

Examen Final de Sistemas Inteligentes – Test Bloque 1
ETSINF, Universitat Politècnica de València, 26 de enero 2015
(2 puntos, tiempo estimado: 30 minutos)

Apellidos:								
Nombre:								
Grupo:	A	B	C	D	E	F	RE1	RE2

- 1) Dado un problema de búsqueda en el que todos sus operadores tienen el mismo coste, indica cuál de las siguientes afirmaciones es **correcta**:

- A. Un algoritmo de búsqueda con una heurística admisible devolverá la solución más corta
- B. Un estrategia en profundidad devolverá siempre la solución de menor coste
- C. La estrategia en anchura devolverá la solución más corta pero no la solución de menor coste
- D. La estrategia de coste uniforme devolverá la solución de menor coste pero no la solución más corta

-
- 2) Dados cuatro métodos de búsqueda: M1 aplica un algoritmo en anchura, M2 aplica un algoritmo de coste uniforme, M3 aplica un algoritmo A con una heurística admisible y M4 aplica un algoritmo A con una heurística no admisible, indica cuál es la respuesta **incorrecta**:

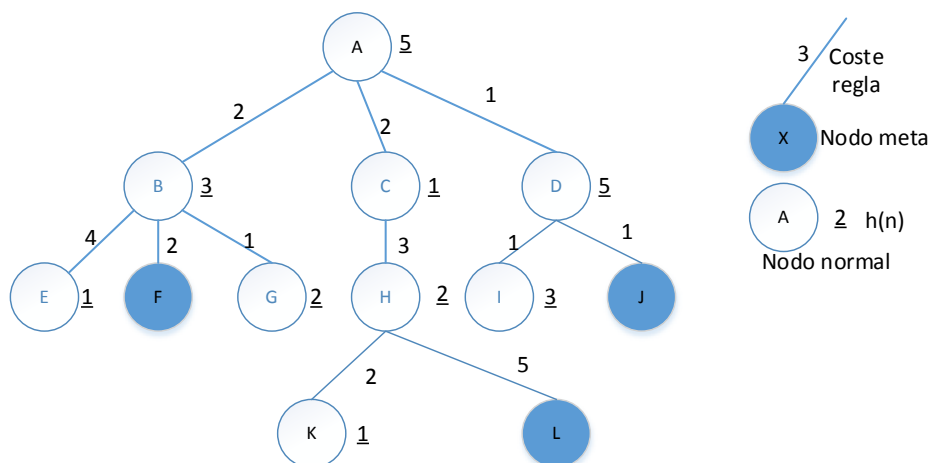
- A. M1, M2 y M3 garantizan que encontrarán la solución óptima independientemente del coste de las acciones
- B. M2 y M3 garantizan que encontrarán la solución óptima independientemente del coste de las acciones
- C. M3 expandirá menos nodos que M2
- D. M4 podría encontrar la solución óptima

-
- 3) La figura muestra un tablero donde R es un robot cuyo objetivo es desplazarse a la posición donde se encuentra el objeto O1 y luego desplazarse a la posición del objeto O2. El robot solo puede moverse horizontal o verticalmente. La figura muestra una instancia concreta de este problema, pudiendo estar R, O1 y O2 en cualquiera de las casillas del tablero. Sea n un nodo de un árbol de búsqueda que representa una situación particular de R, O1 y O2, y $\text{manh}(x,y)$ la distancia de Manhattan entre x e y , donde $x, y \in \{R, O1, O2\}$. Indica la afirmación que es **correcta**:

R			
			O1
	O2		

- A. $h(n) = \text{manh}(R, O1) + \text{manh}(R, O2)$ es una heurística admisible para este problema
- B. $h(n) = \text{manh}(R, O1) + \text{manh}(O1, O2)$ es una heurística admisible para este problema
- C. $h(n) = \text{manh}(R, O1) * 2$ es una heurística admisible para este problema
- D. No se puede definir una heurística admisible para este problema

- 4) Para el espacio de estados de la figura y dada una búsqueda de tipo A ($f(n)=g(n)+h(n)$) ¿cuántos nodos es necesario generar para encontrar la solución?

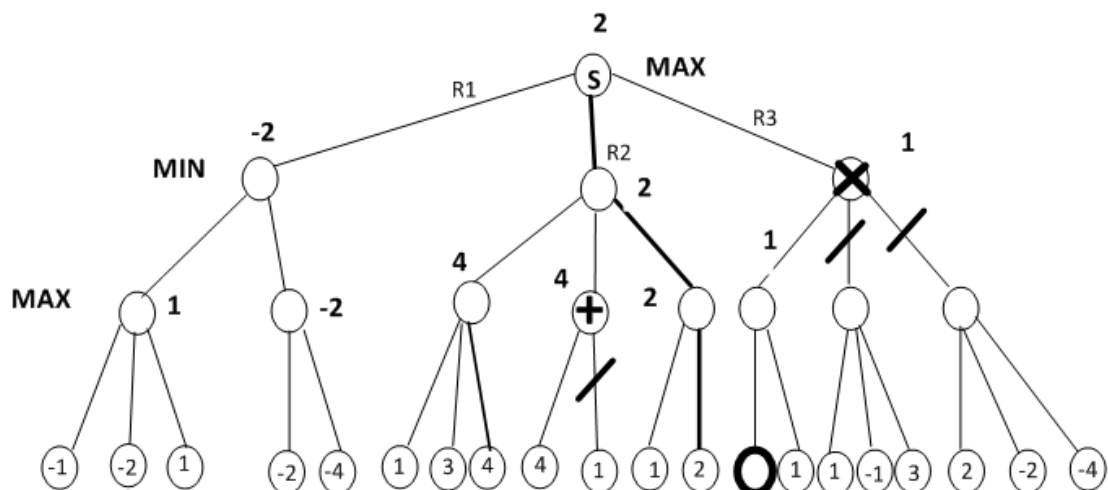


- A. 6
B. 8
 C. 10
 D. 12

- 5) Para el espacio de estados de la pregunta 4, indica cuál es la afirmación **correcta**:

- A. La aplicación de un algoritmo de tipo A devuelve la solución óptima
 B. La función $h(n)$ es consistente (monótona)
 C. Una estrategia de coste uniforme devolverá la misma solución que un algoritmo de tipo A
D. Ninguna de las anteriores

- 6) Dado el espacio de búsqueda de un juego que se muestra en la figura, ¿qué valor tendría que tener el nodo terminal señalado en negrita?



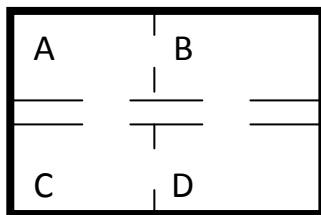
- A. $[-\infty, 1]$**
 B. 1
 C. $[2, +\infty]$
 D. No se puede determinar con los datos disponibles

Examen final de Sistemas Inteligentes – Problema Bloque 1
ETSINF, Universitat Politècnica de València, 26 de enero 2015
(3 puntos, tiempo estimado: 60 minutos)

Se dispone de una zona distribuida en 4 habitaciones (A, B, C, D) y un pasillo que conecta con todas las habitaciones (ver figura). Existen 2 robots encargados de servir tazas de té o café a diferentes personas. Cada robot tiene asociada una zona por la que se puede mover y de la que no puede salir. La zona del primer robot son las habitaciones A y B y el pasillo. La zona del segundo robot son las habitaciones C y D y el pasillo.

En la habitación A se encuentra la máquina dispensadora de té y café. En la habitación C se encuentra el armario de las tazas. Las personas que solicitan tazas de té o café solo pueden estar en las habitaciones B o D. Cada robot solo puede llevar una taza en cada momento. Para pasar una taza (vacía o llena) de un robot X a un robot Y, el robot Y no debe llevar otra taza.

Las operaciones que se pueden hacer en el sistema son: (a) el robot coge una taza vacía, (b) los robots se pasan una taza entre sí, (c) el robot llena una taza vacía de té o café según petición, (d) un robot se mueve de una habitación a otra contigua o al pasillo y (e) el robot le da la taza a la persona que ha realizado la petición.



Dado el siguiente patrón para representar la información dinámica del problema,

(local robot 1 pos ?p1 lleva ?l1 robot 2 pos ?p2 lleva ?l2 peticion [persona ?p ?t]^m)

donde:

?p1 ∈ {A, B, PAS}, ?p2 ∈ {C, D, PAS}, ?p ∈ {B, D}, ?l1, ?l2 ∈ {nada, vacia, te, cafe}, ?t ∈ {te, cafe}

a) (0.4 puntos) Describe una base de hechos inicial donde el robot 1 está en la habitación A y el robot 2 en la habitación D. Hay dos personas en la habitación B que han pedido un café y un té, respectivamente.

(local robot 1 pos A lleva nada robot 2 pos D lleva nada peticion persona B cafe persona B te)
 (conexion robot 1 A PAS) (conexion robot 1 PAS A)(conexion robot 1 A B) (conexion robot 1 B A)
 (conexion robot 1 B PAS) (conexion robot 1 PAS B)
 (conexion robot 2 C PAS) (conexion robot 2 PAS C)(conexion robot 2 C D) (conexion robot 2 D C)
 (conexion robot 2 D PAS) (conexion robot 2 PAS D)

b) (0.2 puntos) Describe un posible estado final donde todas las peticiones han sido servidas.

(local robot 1 pos A lleva nada robot 2 pos D lleva nada peticion)

:: el resto de hechos son estáticos y se mantienen

c) (0.8 puntos) Escribe una única regla que sirva para mover cualquiera de los dos robots entre dos puntos de su zona permitida.

```
(defule mover
  (local $?x robot ?rob pos ?p1 $?y)
  (conexion robot ?rob ?p1 ?dest)
=>
  (assert (local $?x robot ?rob pos ?dest $?y)))
```

d) (0.8 puntos) Escribe una única regla que permita pasar una taza (vacía o llena) entre los dos robots.

```
(defule pasar_taza
  (local robot 1 pos ?p1 lleva ?l1 robot 2 pos ?p2 lleva ?l2 $?y)
  (test (or (and (eq ?l1 nada)(neq ?l2 nada)) (and (eq ?l2 nada)(neq ?l1 nada))))
=>
  (assert (local robot 1 pos ?p1 lleva ?l2 robot 2 pos ?p2 lleva ?l1 $?y)))
```

e) (0.8 puntos) Escribe una única regla para servir una petición a una persona.

```
(defule servir
  (local $?x robot ?rob pos ?pos lleva ?lle $?y persona ?pos ?lle $?z)
=>
  (assert (local $?x robot ?rob pos ?pos lleva nada $?y $?z)))
```