

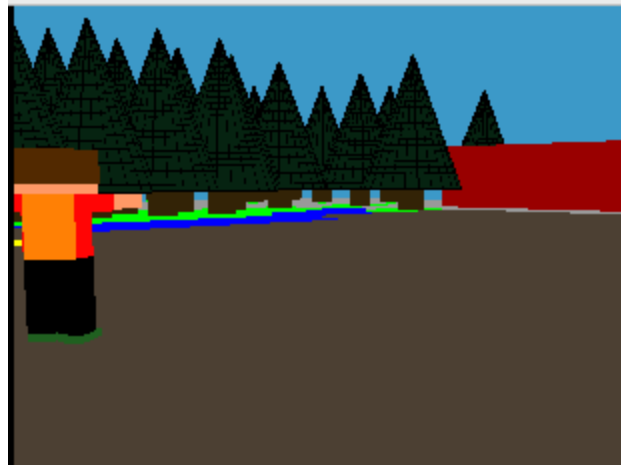
# INTELIGENCIA ARTIFICIAL

E.T.S. de Ingenierías Informática y de  
Telecomunicación

## Ejemplos prácticos: Práctica 2

**Agentes Reactivos/Deliberativos**

**(Los extraños mundos de BelKan)**



DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN E INTELIGENCIA  
ARTIFICIAL

UNIVERSIDAD DE GRANADA

Curso 2023-2024

## 1. Introducción

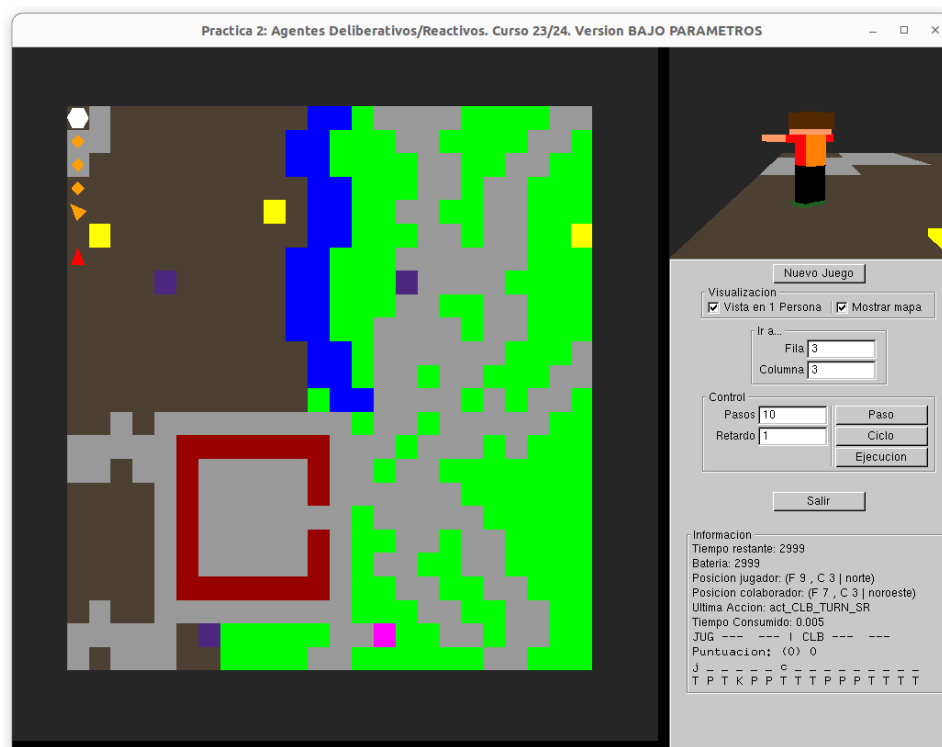
Este documento se aporta, de forma complementaria al material disponible de la Práctica 2, con el fin de aclarar el comportamiento esperado en cada uno de los niveles descritos en el guion mediante ejemplos concretos. Dado que el Nivel 0 se implementa siguiendo el “Tutorial: Práctica 2”, se mostrarán ejemplos prácticos a partir del Nivel 1.

## 2. Nivel 1

En el Nivel 1 se pide encontrar el camino óptimo en cuanto a número de acciones que lleve al agente colaborador a una casilla objetivo. Por tanto, el comportamiento que se espera tras implementar la búsqueda en anchura es que el jugador encuentre el camino que necesite el colaborador con un menor número de acciones independientemente del terreno que se encuentre (siempre que sea transitable). Esto es:

- Si el colaborador se encuentra en el campo de visión del jugador y no se encuentran obstáculos entre la posición del colaborador y el objetivo, el camino solución será lo más recto posible minimizando el número de giros, que contabilizan como acciones.

**Ejemplo 1: ./practica2 ./mapas/mapa30.map 1 1 9 3 0 7 3 6 3 3**



Instantes de simulación no consumidos: 2994  
Tiempo Consumido: 0.00564  
Nivel Final de Bateria: 2992  
Plan: cr cr cw cw I I (longitud 6)

- En el caso de que se encuentre algún obstáculo no transitable (por ejemplo, muros) entre el colaborador y el objetivo, el jugador deberá esquivarlo por el camino más corto y que minimice el número de giros, de nuevo, de forma independiente al coste de transitar cada terreno. Además, cuando el colaborador no esté en el campo de visión del jugador, este deberá acercarse a él para poder darle las órdenes del plan establecido.

**Ejemplo 2: `./practica2 ./mapas/mapa30.map 1 1 24 12 6 18 13 6 18 15`**



Instantes de simulación no consumidos: 2976

Tiempo Consumido: 8.36994

Nivel Final de Batería: 2969

Plan: L L W R L W R cW cr cr cr W L cW cW L cr r r W L cW I I  
(longitud 24)

- Otro ejemplo de uso de este nivel puede ser para llegar a objetivos a los que el jugador no puede acceder (por haber terrenos que no son transitables), pero el colaborador sí (siempre que esté en el campo de visión del jugador). Un ejemplo de esta casuística es el ejemplo 3:

Ejemplo 3: `./practica2 ./mapas/mapa30_nuevo.map` 1 1 8 9 4 5 5 6 3 3



Instantes de simulación no consumidos: 2994  
Tiempo Consumido: 0.005874  
Nivel Final de Batería: 2786  
Plan: r r W cr cw I (longitud 6).

Es importante tener en cuenta que la búsqueda en anchura **debe ser eficiente**. La búsqueda en anchura necesita mucha memoria para funcionar correctamente, por lo que la implementación debe ser lo más eficiente posible. **En esta búsqueda los abiertos se sacan en el mismo orden en el que entran, por lo que una implementación adecuada debería poder parar en cuanto se genera la solución, y no esperar hasta que esta se extraiga de abiertos.** También es importante que no se visiten nodos repetidos y solo se expandan los nodos la primera vez que se visitan. Esto ha de tenerse en cuenta al desarrollar el algoritmo para la práctica.

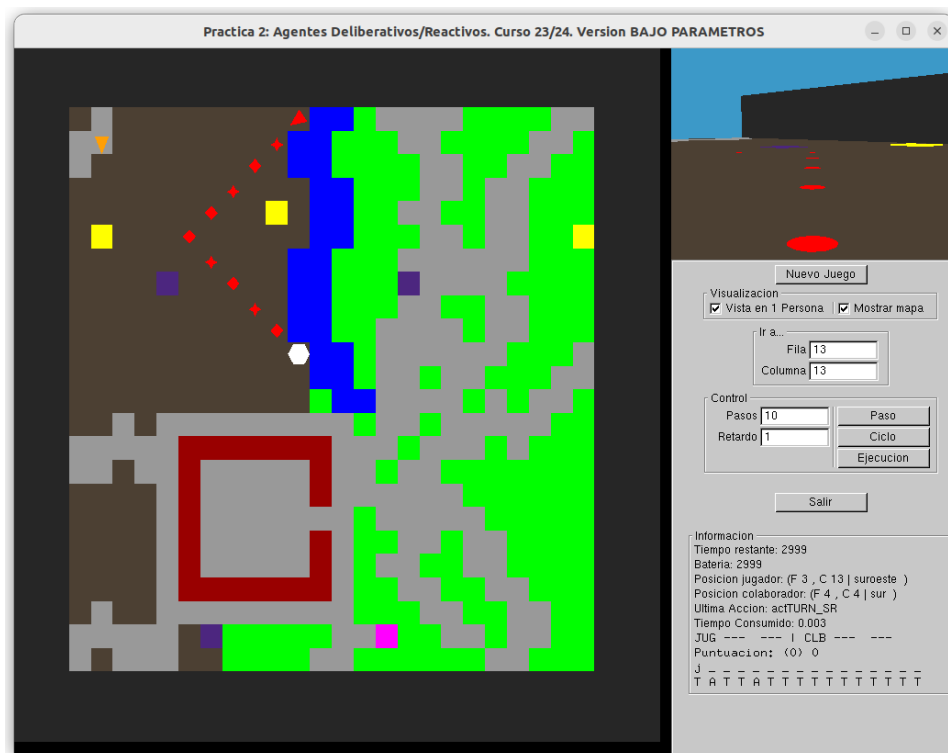
### 3. Nivel 2

La principal limitación del Nivel 1 es que se encuentra el camino óptimo en cuanto a número de acciones, lo cual no tiene porqué corresponderse con el mínimo consumo de batería dado que en el Mundo de Belkan nos encontramos con diferentes tipos de terreno que conllevan consumo diferente de batería. Esto puede producir que se consuma mucha más batería de la necesaria, llegando incluso a agotarse.

En este nivel, se pide encontrar el camino que lleve al jugador al objetivo que consuma menos batería. Para ello, se pide implementar una búsqueda por coste uniforme. Los resultados esperados serán, por regla general, más largos que los del Nivel 1 en cuanto a número de acciones pero optimizarán el consumo de batería de diferentes maneras: eligiendo caminos menos costosos (como suelo pedregoso) o utilizando el bikini o las zapatillas para minimizar el coste de transitar aquellos terrenos más costosos.

- En el ejemplo 4, vemos como el jugador toma un desvío para evitar pasar por el agua. Aunque para ello tenga que hacer más acciones (incluyendo giros) el consumo de batería es menor ya que atravesar el agua sin el bikini es muy costoso.

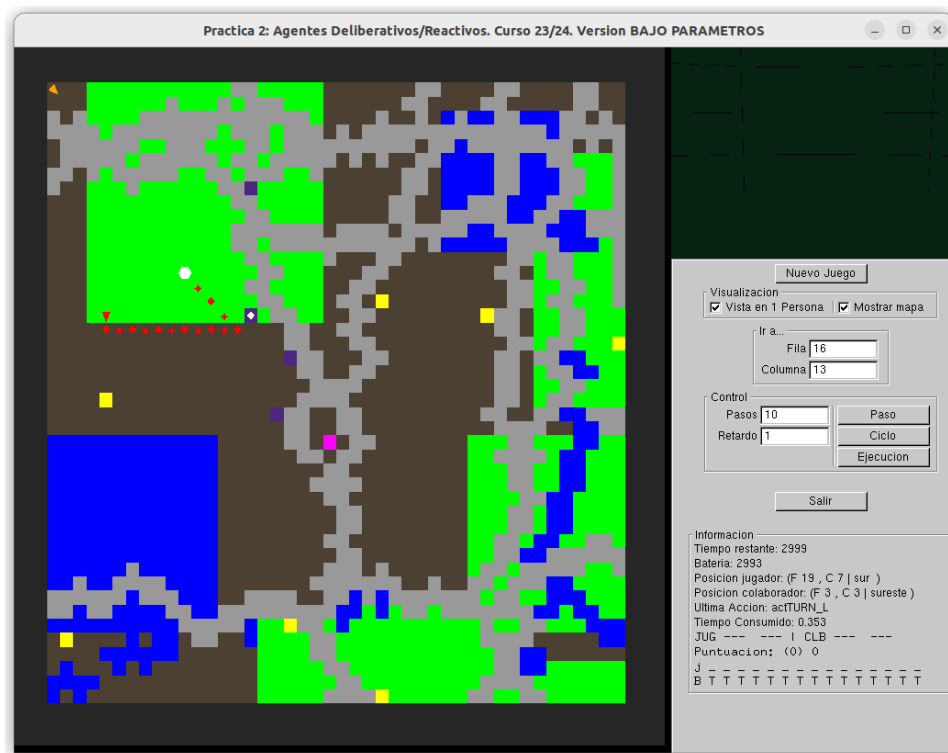
**Ejemplo 4:** `./practica2 ./mapas/mapa30.map 1 2 3 13 4 4 4 4 13 13`



Instantes de simulación no consumidos: 2992  
 Tiempo Consumido: 0.003496  
 Nivel Final de Batería: 2981  
 Plan: r R R W L R R W (longitud 8)

- En el ejemplo 5, vemos como el agente da un rodeo antes de entrar al bosque para conseguir las zapatillas, lo cual disminuirá el consumo de batería de andar por el bosque, consiguiendo un resultado óptimo en cuanto a consumo de batería.

**Ejemplo 5:** `./practica2 ./mapas/mapa50.map 1 2 19 7 6 3 3 3 16 13`



Instantes de simulacion no consumidos: 2982  
Tiempo Consumido: 0.352672  
Nivel Final de Bateria: 2888  
Plan: L W L R R R R R r L W L L W r r R R (longitud 18)

#### 4. Nivel 3

En este nivel se extiende el objetivo de encontrar caminos óptimos en cuanto a consumo de batería para el problema de encontrar un plan que lleve al colaborador al objetivo. Además, otra diferencia es que en este caso se pide implementar el algoritmo A\* en lugar del algoritmo de coste uniforme. En este caso, es importante usar una heurística apropiada que sea admisible. Si no es admisible, el camino que puede encontrar el A\* al objetivo podría no ser el de coste óptimo. En los ejemplos 6 y 7 vemos este comportamiento. En el ejemplo 6 aplicamos el A\* con una heurística admisible, mientras que en el ejemplo 7 lo aplicamos con una heurística no admisible.

### Heurística admisible

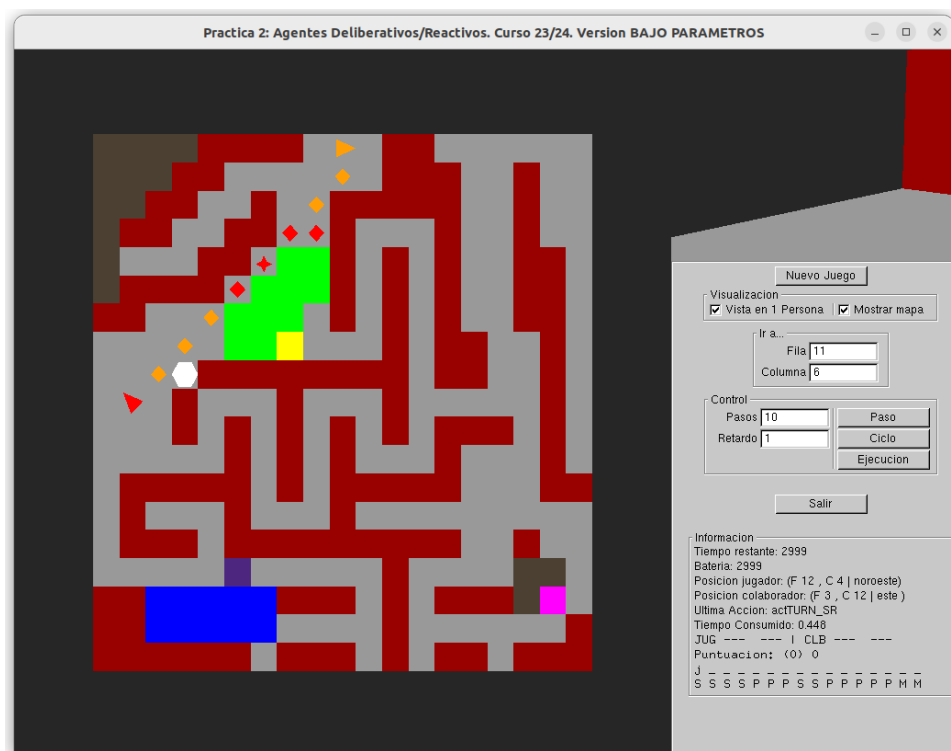
**Ejemplo 6:** `./practica2 mapas/lumi25.map 1 3 12 4 6 3 12 2 11 6`



Instantes de simulación no consumidos: 2974  
 Tiempo Consumido: 0.006833  
 Nivel Final de Bateria: 2970  
 Plan: L R cr I I I cW I I (longitud 9)

**Heurística no admisible**

**Ejemplo 7: ./practica2 mapas/lumi25.map 1 3 12 4 6 3 12 2 11 6**



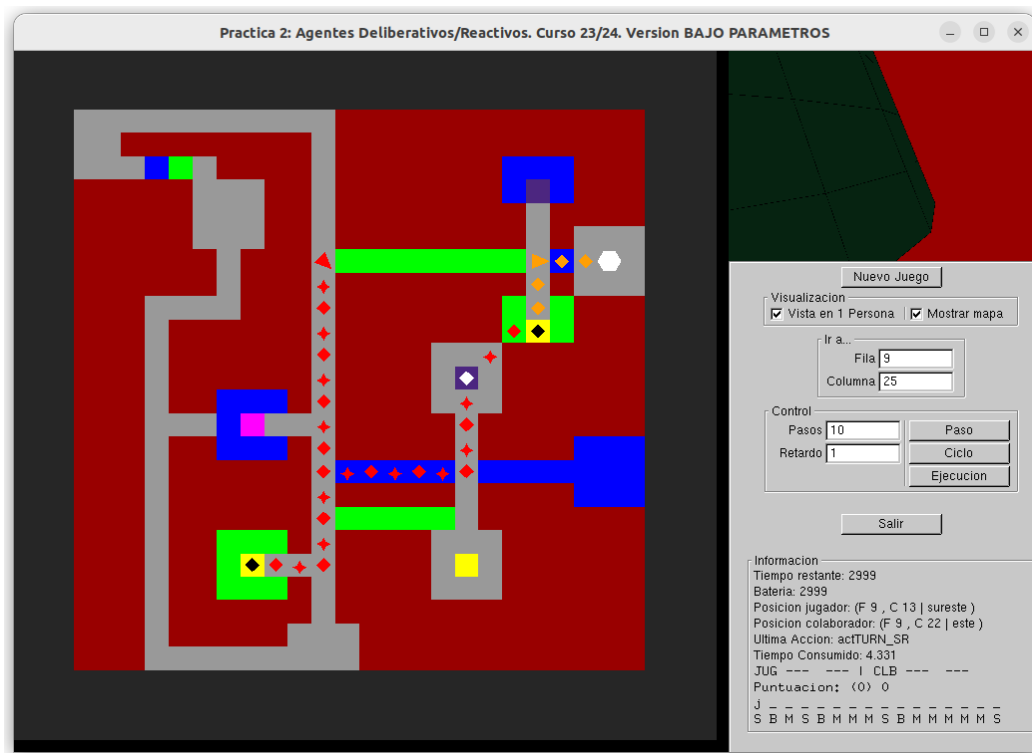
Instantes de simulación no consumidos: 2973  
Tiempo Consumido: 0.006833  
Nivel Final de Batería: 2969  
Plan: L R cr I I I cW I I (longitud 9)

Vemos que aunque ambos planes consiguen una solución (el colaborador llega al objetivo) en el segundo caso no es la solución óptima (se consumen 1 unidad más de batería). Por esto es importante que nos aseguremos de que la heurística que utilizamos es admisible.

Otro aspecto a destacar cuando nos referimos a camino óptimo en coste, es que pedimos aquel camino que implique un menor coste realizarlo, no aquel en el que terminemos con mayor batería. Por tanto, dedicar instantes de simulación a recargar batería con el fin de terminar la simulación con más batería es lo que se está buscando, si no encontrar un camino que de por sí consuma la menor batería posible (independientemente de si se pasa por una casilla de recarga o no).

En el siguiente ejemplo se muestra un caso donde para conseguir el camino óptimo cada uno de los agentes necesitará conseguir los objetos para reducir el consumo de batería. En este caso concreto, ambos necesitarán pasar por una casilla que les conceda el bikini.

**Ejemplo 8:** ./practica2 ./mapas/scape2.map 1 3 9 13 2 9 22 2 9 25



Instantes de simulación no consumidos: 2955  
Tiempo Consumido: 4.43  
Nivel Final de Batería: 2917  
Plan: r r R R R R R R r W r R L L W R L R R r r R R R L R R r R cr I cW I I  
cr I I I cW I cr cW cr cW I ( longitud 45)



En el ejemplo 9 se muestra un caso donde en al menos uno de los caminos óptimos es necesario ejecutar una acción `act_CLB_STOP` para evitar que los dos agentes colisionen entre sí.

**Ejemplo 9:** `./practica2 ./mapas/mapa30.map 1 3 21 10 4 18 18 6 18 11`



Instantes de simulación no consumidos: 2974

Tiempo Consumido: 3.79895

Nivel Final de Batería: 2974

Plan: L R cr I I I cW I IL W R L r R cr cW cr I I I I I cW cS r W L cW L cr  
cW cr cW I (longitud 26)

## 5. Nivel 4

Como ya se menciona en el guion de la práctica, en el nivel 4 se espera que el estudiante defina un agente que combine comportamientos reactivos y deliberativos con la intención de conseguir la máxima puntuación posible en la obtención de las misiones bajo las condiciones definidas en cuanto a número de instantes de simulación, valor de la batería y tiempo. A continuación vamos a mostrar algunos problemas usando algunos de los mapas proporcionados para ilustrar el como se valorarán los resultados obtenidos por el estudiante. La lista es la siguiente:

Id	Definición del Problema
P1	<code>./practica2SG mapas/mapa30.map</code> 1 4 4 4 6 12 12 2 8 8 16 6 26 9 25 16 3 3 5 10 16 14 18 10 21 4 7 4 24 4 12 3 6 15 25 18 8 19 15 14 23 15 26 8 10 24 19 26 25 7 16 11 22 15 20 15 22 19 10 20 4 13 26 24 9 6 26 10 17 19 25 13 24 20 26 19 12 18 8 23 9 13 6 5 8 16 12 5 3 14 11 22 11 8 6 17 7 4 21 3 23 4 15 5 7 23 21 19 4 15 6 13 24 17 6 26 5 4 24 10 16 17 13 20 22 9 26 22 22 24 14 4 24 26 7 18 6 21 9 9 18 6 19 15 16 21 3 14 13 10 25 13 17 24 7 20 14 14 9 21 5 18 20 20 19 10 5 18 18 21 9 22 20 19 7 15 26 20 10 17 17 19 8 23 8 9 9 5 20 3 20 11 6 23 16 8 26 14 17 4 8 25 14 13 25 14 8 5 20 21 4 18 14 25
P2	<code>./practica2SG mapas/mapa50.map</code> 1 4 28 25 4 28 20 2 36 23 39 8 46 26 39 34 26 37 18 46 3 46 3 3 10 17 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
P3	<code>./practica2SG mapas/mapa75.map</code> 1 4 11 47 0 6 47 4 9 51 19 33 29 31 33 31 54 43 51 67 59 68 57 29 47 16 68 6 29 4 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
P4	<code>./practica2SG mapas/mapa100.map</code> 1 4 63 31 6 63 32 2 66 40 75 24 85 36 83 6 60 10 33 11 84 7 86 40 68 77 79 91 19 33 76 27 55 47 62 36 51 95 91 63 71 14 24 13 80 15 21 51 83 61 29 63 52 49 77 52 76 40 89 28 35 80 91 6 94 52 8 19 66 89 34 6 6 23 85 26 53 37 76 51 66 53 3 43
P5	<code>./practica2SG mapas/pinkworld.map</code> 1 4 46 26 2 41 27 2 44 46 26 59 26 10 59 10 59 63 70 38 41 35 47 27 38 71 68 47 18 10 60 11 30 9 66 21 68 5 71 39 6 14 49 66 4 17 40 28 50 39 69 69 12 33 66 62 67 26 51 71 44 49 64 32 69 29 68 36
P6	<code>./practica2SG mapas/islas.map</code> 1 4 47 53 2 49 53 2 41 56 52 53 74 54 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
P7	<code>./practica2SG mapas/marymonte.map</code> 1 4 46 26 2 41 27 2 44 46 26 59 26 10 59 10 59 63 70 38 41 35 47 27 38 71 68 47 18 10 60 11 30 9 66 21 68 5 71 39 6 14 49 66 4 17 40 28 50 39 69 69 12 33 66 62 67 26 51 71 44 49 64 32 69 29 68 36
P8	<code>./practica2SG mapas/medieval.map</code> 1 4 3 3 2 9 20 2 9 88 21 75 34 62 48 16 57 28 72 24 89 27 84 51 61 93 13 48 9 87 84 39 91 19 33 76 87 39 57 29 84 53 96 96 63 3 88 54 32 44 19 75 29 95 96 48 21 33 87 41 9 20 13 48 9 87 84 39 91 19 33 76 87 39

Como se puede observar, los 8 ejemplos se ejecutan usando la versión del software que no usa gráficos (*practica2SG*).

También vamos a definir 3 valores (**Bajo**, **Medio** y **Alto**) para que el estudiante pueda estimar como es el funcionamiento de su agente en base a los objetivos alcanzados. Evidentemente, sabemos que el resultado exacto no depende solo del mapa, sino de la configuración inicial completa, pero si que servirá como guía para que el estudiante pueda situar sus resultados con los esperables en cada categoría. En la siguiente tabla se muestran los intervalos asociados a cada problema.

	Bajo	Medio	Alto
<b>P1</b>	60	167	443
<b>P2</b>	34	102	210
<b>P3</b>	27	62	113
<b>P4</b>	23	40	60
<b>P5</b>	17	25	40
<b>P6</b>	12	30	53
<b>P7</b>	17	25	40
<b>P8</b>	16	45	60

En el cálculo numérico de la calificación para el nivel 4 se procederá de la siguiente forma:

1. Se obtendrán **3 puntos** en las ejecuciones que consigan una puntuación igual o superior al valor **Alto**.
2. Aquellas ejecuciones que consigan una puntuación entre **Medio** y **Alto**, tendrá una calificación entre **2 y 3 puntos**. Su valoración concreta será el resultante de la interpolación líneal de su puntuación en función de los valores concretos para **Medio** y **Alto**.
3. De igual manera, para los que obtengan puntuaciones entre **Bajo** y **Medio** tendrán una calificación entre **1 y 2 puntos** y las puntuaciones inferiores a **Bajo** entre **0 y 1 punto**.
4. En la evaluación se aplicarán varias ejecuciones distintas (pudiendo ser sobre mapas distintos) y la valoración final para este nivel de una práctica concreta será la media sobre esas ejecuciones.
5. Las ejecuciones que no consigan terminar de forma normal la simulación<sup>1</sup> tendrán penalización y entrarán a formar parte en el cálculo de la media con valor **-0,1**.

---

<sup>1</sup> Se considerarán finalizaciones anormales aquellas que hagan que la ejecución termine antes que alguna de las condiciones que se especifican en la práctica (consumir el número de instantes de simulación, consumir los puntos de batería o consumir los 300 segundos de pensar) se satisfaga. Morir por caer por un precipicio también tiene la consideración de finalización anormal.