

**Sistemas Inteligentes – Test Bloque 1 (tipo A)**  
**ETSINF, Universitat Politècnica de València**  
**6 noviembre 2018**

**Apellidos:**

**Nombre:**

**Grupo:**      A      B      C      D      E      F      G      4GIA

**Marca solo una respuesta entre las opciones dadas. Cada acierto sumará 1 punto y cada error descontará 1/3 puntos (las cuestiones no contestadas no afectarán a la nota)**

- 1) Sea la siguiente regla para calcular el Máximo Común Divisor (mcd) de dos números enteros positivos. Indica la respuesta **CORRECTA**:

```
(defrule mcd
  ?a <- (num ?n1)
  ?b <- (num ?n2)
  (test (> ?n1 ?n2))
=>
  (retract ?a)
  (assert (num (- ?n1 ?n2))))
```

- A. **Calcula correctamente el mcd quedando un hecho 'num' con dicho valor**
- B. Es necesario añadir una regla parada sin prioridad para que el sistema no entre en una ejecución sin fin
- C. Es necesario añadir una regla parada con prioridad para que el sistema no entre en una ejecución sin fin
- D. Ninguna de las anteriores es correcta

- 2) Dado un SBR cuya Base de Hechos inicial es (lista b a c c a b b a resto), y una única regla:

```
(defrule pasar
  ?a <- (lista $?x ?y ?y $?x $?z resto $?m)
=>
  (retract ?a)
  (assert (lista $?x $?x $?z resto $?m ?y)))
```

El contenido final de la Base de Hechos será:

- A. (lista b a a b b a resto c)
- B. (lista b a resto c a b)
- C. (lista b a b a resto c b)
- D. **Ninguna de las anteriores**

3) Dado el siguiente hecho: (Dueños coches a b c dueño P coches d dueño Q coches e f dueño R), donde se relaciona los coches y posteriormente su dueño ¿Cuál de los siguientes patrones serviría para obtener el nombre del dueño de un solo coche?

- A. (Dueños \$?x coches ?a dueño \$?z)
  - B. (Dueños \$? coches ? dueño ?z \$?)
  - C. (Dueños \$?x coches ?a dueño ?z \$?x)
  - D. (Dueños \$? coches ? dueño ?z)
- 

4) Sea un SBR, con la BH inicial {(lista A B C A B C C B A C B A)}, y la siguiente regla:

```
(defrule regla1
  ?f1 <- (lista $?x1 ?y $?x2 ?y $?x3)
    (test (> (length $?x2) 0))
    (test (not (member ?y $?x2))))
=>
  (retract ?f1)
  (assert (lista $?x1 ?y ?y $?x3)))
```

Si se ejecuta este SBR, el resultado es:

- A. Una lista que solo contiene letras A
  - B. Una lista que solo contiene letras B
  - C. Una lista que solo contiene letras C
  - D. Dependerá de la estrategia de control que se aplique, sea anchura o profundidad.
- 

5) Dado el hecho (prueba 1 2 3 4 5 6 7 8 9 10) y la regla :

```
(defrule regla1
  ?f1 <- (prueba $?a $?c)
=>
  (retract ?f1)
  (assert (lista $?c)))
```

En el primer ciclo inferencial:

- A. No se producirá ninguna instanciación
  - B. Se producirán 9 instanciaciones
  - C. Se producirán 10 instanciaciones
  - D. Se producirán 11 instanciaciones
-

- 6) Dada la base de hechos inicial: BH={{(lista 3 5 2 5 3 4 2 9 8 8 9 6) (numero 5) (repeticiones 0)}} y la siguiente regla para calcular el número de repeticiones de un elemento de una lista de números naturales

```
(defrule REGLA
  ?f1 <- (lista $?a ?b $?c)
  ?f2 <- (numero ?x)
  ?f3 <- (repeticiones ?z)
  (test (= ?b ?x))
=>
  (assert (lista $?a $?c))
  (assert (repeticiones (+ 1 ?z))))
```

Si nuestro objetivo es obtener una base de hechos final (tras la ejecución sucesiva de la regla) en la cual el hecho (repeticiones ...) solo puede aparecer una vez (conteniendo el número de repeticiones del número indicado en el patrón (numero ..) en la lista). ¿Cuál de las siguientes afirmaciones es CIERTA para obtener nuestro objetivo?

- A. La regla es correcta
- B. Sería necesario añadir (retract ?f1)
- C. Sería necesario añadir (retract ?f1) y (retract ?f3)
- D. Sería necesario añadir (retract ?f3)

- 
- 7) Sea un SBR formado por BHinicial={{(lista 2 1 5 3)}}, y las siguientes reglas:

```
(defrule R1
  (declare (salience 200))
  ?f <- (lista $?x ?z ?y $?w)
  (test (< ?z ?y))
=>
  (assert (lista $?x ?z ?y $?w)))
```

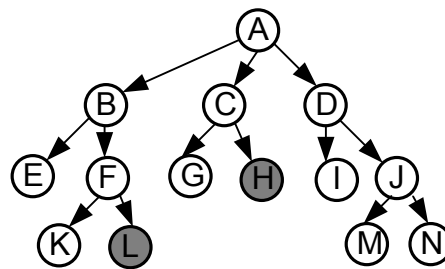
```
(defrule R2
  (declare (salience 50))
  ?f <- (lista $?x ?z ?y $?w)
  (test (>= ?z ?y))
=>
  (assert (lista $?x ?z ?y $?w)))
```

```
(defrule final
  (declare (salience 150))
  (lista $?list)
=>
  (halt))
```

tras el primer *pattern-matching*, ¿como quedarían ordenadas las instancias el Conjunto Conflicto (Agenda)?

- A. Una instancia de la regla R1, una instancia de la regla final, y dos instancias de la regla R2
  - B. Dos instancias de la regla R2, una instancia de la regla R1, una instancia de la regla final
  - C. Una instancia de la regla final, dos instancias de la regla R2, una instancia de la regla R1
  - D. Una instancia de la regla final
-

- 8) Considerando el siguiente árbol de búsqueda, ¿en qué orden se generarían los nodos y qué nodo meta se encuentra mediante un procedimiento de búsqueda por profundidad iterativa?

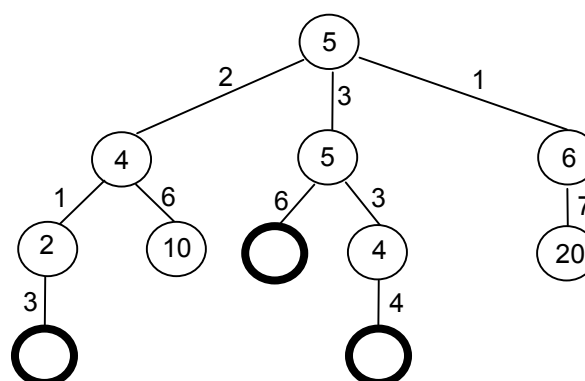


- A. ABCDEFKL y encuentra nodo meta L
- B. ABCDEFGHIJKL y encuentra nodo meta H
- C. AABCDABCDEFHG y encuentra nodo meta H
- D. ABCDEFGH y encuentra nodo meta H

- 9) Dados dos algoritmos A\* para un mismo problema, A1 con heurística  $h_1(n)$  y A2 con heurística  $h_2(n)$ , tal que  $\forall n, h^*(n) \geq h_2(n) > h_1(n)$ .

- A. Es seguro que A1 tardará menos que A2
- B. Es seguro que A1 expandirá menos nodos que A2
- C. La solución encontrada por A2 será mejor que la encontrada por A1
- D. Ninguna de las anteriores es cierta

- 10) Sea el árbol de la figura donde los nodos de trazo grueso son nodos meta, el valor dentro del nodo es el valor de la función heurística aplicada a cada nodo y el valor de los arcos es el coste del operador correspondiente. Indica la respuesta **CORRECTA**:



- A. La heurística es admisible y consistente
- B. La heurística no es admisible ni consistente
- C. Aplicando un algoritmo de tipo A se encuentra la solución óptima
- D. Ninguna de las opciones anteriores es correcta

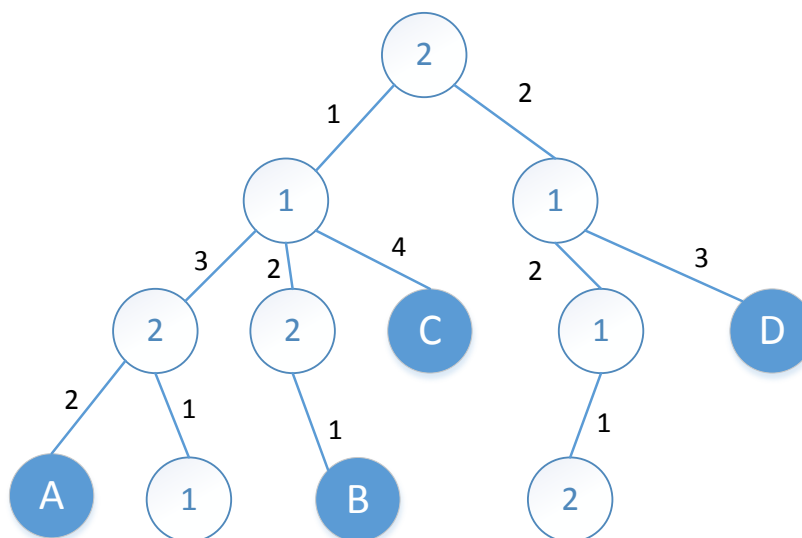
11) Sea un problema de búsqueda donde los operadores tiene distinto coste. Existe un nodo solución,  $G1$ , en el nivel  $d1$  del árbol de búsqueda y un nodo solución,  $G2$ , que es óptimo y se encuentra en un nivel  $d2$ , tal que  $d2 > d1$ . Indica la respuesta **CORRECTA**:

- A. La complejidad temporal de anchura respecto al número de nodos generados es  $O(b^{d1})$
- B. Una estrategia en profundidad nunca devolverá la solución  $G1$
- C. Una estrategia por profundidad iterativa nunca devolverá la solución  $G1$ .
- D. Una estrategia de coste uniforme devolverá siempre la solución  $G2$

12) En el árbol de búsqueda que se genera con un algoritmo de tipo A\* tenemos dos nodos,  $n1$  y  $n2$ , que se corresponden con dos estados repetidos. Se sabe, además, que  $n1$  es un nodo que se encuentra en el camino óptimo a un nodo solución,  $G$ , mientras que  $n2$  no está en el camino óptimo a  $G$ . Indica la respuesta **INCORRECTA**:

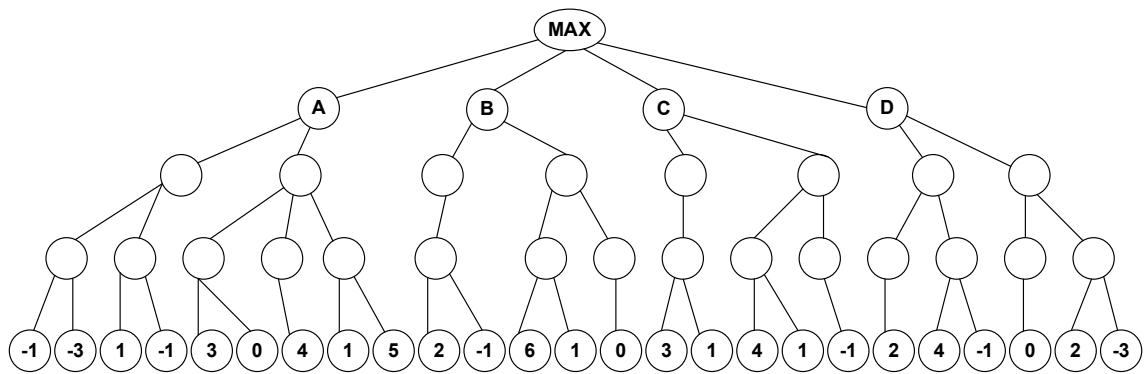
- A. Se cumple siempre  $f(n1) \leq f(G)$
- B. Se cumple siempre  $g(n1) < g(n2)$
- C. Se cumple siempre  $h(n1) < h(n2)$
- D. Se cumple siempre  $h(n2) \leq h^*(n2)$

13) Sea el siguiente árbol de búsqueda, donde el valor dentro del nodo denota el valor de una heurística para dicho nodo, y el valor junto a una flecha el coste de dicho operador. Los nodos etiquetados como A, B, C y D son nodos meta. Si se realiza una búsqueda de tipo A, que solución se obtendría en primer lugar.



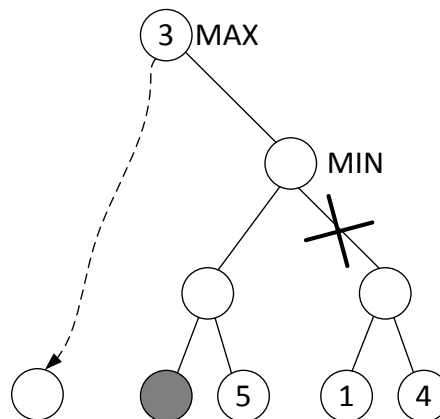
- A. A
- B. B
- C. C
- D. D

14) Indica la rama que se elegiría al aplicar la poda  $\alpha$ - $\beta$  al árbol de juego de la figura.



- A. A
- B. B
- C. C
- D. D**

15) ¿Qué valor provisional debería tener el nodo sombreado para que se produzca el corte indicado?



- A.  $[-\infty, 2]$
- B.  $[-\infty, 3]$
- C.  $[-\infty, 5]$
- D. No es posible el corte**