

Sistemas Inteligentes – Examen Bloque 1, 1 febrero 2024
Test A (1,75 puntos) puntuación: max (0, (aciertos – errores/3)* 1,75/6)

Apellidos:								Nombre:
Grupo:	A	B	C	D	E	F	G	4IA

1) Sea un algoritmo de tipo A ($f(n)=g(n)+h(n)$) donde sabemos que la función $h(n)$ es consistente. Indica la respuesta **INCORRECTA**:

- A. El algoritmo devuelve la solución óptima.
- B. Sea n un nodo padre y n' un nodo hijo. Se cumple siempre que $h(n) > h(n')$.
- C. Se puede generar un nodo $n1$ igual a otro nodo $n2$ tal que $n2$ está en OPEN y $f(n1) < f(n2)$
- D. Se puede generar un nodo $n2$ igual a otro nodo $n1$ tal que $n1$ está en CLOSED y $f(n1) < f(n2)$

2) El hecho CLIPS (transporte bus 1 Plaza-Ayuntamiento Malvarrosa bus 2 Benimaclet San-Marcelino bus 3 Torreíel Hospital-La-Fe) representa el origen y destino de las rutas de tres autobuses. Determina cuál sería el patrón correcto para averiguar el número de autobús que tengo que coger si quiero llegar a San-Marcelino:

- A. (transporte \$?x bus ?numbus San-Marcelino \$?y)
- B. (transporte \$?x bus ?numbus San-Marcelino ?y)
- C. (transporte \$?x bus ?numbus ? San-Marcelino \$?y)
- D. (transporte \$?x bus ?numbus ? San-Marcelino ?y)

3) Sea el siguiente conjunto de hechos y regla en CLIPS

(defacts datos	(defrule R1
(cadena B C A)	(cadena \$?x ?letra \$?y)
(cadena A C A B)	(union \$?lis)
(union))	(test (not (member ?letra \$?lis)))
	=>
	(assert (union ?letra \$?lis)))

Tras la ejecución del SBR, indica la respuesta **CORRECTA**:

- A. Se genera un único hecho 'union' que contiene las tres letras A B y C sin repeticiones
- B. Se generan 6 hechos 'union' cada uno de los cuales contiene las tres letras A B y C en diferente orden
- C. Se generan 6 hechos 'union' cada uno de los cuales contiene las tres letras A B y C en diferente orden y 6 hechos 'union' cada uno de los cuales contiene una combinación de dos letras en diferente orden
- D. Se genera un total de 15 hechos 'union'

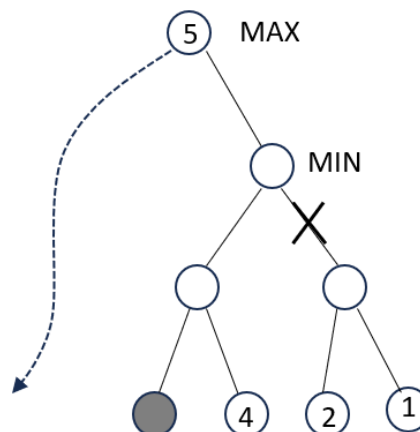
4) Indica la sentencia **CORRECTA**:

- A. Un algoritmo IDA* expande los nodos siguiendo una función de evaluación $f(n)=g(n)+h(n)$.
 - B. Un algoritmo IDA* mantiene un único valor-f para todo el árbol de una iteración.
 - C. Un algoritmo RBFS expande los nodos siguiendo una expansión en profundidad.
 - D. Un algoritmo de Profundidad sin control de nodos repetidos garantiza siempre que se encontrará una solución.
-

5) En un algoritmo Alfa-Beta, indica la sentencia **CORRECTA**:

- A. Los valores α provisionales de un nodo MAX son cotas inferiores del nodo y nunca pueden disminuir al desarrollar la búsqueda.
 - B. El valor β de un nodo MIN no terminal es el mayor valor entre los valores de sus nodos sucesores
 - C. Los valores β provisionales de un nodo MIN son cotas inferiores del nodo y nunca pueden disminuir al desarrollar la búsqueda.
 - D. Ninguna de las anteriores es correcta.
-

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

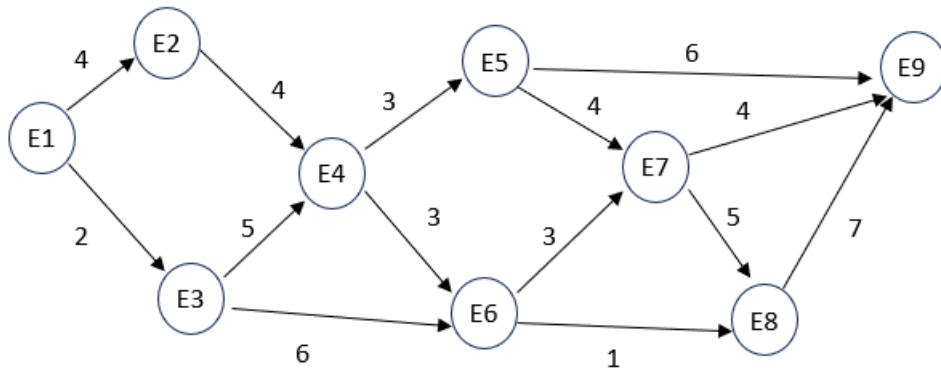


- A. Mayor o igual que 4
- B. Menor o igual que 5
- C. Entre 2 y 6
- D. No es posible el corte

Sistemas Inteligentes – Examen Bloque 1, 1 febrero 2024

Problema: 2 puntos

Se tiene un grafo ponderado que representa un **sistema de rutas de transporte público urbano**. Cada nodo en el grafo representa una estación de transporte público, como una parada de autobús, una estación de metro o tranvía, y cada arista representa las conexiones directas entre estaciones. El peso de cada arista representa el tiempo de viaje entre estaciones. El objetivo de un viajero es ir de la estación E1 a la estación E9.



La siguiente tabla muestra el valor heurístico de cada nodo, una estimación del tiempo que se ha obtenido a partir de datos de usuarios.

n	E1	E2	E3	E4	E5	E6	E7	E8	E9
h(n)	15	11	14	8	6	6	3	5	0

- 1) (0.7 puntos) Muestra el árbol de búsqueda que resultaría de la aplicación de un algoritmo de tipo A ($f(n)=g(n)+h(n)$). Aplicar la versión grafo del algoritmo evitando nodos repetidos. Indica al final el número de nodos generados y expandidos. Indica claramente el valor de la función de evaluación ($f(n)$) en cada nodo y el orden de expansión de los nodos. Ante igualdad de valor de la función de evaluación, expandir antes el nodo numéricamente anterior.

E1, E3, E6, E7, E9. Coste 15

- 2) (0.3 puntos) De acuerdo con los datos del problema y el árbol desarrollado en el apartado anterior: ¿Devuelve el algoritmo la solución óptima? ¿La función heurística es admisible? ¿Y consistente (monótona)? Justifica todas las respuestas.

Encuentra el óptimo pero no es admisible

$h(E3) = 14$ y coste real es 13 \rightarrow E3, E6, E7, E9

Por tanto, tampoco es consistente

- 3) (0.5 puntos) Si aplicamos un algoritmo de búsqueda voraz en este problema, ¿qué solución encontraría? Muestra el árbol que se genera, indica los nodos del camino solución, así como el número total de nodos generados. Ante igualdad de valor de la función de evaluación, expandir antes el nodo numéricamente anterior.

E1, E2, E4, E5, E9 coste 17

- 4) (0.5 puntos) Si aplicamos el algoritmo IDA* a este problema, ¿cuántas iteraciones se necesitarían para encontrar la solución? Justifica la respuesta y muestra el árbol de la primera iteración.

En la segunda iteración el límite es 16 y ya no se supera