

INTELIGENCIA ARTIFICIAL

CURSO 2021-22

PRACTICA 1: Repertorio de preguntas para la autoevaluación de la práctica 1.

APELLIDOS Y NOMBRE	Alconchel Vázquez Daniel		
GRUPO TEORÍA	DGIIM	GRUPO PRÁCTICAS	1

Instrucciones iniciales

En este formulario se encontrarán preguntas que tienen que ver con (a) descripciones en lenguaje natural del comportamiento implementado en tu agente o (b) con resultados sobre ejecuciones concretas del software desarrollado por los estudiantes para problemas muy concretos.

En relación a los resultados sobre ejecuciones concretas, estas se expresarán usando la versión de invocación en línea de comandos cuya sintaxis se puede consultar en el guion de la práctica. Para ello, toma los nuevos mapas (**mapa30_e.map**, **mapa70_e.map** y **vertigo_e.map**) que se adjuntan con la autoevaluación y copialos en la carpeta **mapas** donde se encuentre tu software.

Poner en los recuadros la información que se solicita.

- (a) *Describe de una manera simple, breve y concisa (usando lenguaje natural) como has definido la forma en la que tu agente se mueve.*

Avanza en línea recta, hasta que se encuentra con un obstáculo. Entonces mira en qué dirección hay más casillas sin explorar, girando y avanzando en dicha dirección, hasta, nuevamente, encontrar un obstáculo. En caso de querer avanzar detrás de un muro o precipicio, entonces escoge una dirección de giro aleatoria y se mantendrá avanzando y girando en esa dirección hasta encontrar una apertura en el muro o precipicio por la que avanzar de zona.

- (b) *¿Tu agente va de forma activa hacia los objetos cuando estos aparecen en su sensor de visión? En caso afirmativo, describe la forma en que se implementa ese comportamiento activo.*

Sí. Cuenta con una función que va comprobando si hay alguna casilla interesante en su rango de visión. En caso afirmativo, almacena en que posición del sensor de terreno se encuentra y se lo pasa a otra función, la cuál planea como llegar hasta dicho objeto. Se le da mayor prioridad a la casilla de posicionamiento (si no esta posicionado), luego a zapatillas y bikinis (si no los tiene) y, por último, a la de recarga.

- (c) *¿Influye en el comportamiento que has definido el hecho de tener o no el bikini o las zapatillas? En caso afirmativo describe la forma en la que influye.*

Sí. Para empezar, si no tiene estos objetos y ve una casilla de zapatilla y/o bikini, el agente cancelará el movimiento que tenía planeado e irá por ellos. Además, en el apartado a) he

argumentado que mi agente decide hacia donde avanzar cuando choca con un obstáculo. Se considera obstáculo las casillas de bosque y agua si no tienes zapatillas y bikini, respectivamente (Este concepto de obstáculo desaparece si ve un objeto, si aparece rodeado de bosque o agua, o si está siguiendo un muro o precipicio con el objetivo de atravesarlo).

(d) *¿Has tenido en cuenta en el comportamiento la existencia de casillas que permiten la recarga de batería. En caso afirmativo describe como lo has tenido en cuenta.*

Sí, pero no demasiado. El agente considera las casillas de batería como interesantes, tal y como he indicado en b), pero solo pasa por encima de ellas. No se queda parado en la casilla recargando, simplemente, una vez que pasa por encima, sigue su camino.

(e) *¿Has definido alguna estrategia para intentar eludir las colisiones con los aldeanos y los lobos?*

Sí, pero no es demasiado compleja. Simplemente si tienes un aldeano o lobo enfrente (en la posición 2 de sensores de superficie), gira aleatoriamente hacia un lado y continua su camino. Esto puede provocar que gire, avance y luego, al intentar volver a avanzar, se choque con un lobo, el cuál ha decidido moverse a la misma casilla (si le ocurre con un aldeano simplemente se activa el sensor de colisión y se queda quieto)

(f) *¿Has incluido comportamientos que son específicos para los niveles 3 y 4? Describe los comportamientos y brevemente las razones que te impulsaron a incluirlos.*

Los comportamientos específicos de estos niveles son los descritos en el apartado anterior. Simplemente gira aleatoriamente para tratar de esquivar a un aldeano o lobo. Por otro lado, desde nivel 1 en adelante, tiene una matriz auxiliar donde almacena lo que ve y después la roto y traslado para superponerla con mapaResultado, una vez posicionado.

(g) *Describe cuáles son los puntos fuertes de tu agente.*

En caso de no ser un mapa con obstáculos excesivamente cercanos entre ellos, el agente siempre avanza hacia las zonas menos descubiertas con bastante eficiencia.

(h) *Describe cuáles son los puntos débiles de tu agente.*

En caso de tener muchos obstáculos, se mete en bucles de los que no puede salir. Por ejemplo, en vértigo, al aparecer entre dos precipicios, tal y como he explicado en a), decide girar hacia un lado aleatoriamente y se mantendrá girando y avanzando en ese sentido hasta encontrar dicha apertura, pero el agente solo entiende como apertura si hay un muro (o precipicio), un hueco vacío y luego otro muro o precipicio, por lo que en vértigo no sabe salir de entre dos acantilados (de hecho si no fuera por la función de inferir en las casillas, probablemente haría cero siempre, a no ser que aparezca en una casilla muy beneficiosa, pero aun así, no haría mas

de 20)

Por otro lado, también ocurre (con menos frecuencia) que sigue los bordes del mapa (como si tratara de buscar una apertura) y se choca con bosques y agua (sin los objetos de zapatos y bikini) y, o bien lo considera obstáculo, metiéndose en un bucle del que no sabe salir, o sigue de frente, gastando gran cantidad de batería.

Por último, no optimiza las recargas de batería.

- (i) *Incluye aquí todos los comentarios que desees expresar sobre la práctica que no hayas descrito en las preguntas anteriores.*

No tengo mucho más que explicar sobre el funcionamiento (Todo los detalles más específicos están en el PDF adjunto con los .cpp y hpp). Quería simplemente usar este hueco para comentar que, en mi caso, se me ha atragantado bastante la práctica. Había algunas directrices que no quedaban del todo claras (por ejemplo, no se puede poner previamente los bordes, pero puedes rellenar todo los huecos con '?' con precipicios al acabar, lo cual es prácticamente equivalente y, a mi parecer, resulta una incoherencia). Entre esto y que no tenía las ideas muy claras de como hacer un agente reactivo, me han llevado a cambiar el código y la funcionalidad del agente más de 7 u 8 veces (y con funcionalidad me refiero a la eucarística que usa para moverse al completo, no simples funciones), como consecuencia, el código ha quedado bastante estropeado, por lo que pido perdón, ya que hay partes que cuesta leerlas y entenderlas; y, por otro lado, no he quedado satisfecho con la funcionalidad final del agente, ya que tiene bastantes defectos, como bien he comentado en el apartado anterior.

- (j) *Ejecución 1: Ejecuta el siguiente comando en un terminal*

`./practica1SG mapas/mapa30_e.map 1 n 24 4 0`

para los 5 valores de **n**, desde 0 hasta 4 y coloca los resultados de porcentaje de mapa descubierto con dos decimales en la siguiente tabla. Si la ejecución da un error y no termina dando un resultado, pon "core" en la casilla de la tabla correspondiente.

n = 0	n = 1	n =2	n =3	n = 4
94.44	95	95	100	96.22

- (k) *Ejecución 2: Ejecuta el siguiente comando en un terminal*

`./practica1SG mapas/mapa70_e.map 1 n 33 19 0`

para los 5 valores de **n**, desde 0 hasta 4 y coloca los resultados de porcentaje de mapa descubierto con dos decimales en la siguiente tabla. Si la ejecución da un

error y no termina dando un resultado, pon "core" en la casilla de la tabla correspondiente.

n = 0	n = 1	n =2	n =3	n = 4
30.71	51.84	51.84	82.10	43.55

(l) Ejecución 3: Ejecuta el siguiente comando en un terminal

./practica1SG mapas/vertigo_e.map 1 n 79 79 0

para los 5 valores de **n**, desde 0 hasta 4 y coloca los resultados de porcentaje de mapa descubierto con dos decimales en la siguiente tabla. Si la ejecución da un error y no termina dando un resultado, pon "core" en la casilla de la tabla correspondiente.

n = 0	n = 1	n =2	n =3	n = 4
46.04	11.64	11.64	11.64	18.25