno, y los ordeno en función del más cercano.

- (l) ¿Qué coste le has asignado a la casilla desconocida en la construcción de planes cuando el mapa contiene casillas aún sin conocer?. Justifica ese valor.

1, ya que si le pongo valor menos a ese recorre demasiado camino perdiendo bateria y pudiendo quedarse sin ella. Si pongo mayor no recorrerá con prioridad la zona desconocida impidiendo que descubra mejores caminos.

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

Si, lo he explicado antes en la replanificación.

- (n) 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.

Creo, que si no queda muy convencido de mi implicación en esta práctica he incluido, para que demuestre que si he trabajado, la memoria que hice durante su desarrollo. Explico cuales fueron mis ideas para solucionar cada meta y problemas que me iba encontrando. Al final incluyo una parte de la experiencia sobre ésta. He de decir que sigo queriendo mejorar el Nivel 4. Gracias.

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

**Belkan mapas/islas.map 1 4 47 53 2 74 47 46 42 71 56 83 52 58 65 85 43 92  
39 81 68 91 48 21 95 92 14 88 64 43 61 28 78 30 44 22 18 27 55 41 16 90 10  
12 49 76 68 38 74**

<b>Instantes de simulación no consumidos</b>	2870
<b>Tiempo Consumido</b>	3.21875
<b>Nivel Final de Batería</b>	0
<b>Objetivos</b>	2

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

**Belkan 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**

<b>Instantes de simulación no consumidos</b>	2876
<b>Tiempo Consumido</b>	6.03125
<b>Nivel Final de Batería</b>	0
<b>Objetivos</b>	1