



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA

DSiC
DEPARTAMENTO DE SISTEMAS
INFORMÁTICOS Y COMPUTACIÓN

Escuela Técnica Superior de Ingeniería Informática



**Departamento de Sistemas Informáticos y Computación
Escuela Técnica Superior de Ingeniería Informática
Universitat Politècnica de València**

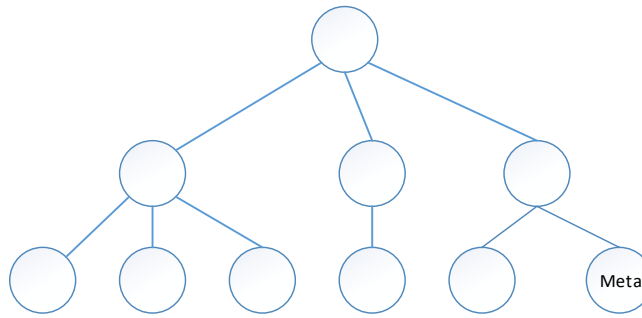
BOLETÍN DE EJERCICIOS SISTEMAS INTELIGENTES

Bloque 1: Búsqueda

Septiembre 2023

CUESTIONES

- 1) Dado el espacio de estados de la figura, si realizáramos una búsqueda en profundidad con *backtracking* (expandiendo primero nodos más a la izquierda). ¿Cuál sería el máximo número de nodos que se almacenarían en memoria simultáneamente (OPEN+CLOSED)?



- A. 7
- B. 5
- C. 10
- D. 4

-
- 2) En el problema del puzzle, con h_1 la heurística descolocadas y h_2 distancias de Manhattan, si tenemos $h_3 = \min(h_1, h_2)$ y $h_4 = \text{abs}(h_1 - h_2)$, ¿cuáles de estas dos heurísticas, h_3 y h_4 , serían admisibles?

- A. Solo h_3
- B. Solo h_4
- C. Ambas
- D. Ninguna de las dos

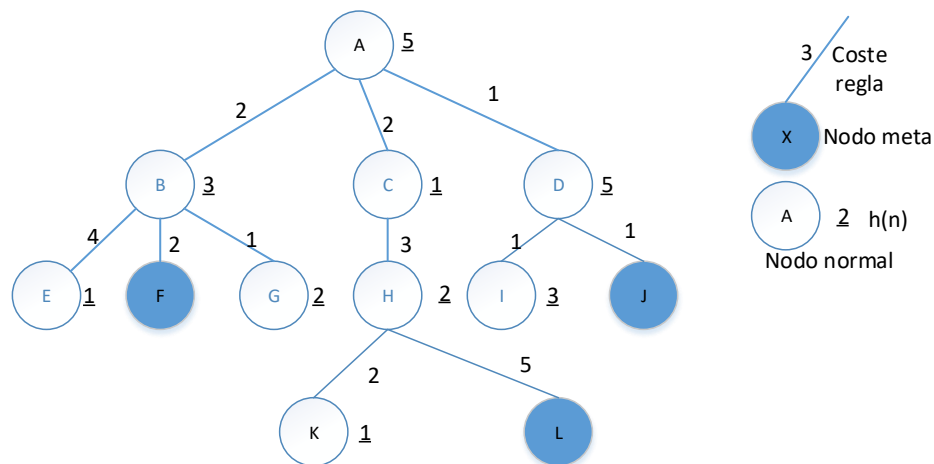
-
- 3) ¿Cuál sería la ordenación (de mayor a menor) en cuanto a número de nodos generados en el peor de los casos de las siguientes estrategias: anchura, profundización iterativa (PI), y profundidad limitada (PL) a nivel 5 por el usuario, para un problema con factor de ramificación $b=10$ y profundidad de la solución $d=5$?

- A. Anchura > PI > PL
- B. Anchura > PL > PI
- C. PI > Anchura > PL
- D. PL > Anchura > PI

-
- 4) Decir cuál de las siguientes afirmaciones es FALSA para una heurística consistente:

- A. $f(n)$ es no decreciente
- B. Garantiza la mejor solución en búsqueda en grafo incluso sin re-expandir
- C. Nunca genera un nodo n_1 igual a otro nodo n_2 ya generado y donde $f(n_1) < f(n_2)$
- D. El valor de $h(n)$ de un nodo puede ser menor que el de su nodo padre.

- 5) Para el espacio de estados de la figura y dada una búsqueda voraz, ¿cuál es el nodo meta que se elegirá en primer lugar como solución?



- A. J
- B. L
- C. I
- D. F

- 6) Dado el espacio de estados de la pregunta 5, si aplicamos un algoritmo de profundización iterativa (expandiendo en primer lugar por la izquierda), ¿cuántos nodos se generarán en total?

- A. 8
- B. 7
- C. 10
- D. 12

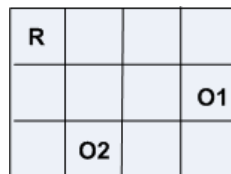
- 7) Dado un problema de búsqueda en el que todos sus operadores tienen el mismo coste, indica cuál de las siguientes afirmaciones es **correcta**:

- 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

- 8) 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**:

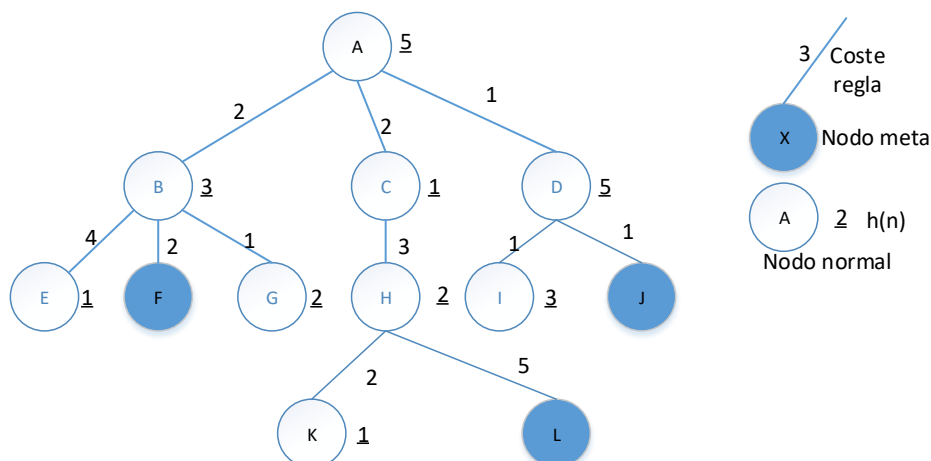
- A. M1, M2 y M3 garantizan que encontrarán la solución óptima independientemente del coste de las acciones
- B. M2 y M3 garantizan que encontrarán la solución óptima independientemente del coste de las acciones
- C. M3 expandirá menos nodos que M2
- D. M4 podría encontrar la solución óptima

- 9) 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**:



- 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

- 10) 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?

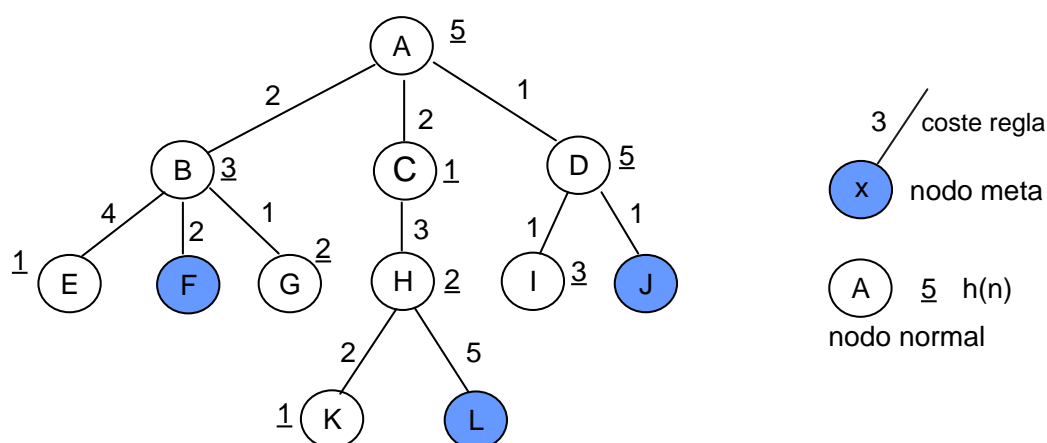


- A. 6
- B. 8
- C. 10
- D. 12

11) Para el espacio de estados de la pregunta 10, 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

12) Para el espacio de estados de la figura y dada una búsqueda de tipo A ($f(n)=g(n)+h(n)$), indica cuál de las siguientes afirmaciones es CORRECTA:

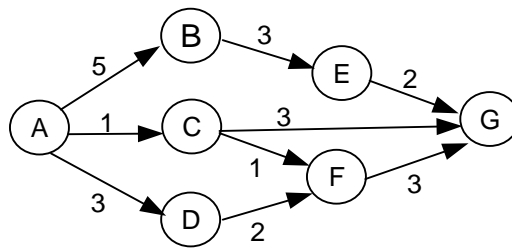


- A. La solución que encuentra la búsqueda de tipo A es el nodo J.
- B. Es necesario generar un total de 10 nodos para encontrar una solución con un algoritmo de tipo A.
- C. La función heurística $h(n)$ no es admisible.
- D. Ninguna de las anteriores.

13) Dados cuatro métodos de búsqueda: M1 aplica un algoritmo anchura, M2 aplica un algoritmo de coste uniforme, M3 aplica un algoritmo en profundidad, M4 aplica un algoritmo de profundización iterativa, si los costes de las acciones son todos iguales indica cuál es la respuesta INCORRECTA:

- A. M1, M2, M3 y M4 encontrarán la solución óptima si existe.
- B. M1 y M2 garantizan que encontrarán la solución óptima.
- C. M4 encontrará la solución óptima.
- D. M1 tendrá un coste espacial mayor que M4.

- 14) Dado el grafo de la figura, donde se señala el coste de los arcos, indica cuál de las siguientes afirmaciones es CORRECTA:



- A. La aplicación de un algoritmo de Búsqueda en Anchura encontrará el camino A-D-F-G
- B. La aplicación de un algoritmo de Coste Uniforme devolverá una solución de coste 5
- C. Un algoritmo de Búsqueda en Anchura y Coste Uniforme encontrarán la misma solución
- D. Ninguna de las anteriores.

-
- 15) Si se aplica un algoritmo de Profundización Iterativa sobre el grafo de la figura 10, ¿cuántas iteraciones serían necesarias hasta encontrar la solución?:

- A. 2
- B. 3
- C. 4
- D. Ninguna de las anteriores.

-
- 16) Sea una búsqueda de tipo A ($f(n)=g(n)+h(n)$) donde la función $h(n)$ es admisible y consistente. El algoritmo devuelve una solución desde el nodo inicial A al nodo objetivo G que atraviesa un nodo n1. Indica cuál de las siguientes afirmaciones es INCORRECTA:

- A. $f(A) \leq f(n1) \leq f(G)$
- B. $f(G)=h^*(A)$
- C. $h^*(A) < h(n1)$
- D. $f(G)=g(G)$.

-
- 17) Supongamos dos funciones de evaluación para un mismo problema $f1(n)=g(n)+h1(n)$ y $f2(n)=g(n)+h2(n)$ tales que $\forall n \ h1(n) \leq h2(n) \leq h^*(n)$. Dado un algoritmo de tipo A que utilice estas funciones, indica cuál de las siguientes afirmaciones es CORRECTA:

- A. Solo una de las dos funciones encontrará la solución óptima.
- B. El algoritmo que utilice $f1(n)$ expandirá menos nodos que el que utilice $f2(n)$
- C. El algoritmo que utilice $f1(n)$ expandirá más nodos que el que utilice $f2(n)$
- D. Ninguna de las dos funciones desarrollará una búsqueda completa.

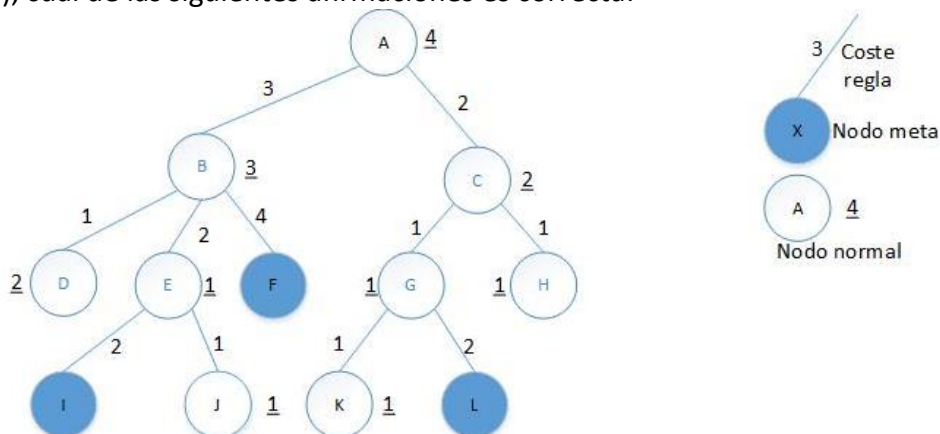
18) Dados 3 algoritmos de búsqueda, M1 implementa una búsqueda de coste uniforme, M2 es algoritmo de tipo A con una heurística admisible y M3 implementa una búsqueda voraz, ¿cuál de las siguientes afirmaciones es INCORRECTA?:

- A. M1 y M2 encontrarán la solución de coste óptimo
- B. Se garantiza que M3 encontrará la solución más rápidamente que M1 y M2
- C. No se puede garantizar que M3 encontrará la solución óptima
- D. M1 expandirá más nodos que M2

19) Sean dos funciones de evaluación $f_1(n)=g(n)+h_1(n)$ y $f_2(n)=g(n)+h_2(n)$, tales que $h_1(n)$ es admisible y $h_2(n)$ no lo es, indica la respuesta correcta:

- A. El uso de ambas funciones en un algoritmo de tipo A garantiza encontrar la solución óptima
- B. Se garantiza que $f_2(n)$ generará un menor espacio de búsqueda que $f_1(n)$
- C. Sólo si $h_1(n)$ es una heurística consistente, $f_1(n)$ generará un menor espacio de búsqueda que $f_2(n)$
- D. Existe algún nodo n para el que $h_2(n) > h^*(n)$

20) Para el espacio de estados de la figura y dada una búsqueda en anchura (expandiendo por la izquierda), cuál de las siguientes afirmaciones es correcta:



- A. Devuelve el nodo I
- B. Genera 8 nodos
- C. Expande 4 nodos
- D. Ninguna de las tres anteriores

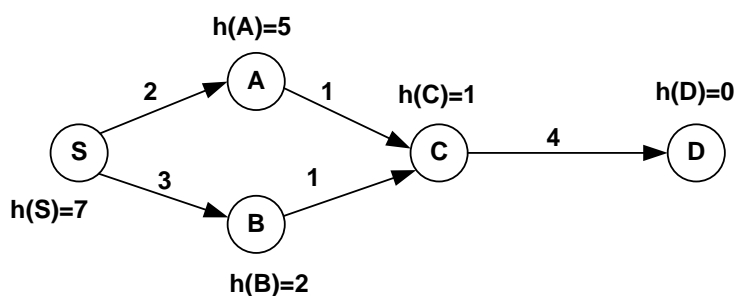
21) Para el árbol de estados de la pregunta anterior, y suponiendo una búsqueda de tipo A ($f(n)=g(n)+h(n)$), ¿cuál de las siguientes afirmaciones es FALSA?:

- A. Es admisible
- B. Devuelve el nodo L
- C. Expande 3 nodos
- D. Genera 7 nodos

22) Asumiendo que todos los nodos de un espacio de búsqueda tienen más de un hijo ¿en cuál de las siguientes estrategias el orden de generación de los nodos nunca puede ser el mismo que el orden de expansión de los mismos?

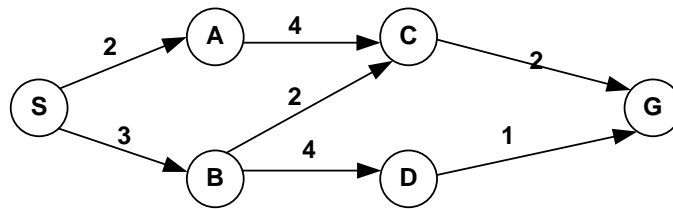
- A. Anchura
- B. Coste uniforme
- C. Profundidad
- D. Búsqueda voraz

23) Dado el espacio de estados de la figura, donde S es el estado inicial, D el nodo meta, y se indican los costes de cada arco y la estimación $h(n)$ en cada nodo, marca la opción correcta:



- A. La aplicación de un algoritmo A (tree search, con control de nodos repetidos en OPEN) no obtendrá la senda óptima.
- B. La aplicación de un algoritmo A (graph search, con control de nodos repetidos en CLOSED, tal que un nuevo nodo se descarta si ya existe en CLOSED) obtendrá la senda óptima.
- C. La respuesta A no es cierta, debido a que $h(n)$ no es admisible
- D. La respuesta B no es cierta, debido a que $h(n)$ no es consistente

24) Dado el espacio de estados de la figura, el número de nodos que genera una búsqueda (Tree-Search) de coste uniforme donde, a igualdad de $f(n)$, se expande el nodo alfabéticamente menor es:



- A. Mayor que si se realizase una búsqueda en anchura
- B. Menor que si se realizase una búsqueda en anchura
- C. Menor que si se realizase una búsqueda en profundidad
- D. Ninguna de las anteriores es cierta

25) ¿Cuál de las siguientes afirmaciones es CORRECTA para una heurística consistente $h(n)$ en una búsqueda de tipo $f(n)=g(n)+h(n)$?

- A. No devuelve la solución óptima
- B. El valor heurístico del padre puede ser igual al del hijo
- C. Nunca genera un nodo 'n1' igual a otro nodo 'n2' ya generado y donde $f(n1) < f(n2)$
- D. Nunca genera un nodo que ya esté en la lista CLOSED

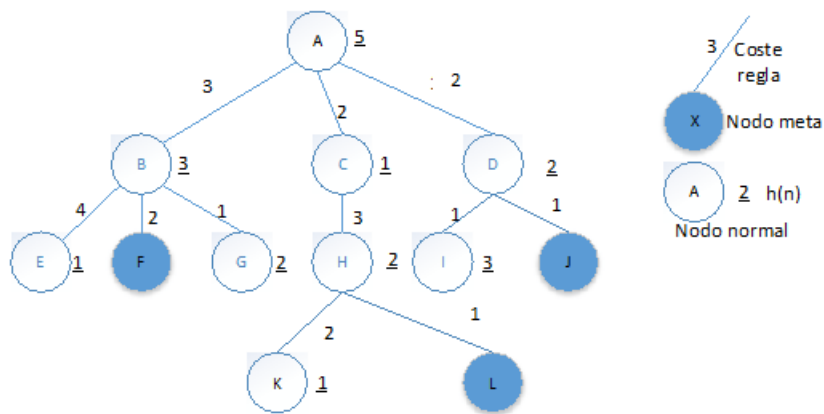
26) Sea una búsqueda de tipo $f(n)=g(n)+h(n)$ con $h(n)$ admisible, dos nodos solución $G1$ y $G2$ donde $G1$ es una solución óptima y $G2$ no lo es y un nodo $n1$ que pertenece al camino solución de $G1$. Indica cuál es la afirmación INCORRECTA:

- A. $g(G1) \leq f(G2)$
- B. $f(n1) \leq g(G2)$
- C. $h^*(n1)+g(n1)=f(G1)$
- D. Ninguna de las anteriores

27) Respecto al número de nodos generados en el peor de los casos para una búsqueda en Profundización iterativa que encuentra la solución en el nivel d y una búsqueda en Profundidad limitada $m=d$ para un mismo problema, ¿cuál de las siguientes afirmaciones es CORRECTA?

- A. Profundidad limitada generará más nodos que Profundización iterativa
- B. Profundización iterativa generará más nodos que Profundidad limitada
- C. Las dos búsquedas generarán el mismo número de nodos
- D. Ninguna de las anteriores

28) 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, incluyendo el nodo raíz, para encontrar la solución?



- A. 7
- B. 8
- C. 10
- D. 12

29) Se desea realizar una búsqueda A* en CLIPS. Para ello, las reglas no deben contener la instrucción **retract** en la parte derecha porque:

- A. Al borrar los hechos no podemos calcular el valor de $g(n)$ necesario para una búsqueda A*.
- B. No permitiría explorar caminos alternativos al elegido en primer lugar
- C. No permitiría encontrar la solución óptima
- D. Ninguna de las anteriores

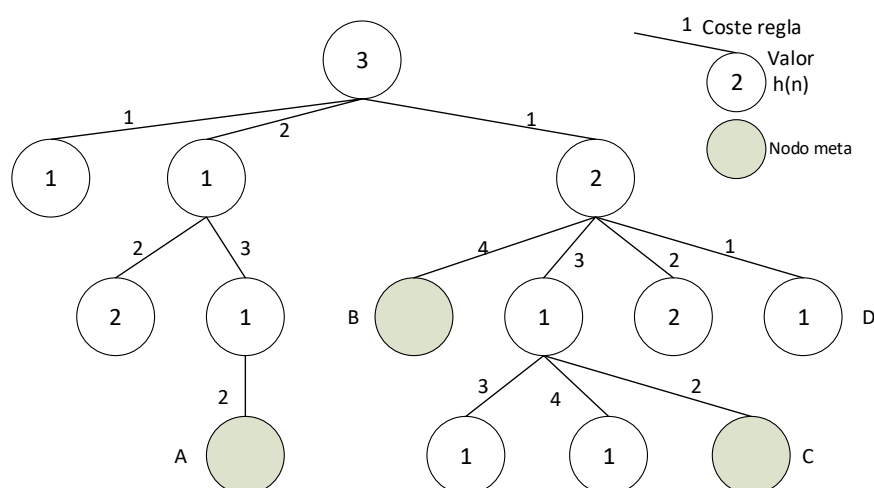
30) Dado un algoritmo de búsqueda de tipo A, ($f(n)=g(n)+h(n)$), señala la afirmación **CORRECTA**:

- A. Si $h(n)$ es consistente (y admisible), expandirá siempre menos nodos que una búsqueda no informada
- B. Con $h(n)$ consistente (y admisible), expandirá siempre menos nodos que no siendo consistente.
- C. Encuentran siempre la misma solución, independientemente de si $h(n)$ es admisible o no.
- D. Ninguna de las anteriores

31) Sea un problema de búsqueda en el que los costes de los operadores son distintos. La aplicación de un algoritmo GRAPH-SEARCH de Coste Uniforme con control de nodos repetidos devuelve una solución de coste 'c' en un nivel de profundidad 'd'. Indica cuál de las siguientes afirmaciones es CIERTA.

- A. No es necesario especificar un nivel de profundidad máxima para el espacio de búsqueda con el fin de evitar que el algoritmo entre en un bucle infinito.
- B. Una solución con coste c' tal que $c' > c$ solo se encontrará en un nivel d' tal que $d' > d$.
- C. Un algoritmo de Profundización Iterativa sobre el mismo problema devolverá siempre la solución óptima.
- D. Un algoritmo de Anchura sobre el mismo problema devolverá siempre la solución óptima.

32) Si se aplica un algoritmo de tipo A en el espacio de estados de la figura siguiente, ¿qué nodo meta se elegirá en primer lugar como solución?



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

33) Dado el espacio de búsqueda de la figura anterior, una búsqueda A que utilice dichos valores, sería:

- A. Admisible y consistente
B. Admisible y no consistente
C. No admisible pero si consistente
D. Ni admisible ni consistente

34) Sea un algoritmo A con una heurística $h(n)$. El número de nodos generados para la obtención de la solución ante un estado inicial concreto (indicar la respuesta CORRECTA):

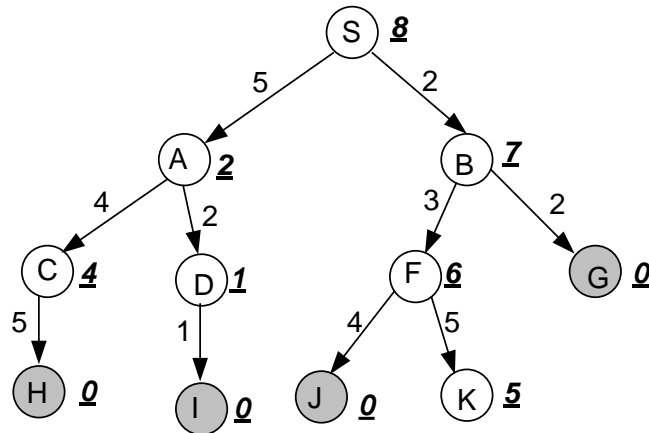
- A. Dependerá del factor efectivo de ramaje de la heurística $h(n)$ y de la profundidad de la solución óptima.
- B. Dependerá del coste de aplicación de cada regla.
- C. Si $h(n)$ es admisible, nunca será mayor que el número de nodos generados con una heurística $h'(n)$ tal que $h'(n) \leq h(n)$
- D. Ninguna de las anteriores

35) En la aplicación de una búsqueda A con método GRAPH-SEARCH, se ha encontrado el camino óptimo hasta cada nodo expandido si:

- A. La función heurística es admisible.
B. Se efectúa un control de nodos repetidos en la lista OPEN.

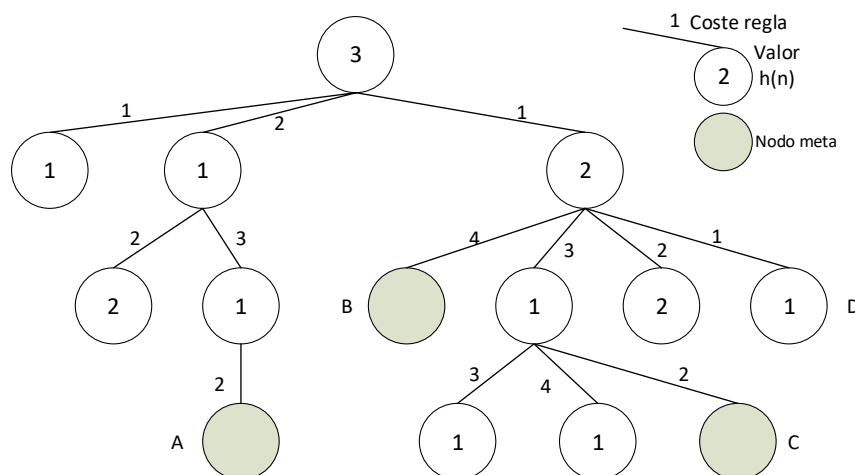
- C. La función heurística es consistente.
- D. Ninguna de las anteriores.

36) Dado el siguiente espacio de búsqueda, donde se indica el coste en las ramas y la estimación $h(n)$ a la derecha de cada nodo, indica la respuesta CORRECTA.



- A. Un algoritmo A generará menos nodos que una búsqueda de coste uniforme.
- B. Un algoritmo A obtendrá la senda óptima.
- C. Una expansión en anchura, empezando por la izquierda, generará igual o menor número de nodos que coste uniforme.
- D. Ninguna de las anteriores es cierta.

37) Si se aplica una búsqueda voraz en el espacio de búsqueda de la figura, ¿qué nodo meta se elegirá en primer lugar como solución y cuántos nodos se generarán para encontrar dicha solución?



- A. Nodo A y se generan 7 nodos
- B. Nodo B y se generan 8 nodos
- C. Nodo B y se generan 11 nodos
- D. Nodo C y se generan 14 nodos

38) Dado el espacio de búsqueda de la figura anterior, indica la respuesta INCORRECTA:

- A. La función $h(n)$ es admisible
 - B. La función $h(n)$ es consistente
 - C. Un algoritmo en anchura encontraría la misma solución que un algoritmo de tipo A
 - D. Un algoritmo en profundidad encontraría la misma solución que un algoritmo voraz
-

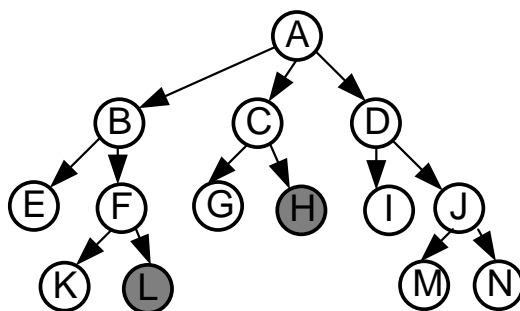
39) Sean tres niveles de un árbol de búsqueda para un problema, d_1 , d_2 y d_3 , donde $d_1 < d_2 < d_3$, tal que una solución se encuentra en el nivel d_1 , otra solución en el nivel d_2 y otra solución en el nivel d_3 (solo hay una solución en cada uno de los niveles). Indica la afirmación CORRECTA:

- A. La complejidad temporal de un algoritmo de Anchura es $O(b^{d_2})$ y la de un algoritmo de Profundización Iterativa es $O(b^{d_1})$
 - B. La complejidad temporal de un algoritmo limitado en Profundidad, con máxima profundidad $m=d_1$, es $O(b^{d_1+1})$
 - C. Asumiendo que se selecciona máxima profundidad $m=d_3$, un algoritmo limitado en Profundidad siempre encontrará antes la solución del nivel d_1 o d_2 .
 - D. Asumiendo que se selecciona máxima profundidad $m=d_1$, la complejidad temporal de un algoritmo limitado en profundidad y un algoritmo de profundización iterativa es $O(b^{d_1})$
-

40) Sea la aplicación de un algoritmo A^* para la resolución de un problema y sea G el nodo solución encontrado. Indica la sentencia que es FALSA:

- A. Si $h(n)$ es consistente entonces $\forall n_1, n_2$ tal que n_2 es un hijo de n_1 se cumple siempre $h(n_2) \geq h(n_1)$
 - B. $\forall n_1, n_2$, tal que n_1 y n_2 son nodos del camino solución a G , se cumple siempre $g(n_1) + h^*(n_1) = g(n_2) + h^*(n_2)$
 - C. $\forall n$, tal que n es un nodo del camino solución a G , se cumple siempre $f(n) \leq g(G)$
 - D. Se cumple siempre que $f(G) = g(G)$
-

41) 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?

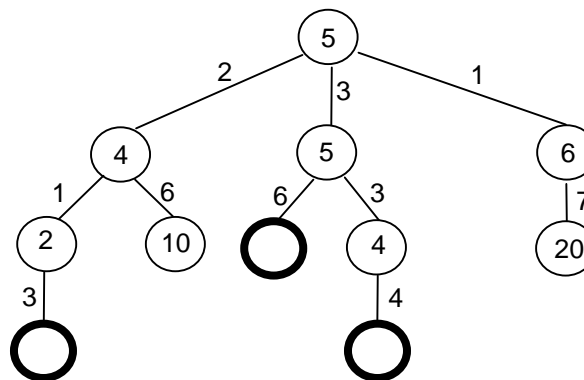


- A. ABCDEFKL y encuentra nodo meta L
- B. ABCDEFGHIJKL y encuentra nodo meta H
- C. AABCDABCDEFHG y encuentra nodo meta H
- D. ABCDEFGH y encuentra nodo meta H

42) 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

43) 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

44) Sea un problema de búsqueda donde los operadores tiene distinto coste. Existe un nodo solución, G_1 , en el nivel d_1 del árbol de búsqueda y un nodo solución, G_2 , que es óptimo y se encuentra en un nivel d_2 , tal que $d_2 > d_1$. Indica la respuesta **CORRECTA**:

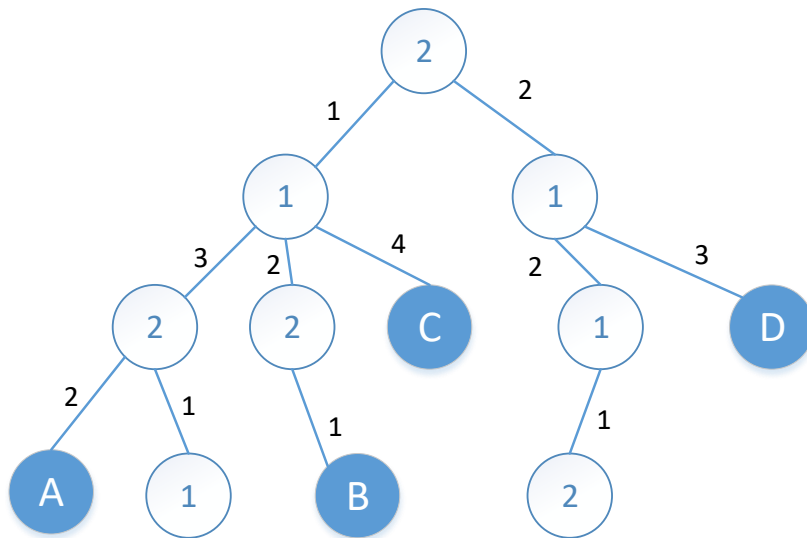
- A. La complejidad temporal de anchura respecto al número de nodos generados es $O(b^{d_1})$
- B. Una estrategia en profundidad nunca devolverá la solución G_1
- C. Una estrategia por profundidad iterativa nunca devolverá la solución G_1 .
- D. Una estrategia de coste uniforme devolverá siempre la solución G_2

45) En el árbol de búsqueda que se genera con un algoritmo de tipo A* tenemos dos nodos, n_1 y n_2 , que se corresponden con dos estados repetidos. Se sabe, además, que n_1 es un nodo que

se encuentra en el camino óptimo a un nodo solución, G , mientras que n_2 no está en el camino óptimo a G . Indica la respuesta **INCORRECTA**:

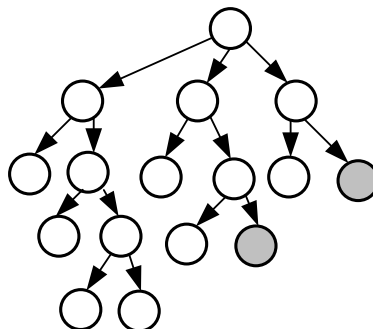
- A. Se cumple siempre $f(n_1) \leq f(G)$
 B. Se cumple siempre $g(n_1) < g(n_2)$
 C. Se cumple siempre $h(n_1) < h(n_2)$
 D. Se cumple siempre $h(n_2) \leq h^*(n_2)$

46) 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.



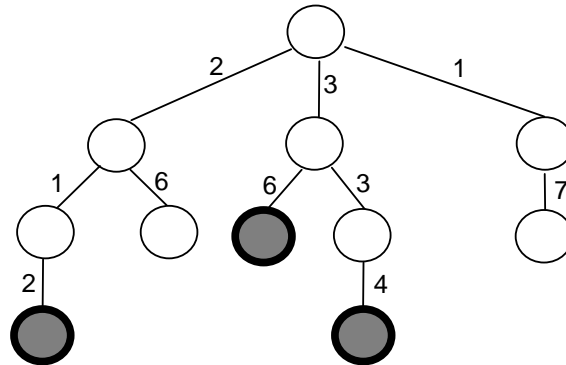
- A. A
B. B
C. C
D. D

47) Considerando el siguiente árbol de búsqueda, ¿cuántos nodos como máximo se almacenan en memoria, aplicando un procedimiento de búsqueda en profundidad iterativa? (Asúmase que a igual profundidad se elige el nodo más a la izquierda)



- A. 6
- B. 3
- C. 4
- D. 5

48) Dado el árbol de la figura, donde los nodos sombreados son nodos objetivo, indica la respuesta **CORRECTA**:

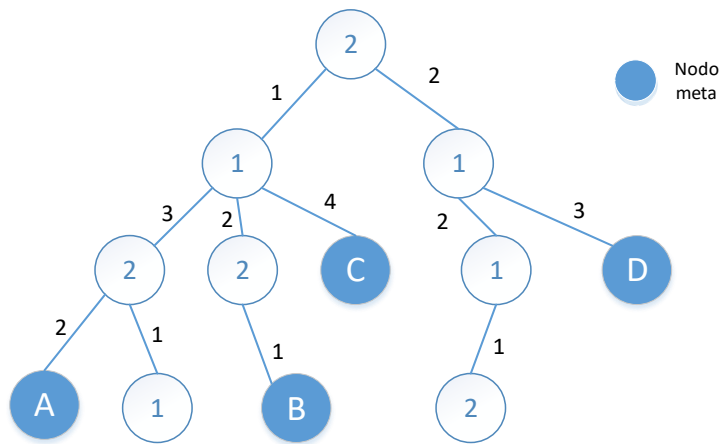


- A. La aplicación de una estrategia en anchura devuelve la misma solución que coste uniforme.
- B. La aplicación de una estrategia en anchura devuelve la misma solución que una estrategia de profundidad a nivel máximo de profundidad $m=2$.
- C. La aplicación de una estrategia en anchura devuelve la misma solución que una estrategia de profundidad a nivel máximo de profundidad $m=3$.
- D. La aplicación de una estrategia por coste uniforme devuelve la misma solución que profundización iterativa.

49) La aplicación de una heurística admisible, h_1 , a un problema devuelve un nodo solución G_1 y el número de nodos que expande es n_1 . La aplicación de una heurística admisible, h_2 , al mismo problema, donde h_2 domina a h_1 , devuelve un nodo solución G_2 y expande un número de nodos igual a n_2 . Indica la respuesta **CORRECTA**:

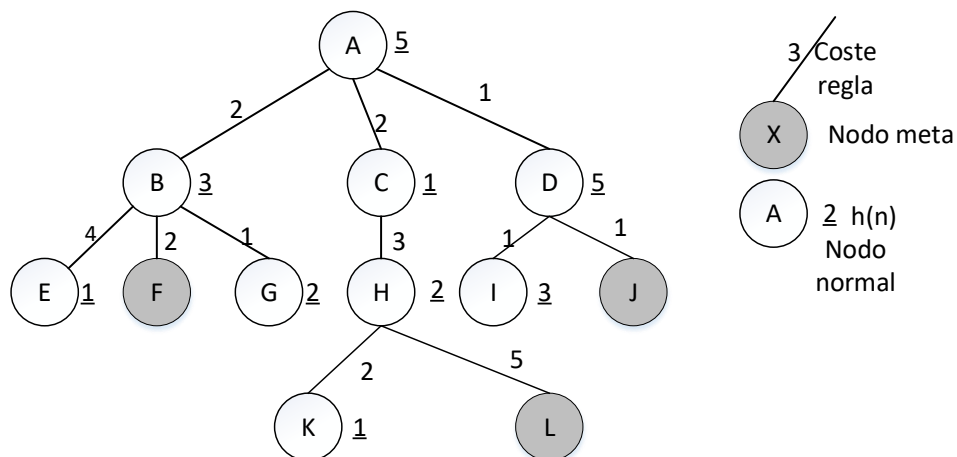
- A. Se cumple que $g(G_1) < g(G_2)$
- B. Se cumple $h_1(G_1) < h_2(G_2)$
- C. Se cumple que $n_1 < n_2$
- D. Ninguna de las respuestas anteriores es correcta.

50) Dado el árbol de la siguiente figura, ¿cuántos nodos se generarían (incluyendo nodo inicial) si se aplicara un algoritmo A? (en caso de igualdad de $f(n)$, se expande el nodo más a la izquierda).



- A. 6
- B. 8
- C. 9
- D. 10

51) Para el espacio de estados de la figura y dada una búsqueda voraz, ¿cuál es el nodo meta que se elegirá en primer lugar como solución? (en caso de igualdad de $f(n)$ se expande el nodo más a la izquierda)

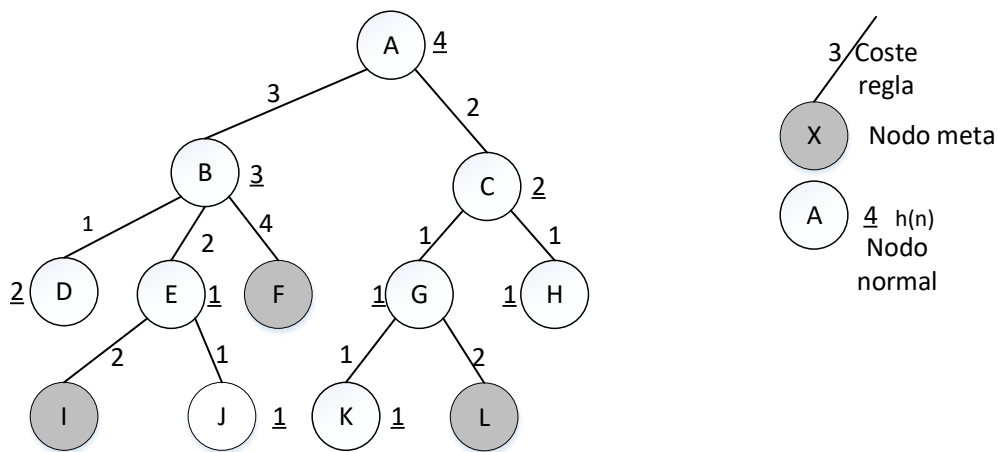


- A. L
- B. J
- C. I
- D. F

52) Sean dos funciones de evaluación $f_1(n)=g(n)+h_1(n)$ y $f_2(n)=g(n)+h_2(n)$, tales que $h_1(n)$ es admisible y $h_2(n)$ no lo es. Indica la respuesta **CORRECTA**:

- A. El uso de ambas funciones en un algoritmo de tipo A garantiza encontrar la solución óptima
- B. Sólo si $h_1(n)$ es una heurística consistente, $f_1(n)$ generará un menor espacio de búsqueda que $f_2(n)$
- C. Existe algún nodo n para el que $h_2(n) > h^*(n)$
- D. Se garantiza que $f_2(n)$ generará un menor espacio de búsqueda que $f_1(n)$

53) Para el espacio de estados de la figura y dada una búsqueda tipo A ($f(n)=g(n)+h(n)$), ¿cuál de las siguientes afirmaciones es **FALSA**:

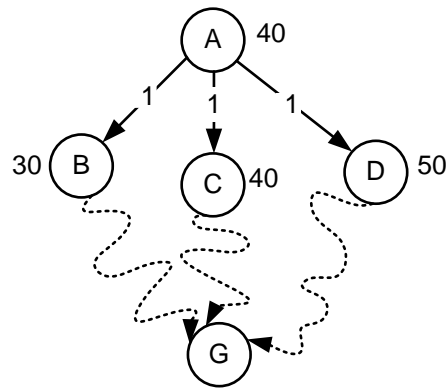


- A. Es admisible
- B. Expande 3 nodos
- C. Genera 7 nodos
- D. Devuelve el nodo L

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

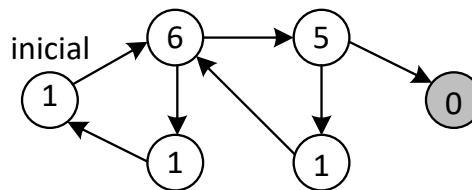
- A. Un algoritmo de búsqueda con una heurística admisible devolverá la solución más corta
- B. La estrategia en anchura devolverá la solución más corta pero no la solución de menor coste
- C. La estrategia de coste uniforme devolverá la solución de menor coste pero no la solución más corta
- D. Una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

55) Supongamos el espacio de búsqueda de la figura en el que se quiere encontrar una senda solución del estado inicial A al único estado meta G. Los arcos tienen un coste unitario y en cada nodo se indica el valor de $h(n)$, que sabemos que es admisible. Aplicando un algoritmo de tipo A ($f(n)=g(n)+h(n)$), con re-expansión, se cumple que:



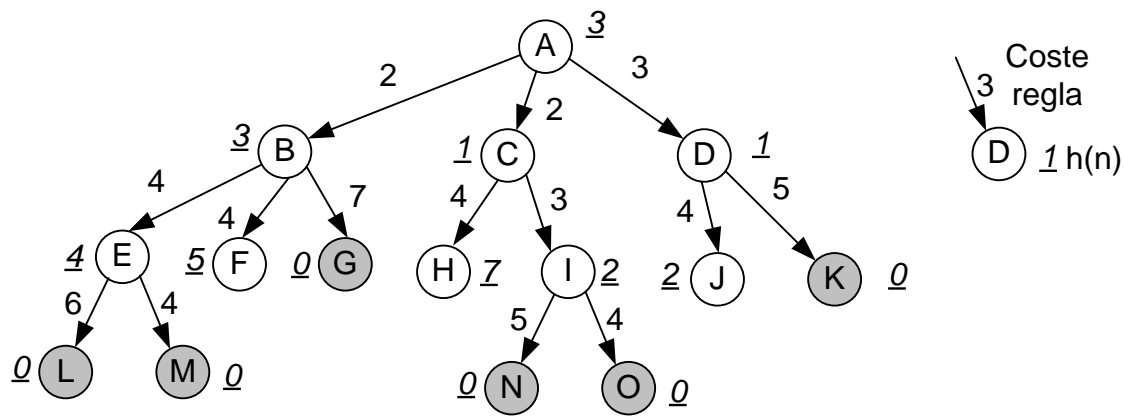
- A. Si obtenemos como solución un camino de A a G de coste 40, será una senda solución óptima.
- B. Si obtenemos como como solución un camino de A a G de coste 51, no será una senda solución óptima.
- C. La aplicación del algoritmo A no obtendrá la solución óptima.
- D. Con los datos de la figura, $h(n)$ es consistente.

56) Supongamos que se aplica un algoritmo de tipo A ($f(n)=g(n)+h(n)$) en el siguiente espacio de estados, donde el nodo sombreado es el nodo meta, todos los arcos tienen coste 1, y en los nodos se indica el valor de la función $h(n)$. Indica la respuesta **CORRECTA**:



- A. No se puede aplicar un algoritmo A es el espacio de estados anterior, ya que $h(n)$ no es consistente.
- B. El algoritmo no terminará, pues entrará en un ciclo infinito.
- C. El algoritmo encontrará la solución, pero no tiene que ser necesariamente la óptima.
- D. El algoritmo encontrará la solución óptima.

57) Para el espacio de estados de la figura, donde los nodos sombreados son nodos meta, y dada una búsqueda de tipo A ($f(n)=g(n)+h(n)$), ¿cuántos nodos es necesario generar, incluyendo el nodo A, para encontrar la solución?



- A. 10.
- B. 11.
- C. 13.
- D. 15.

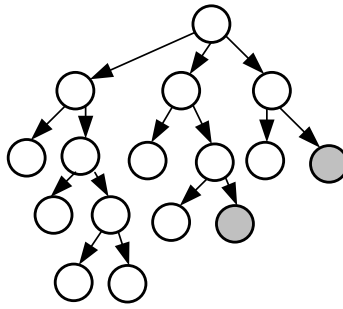
58) Sea el espacio de estados de la figura anterior. Asumiendo que dados dos o más nodos con el mismo valor $f(n)$ se escoge un nodo aleatoriamente, indica la respuesta **CORRECTA**:

- A. La aplicación de una estrategia en Profundidad con $m=3$ (máximo nivel de profundidad) encontrará como solución cualquier nodo meta.
- B. La aplicación de una estrategia por Coste Uniforme encontrará como solución los nodos G o N.
- C. La aplicación de una estrategia por Profundización Iterativa encontrará como solución los nodos L, M, N o O.
- D. Ninguna de las repuestas anteriores es correcta.

59) Supongamos un tablero de 4x4 donde un robot está situado en la casilla inferior izquierda $(x,y)=(1,1)$ y desea llegar a la casilla superior derecha $(x,y)=(4,4)$. El robot puede realizar movimientos horizontal, vertical o diagonalmente, y todos ellos tienen un coste unitario. Indica la respuesta **INCORRECTA**:

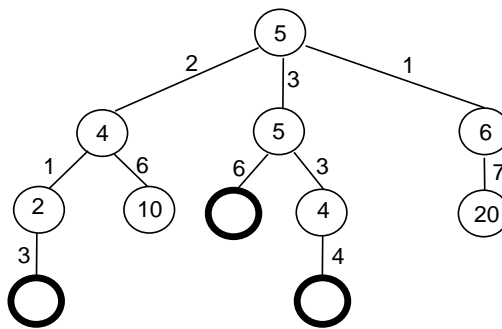
- A. El coste de la solución óptima a este problema es 3.
- B. La distancia de Manhattan del robot a la casilla meta es una heurística admisible.
- C. La aplicación de un algoritmo en anchura devolverá la solución óptima.
- D. Al expandir el nodo $(x,y)=(2,3)$ se generarán 8 nodos hijo.

60) Considerando el siguiente árbol de búsqueda, ¿cuántos nodos como máximo se almacenan en memoria, aplicando un procedimiento de búsqueda en profundidad iterativa? (asúmase que a igual profundidad se elige el nodo más a la izquierda):



- A. 6
- B. 8
- C. 10
- D. Ninguna de las respuestas anteriores es correcta.

61) 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

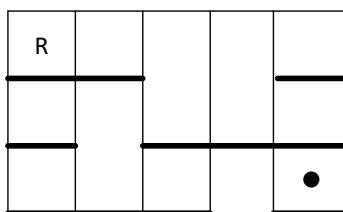
62) Sean dos funciones de evaluación $f_1(n)=g(n)+h_1(n)$ y $f_2(n)=g(n)+h_2(n)$, tales que $h_1(n)$ es admisible y $h_2(n)$ no lo es, indica la respuesta CORRECTA:

- A. El uso de ambas funciones en un algoritmo de tipo A garantiza, en cada caso, encontrar la solución óptima
- B. Se garantiza que $f_2(n)$ generará un menor espacio de búsqueda que $f_1(n)$
- C. Sólo si $h_1(n)$ es una heurística consistente, $f_1(n)$ generará un menor espacio de búsqueda que $f_2(n)$
- D. Existe algún nodo n para el que $h_2(n) > h^*(n)$

63) En una búsqueda GRAPH-SEARCH que aplica un algoritmo de tipo A ($f(n)=g(n)+h(n)$), se tiene un nodo n en la lista CLOSED y un nodo n' en la lista OPEN tal que $n'=n$. Indica la respuesta CORRECTA:

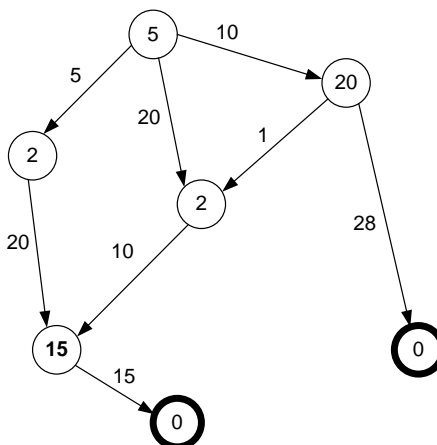
- A. Si la heurística es admisible, se cumple siempre $h(n) < h(n')$.
- B. Si la heurística es consistente, se cumple siempre $g(n) \leq g(n')$.
- C. Independientemente de si la heurística es consistente o no, se cumple siempre $f(n) \leq f(n')$.
- D. Ninguna de las respuestas anteriores es correcta.

64) La figura muestra un grid en el que hay un robot en la posición $(x,y)=(1,3)$ y quiere llegar a la casilla marcada con un punto en posición $(x,y)=(5,1)$. Algunas casillas del grid no tienen suelo, es decir, hay huecos, y el robot nunca puede posicionarse en ellas. El robot puede realizar movimientos horizontales (derecha, izquierda), verticales (arriba, abajo) y diagonales (por ejemplo, desde la casilla $(1,3)$ el robot solo puede moverse a la derecha y abajo). El coste de cada movimiento es 1. Indica la respuesta **CORRECTA**:



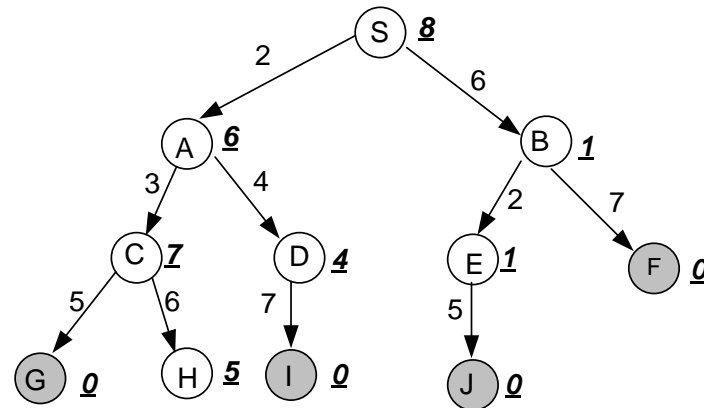
- A. La solución óptima para este problema es 6.
- B. La heurística 'Distancias de Manhattan' aplicada a este problema es una heurística admisible.
- C. La expansión del nodo $(x,y)=(5,2)$ genera 4 nodos hijo.
- D. Ninguna de las respuestas anteriores es correcta.

65) Dado el siguiente espacio de búsqueda, donde se indica el coste en las ramas, la estimación $h(n)$ dentro de cada nodo, y el nodo inicial es el marcado como $h(n)=5$, indica la respuesta **CORRECTA**.



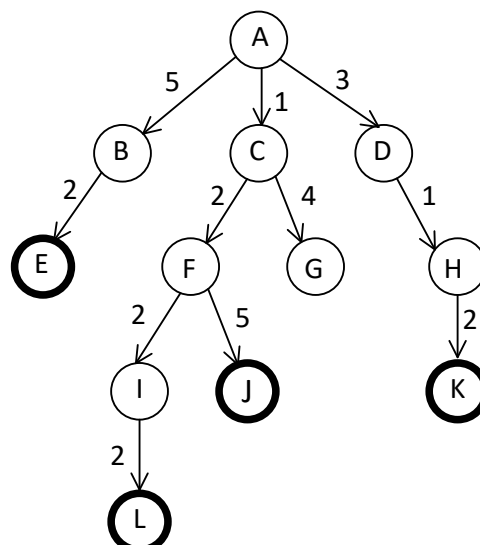
- A. $h(n)$ no es admisible.
- B. Un Algoritmo Tree-Search, de tipo A, con control de repetidos en lista OPEN no obtiene la solución óptima.
- C. Un Algoritmo Graph-Search, de tipo A, con control de repetidos en OPEN y sin re-expansión de nodos repetidos en CLOSED no obtiene la solución óptima
- D. Una búsqueda en anchura obtiene la solución óptima.

66) Dado el siguiente espacio de búsqueda, donde se indica el coste en las ramas y la estimación $h(n)$ a la derecha de cada nodo, indica la respuesta **CORRECTA**.



- A. Una expansión en profundidad generará tantos o más nodos que una expansión en coste uniforme
- B. La $h(n)$ es admisible
- C. El algoritmo A obtendrá la senda óptima.
- D. Ninguna de las anteriores es cierta

67) La figura muestra el espacio que se genera con la aplicación de un algoritmo de búsqueda. Los nodos marcados en negrita son nodos solución, y el algoritmo encuentra el **nodo K** como primera solución. Los valores de las aristas representan el coste de los operadores. ¿Qué algoritmo de búsqueda no informada se ha aplicado?

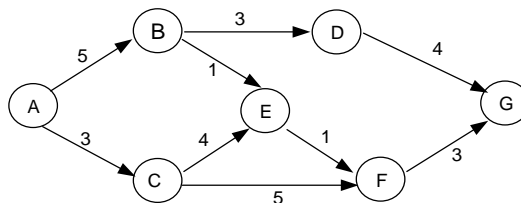


- A. Búsqueda en anchura.
- B. Búsqueda por coste uniforme.
- C. Búsqueda en profundidad con nivel máximo de profundidad $m=3$.
- D. Búsqueda por profundización iterativa.

68) Sea el árbol de búsqueda de la pregunta anterior, y una función heurística $h(n)$ que sabemos que es admisible. Indica la respuesta **INCORRECTA**:

- A. $h(J)=h(E)$
- B. $g(L) \leq g(D)+h(D)$
- C. $h(C) \leq 6$
- D. $h^*(A)=6$

69) Dado el grafo de la figura, donde el nodo G es el nodo Meta, indica la respuesta **CORRECTA** (ante dos nodos con el mismo valor de $f(n)$ se expande antes el nodo alfabéticamente anterior):



- A. Una búsqueda en profundidad con máximo nivel de profundidad $m=4$ encuentra la solución más corta
- B. Una búsqueda en anchura encuentra la solución de coste óptimo
- C. Una búsqueda de coste uniforme encuentra la solución más corta
- D. Una búsqueda por profundización iterativa encuentra la solución en 5 iteraciones

70) Sean tres funciones $f_1(n)=g(n)+h_1(n)$, $f_2(n)=g(n)+h_2(n)$ y $f_3(n)=g(n)+h_3(n)$ tal que se sabe que $h_2(n)$ es admisible, $h_3(n)$ no lo es y $\forall n \ h_1(n) \leq h_2(n)$. Asumiendo que G_1 es el nodo solución que devuelve la búsqueda con f_1 , G_2 el que devuelve la búsqueda con f_2 y G_3 el que devuelve la búsqueda con f_3 , indica la respuesta **CORRECTA**:

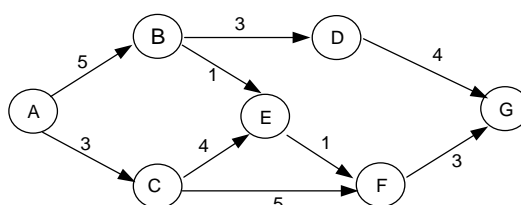
- A. Se cumple $g(G_1) < f_2(G_2)$
- B. Se cumple $f_1(G_1) < f_2(G_2)$
- C. Se cumple $f_2(G_2) \leq f_3(G_3)$
- D. Ninguna de las respuestas anteriores es correcta

71) Sea un grid de 4×4 donde la casilla inferior izquierda es $(x,y)=(1,1)$ y la casilla superior derecha es la $(x,y)=(4,4)$. Hay un robot en la casilla $(1,3)$, una lata en la casilla $(1,2)$ y un contenedor triturador en la casilla $(4,1)$. El robot puede moverse a una casilla adyacente en cuatro direcciones (arriba, abajo, derecha e izquierda) y puede empujar la lata en cualquiera de las cuatro direcciones, respetando los límites del grid. Para empujar una lata, el robot debe

situarse en una casilla adyacente a la lata. Como resultado de una acción de empujar, la posición del robot y la posición de la lata se desplazan una casilla en la dirección del empuje, respetando los límites del grid. El objetivo del problema es empujar la lata al contenedor triturador. Indica la respuesta **INCORRECTA** aplicada a este estado del problema:

- A. Una búsqueda en profundidad (limitada a máximo nivel $m=5$) expandiría el mismo número de nodos que los nodos expandidos en los 5 primeros niveles de una búsqueda en anchura
- B. La solución óptima se encuentra en el nivel 5
- C. Hay dos acciones de movimiento del robot aplicables en el estado
- D. Hay una acción de empujar lata aplicable en el estado

72) Supongamos la aplicación de una función heurística $h(n)$ al grafo que se muestra abajo. Se sabe que $h(A)=8$ y que para todo nodo sucesor n' de un nodo n , $h(n')=h(n)-2$. En el caso de que n' tenga i nodos padre entonces $h(n')=\min(h(n_1), \dots, h(n_i))-2$. Indica la respuesta **CORRECTA**:

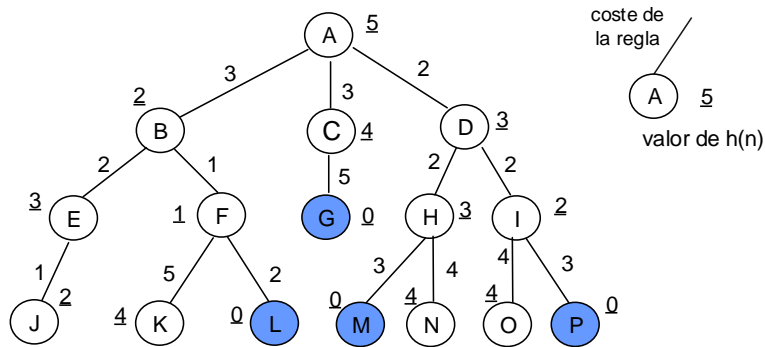


- A. La heurística es admisible
- B. La aplicación de una estrategia de tipo A ($f(n)=g(n)+h(n)$) no devuelve la solución óptima porque la heurística no es admisible
- C. La heurística es consistente
- D. Ninguna de las respuestas anteriores es correcta

73) Sea una búsqueda de tipo A ($f(n)=g(n)+h(n)$) donde la función $h(n)$ es admisible y consistente. El algoritmo devuelve una solución desde el nodo inicial **A** al nodo objetivo **G** que atraviesa un nodo **n1**. Indica cuál de las siguientes afirmaciones es **CORRECTA**:

- A. $f(G) < h^*(A)$
- B. $h^*(A) < h(n1)$
- C. $f(n1) \leq f(G)$
- D. Ninguna de las opciones anteriores es correcta

74) Para el espacio de estados de la figura y dada una búsqueda de tipo A, donde a igualdad de criterio, se elige el nodo alfabéticamente menor, indica cuál de las siguientes afirmaciones es **CORRECTA**:



- A. La aplicación de un algoritmo en anchura devuelve la solución óptima.
- B. La solución que encuentra la búsqueda de tipo A es el nodo P.
- C. La solución que encuentra la búsqueda de tipo A es el nodo M.
- D. La solución que encuentra la búsqueda de tipo A es el nodo L.

75) Sea un problema de búsqueda donde los operadores tienen distinto coste. Existe un nodo solución, G_1 , en el nivel d_1 del árbol de búsqueda y un nodo solución, G_2 , que se encuentra en un nivel d_2 , tal que $d_2 > d_1$ (no existe solución en un nivel menor que d_1 ; además, G_1 y G_2 son las únicas soluciones en sus respectivos niveles). Se sabe que $f(G_2) < f(G_1)$ y que G_2 es una solución óptima. Indica la respuesta **INCORRECTA**:

- A. La complejidad temporal de una estrategia en anchura respecto al número de nodos generados es $O(b^{d_1+1})$
- B. Una estrategia en profundidad nunca devolverá la solución G_1
- C. Una estrategia por profundización iterativa devolverá siempre la solución G_1
- D. Una estrategia de coste uniforme devolverá siempre la solución G_2

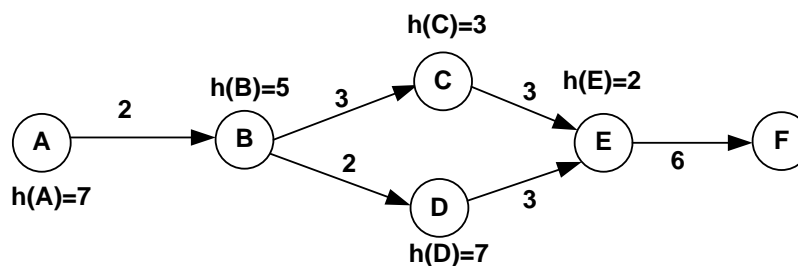
76) Sea una búsqueda de tipo A ($f(n)=g(n)+h(n)$) donde la función $h(n)$ es admisible y consistente. El algoritmo devuelve una solución desde el nodo inicial **A** al nodo objetivo **G** que atraviesa un nodo **n1**. Indica cuál de las siguientes afirmaciones es **CORRECTA**:

- A. $f(G) < f(A)$
- B. $f(G) < f(n_1)$
- C. $f(G) = g(G)$
- D. Ninguna de las opciones anteriores es correcta

77) En una búsqueda en Profundidad, con máximo nivel de profundidad $m \geq 1$, se encuentra una solución en el nivel **m**. Sabiendo que no existe ninguna solución en un nivel menor que **m**, y que la solución encontrada es la única que existe a nivel **m**, indica la respuesta **INCORRECTA**:

- A. Una estrategia de Anchura encontrará la misma solución que Profundidad
- B. Una estrategia de Profundización Iterativa encontrará la misma solución que Profundidad
- C. El número de nodos que generará Anchura será siempre mayor que en Profundidad
- D. El número de nodos que generará Profundización Iterativa será siempre mayor que en Profundidad

78) Dado el espacio de estados de la figura, donde A es el estado inicial, F el nodo meta, y se indican los costes de cada arco y la estimación $h(n)$ de cada nodo, marca la opción **CORRECTA**:

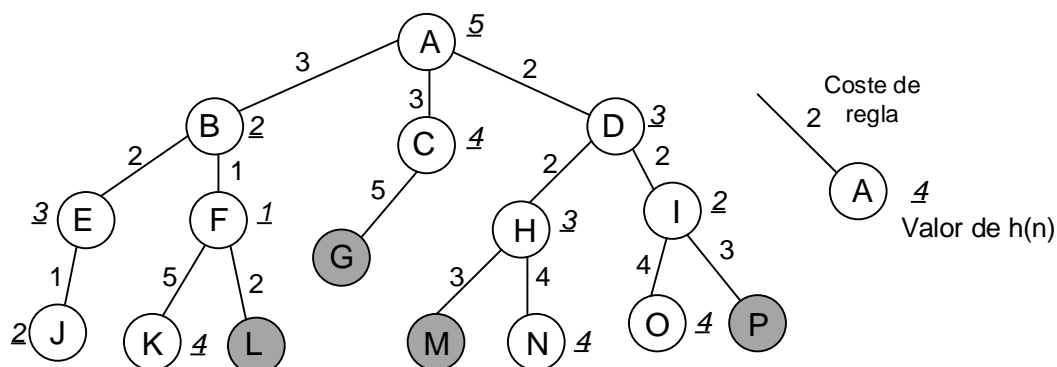


- A. La aplicación de un algoritmo A (TREE SEARCH, con control de nodos repetidos en OPEN) no obtendrá la senda óptima.
- B. La aplicación de un algoritmo A (GRAPH SEARCH, con control de nodos repetidos en CLOSED, descartando directamente el nodo si ya existe en CLOSED) obtendrá la senda óptima.
- C. La respuesta A no es cierta, debido a que $h(n)$ no es admisible
- D. La respuesta B no es cierta debido a que $h(n)$ no es consistente

79) Dados tres algoritmos A para un mismo problema, donde A1 y A2 usan una heurística admisible, tal que $\forall n, h^*(n) \geq h_2(n) > h_1(n)$. La heurística $h_3(n)$ del algoritmo A3 no es admisible. Se puede asegurar que:

- A. El algoritmo A1 generará menos nodos que A2.
- B. La solución encontrada por A1 será mejor que la encontrada por A3.
- C. El algoritmo A3 generará más nodos que A2.
- D. Los tres algoritmos A podrían encontrar la misma solución.

80) Asumiendo que aplicamos diferentes estrategias de búsqueda al espacio de estados de la figura, donde a igualdad de criterio, se elige el nodo alfabéticamente menor, indica cuál de las siguientes afirmaciones es **CORRECTA**:



- A. La aplicación de un algoritmo en anchura devuelve la solución óptima
- B. La solución que encuentra una búsqueda de tipo A es el nodo P
- C. Una estrategia de coste uniforme devolverá la misma solución que un algoritmo de tipo A
- D. La función $h(n)$ no es consistente (monótona)

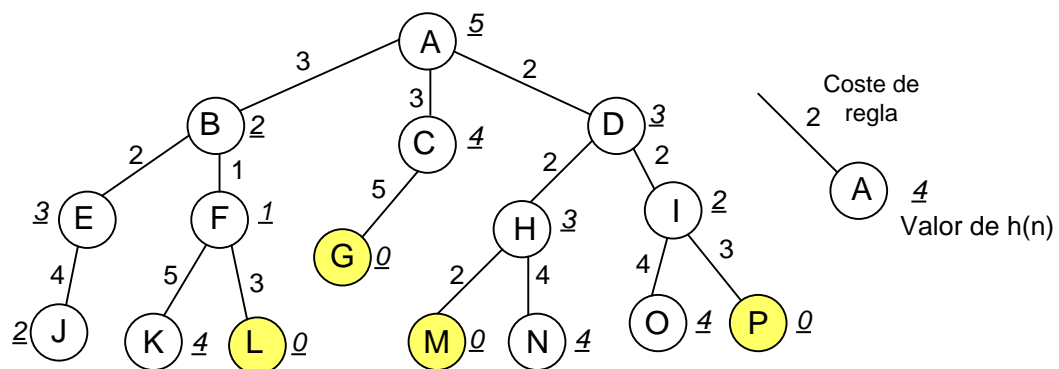
81) Si en una búsqueda en profundidad, limitada a una profundidad máxima P , se encuentra una solución en el nivel Q ($Q \leq P$), indica la respuesta **CORRECTA**:

- A. Puede asegurarse que no existe una solución en un nivel menor que Q
- B. El nivel de la solución que pueda obtener una búsqueda en profundidad, limitada a una profundidad máxima R ($R > P$), será siempre mayor que Q .
- C. Una búsqueda en anchura encontrará la solución en un nivel S tal que $S \leq Q$.
- D. Ninguna de las respuestas anteriores es correcta.

82) Supóngase un problema de búsqueda en el que todas las reglas tienen el mismo coste. Además, se dispone de una heurística $h_1(n)$ tal que $h_1(n) \leq h^*(n)$ para todo n . ¿Cuál de las siguientes estrategias de búsqueda NO garantiza encontrar una solución óptima?

- A. Anchura
- B. Voraz usando $h_1(n)$
- C. Profundidad iterativa
- D. Coste uniforme

83) Para el espacio de estados de la figura, donde a igualdad de criterio, se elige el nodo alfabéticamente menor, indica cuál de las siguientes afirmaciones es **CORRECTA**:



- A. Un algoritmo de tipo A generará igual o mayor número de nodos que una búsqueda de coste uniforme.
- B. Un algoritmo de tipo A generará menor número de nodos que una búsqueda en anchura.
- C. Una expansión en anchura generará igual o menor número de nodos que coste uniforme.
- D. Ninguna de las respuestas anteriores es cierta.

84) Para el espacio de estados de la pregunta anterior, indica cuál de las siguientes afirmaciones es **CORRECTA**:

- A. Una búsqueda en anchura, una búsqueda de coste uniforme y un algoritmo de tipo A encuentran la misma solución.
 - B. La búsqueda de coste uniforme encuentra la solución óptima.
 - C. El algoritmo de tipo A no encuentra la solución óptima porque la función heurística no es consistente.
 - D. Ninguna de las respuestas anteriores es cierta.
-

PROBLEMAS

Ejercicio 1

Dados los siguientes tipos de métodos de búsqueda en grafos:

- M1) Tipo A*, con $f(n)=g(n)+h_1(n)$
- M2) Tipo A*, con $f(n)=g(n)+h_2(n) / h_2(n) > h_1(n)$
- M3) Tipo A, no A*

Responder, justificando la respuesta:

- a) Determinar, si es posible, quién expandirá más nodos:
 - Método M1 o el método M2?
 - Método M1 o el método M3?
- b) Determinar, si es posible, que solución será mejor:
 - La que obtenga M1 o la que obtenga M2?
 - La que obtenga M1 o la que obtenga M3?
- c) Sobre el coste total del proceso de búsqueda, compárase:
 - El del método M1 con el del método M2
 - El del método M1 con el del método M3

Ejercicio 2

Responder a cada una de las siguientes cuestiones. Justifíquese la respuesta.

a) Dadas las siguientes funciones de evaluación:

$$f_1(n) = g(n) + h_1(n)$$

$$f_2(n) = g(n) + h_2(n)$$

$$f_3(n) = g(n) + h_3(n)$$

$$f_4(n) = g(n) \text{ (Búsqueda de Coste Uniforme)}$$

y conociendo que $h_1(n) \leq h_2(n) \leq h^*(n) \leq h_3(n)$,

a.1) Ordenar de mejor a peor las anteriores funciones, siendo el criterio principal la admisibilidad y el secundario la generación de nodos.

a.2) Ordenar de mejor a peor, considerando en primer lugar la generación de nodos y en segundo lugar la admisibilidad.

b) Suponiendo una función $f_5(n) = h_2(n)$, ¿se puede asegurar que garantiza que encuentra la solución óptima con un número de nodos menor que $f_2(n)$?

Ejercicio 3

Dado un algoritmo admisible A* con $f(n)=g(n) + h(n)$, siendo el coste de aplicación de cada regla 1, si cambiara $g(n)$ y el coste de cada regla fuera variable, ¿Seguiría siendo admisible?

Ejercicio 4

Suponer que tenemos dos algoritmos A*, con heurísticas $h_1(n)$ y $h_2(n)$ respectivamente para un mismo problema.

- a) Suponiendo $\forall n \ h_1(n) \leq h_2(n)$, ¿cuál de los dos algoritmos encontrará la mejor solución (coste óptimo)? ¿Cuál generará un menor espacio de búsqueda?
- b) Suponiendo que $h_1(n)$ da lugar a un A* monótono y $h_2(n)$ no, ¿cuál de los dos algoritmos tendrá un factor de ramificación menor?

Ejercicio 5

Dadas dos heurísticas admisibles (h_1, h_2), cuál de las siguientes funciones heurísticas composición de las dos anteriores es admisible y generará menos nodos en el proceso de búsqueda: (i) $\max(h_1, h_2)$, (ii) $\min(h_1, h_2)$, (iii) $h_1 + h_2$.

Ejercicio 6

Dados los siguientes Algoritmos A de búsqueda en grafos, donde $h^*(n)$ representa el coste del camino óptimo a meta:

- 1) $f_1(n) = g(n) + h_1(n) / h_1(n) \leq h^*(n)$
- 2) $f_2(n) = g(n) + h_2(n) / h_2(n) > h_1(n)$
- 3) $f_3(n) = g(n) + h_3(n) / h_1(n) \leq h_3(n) \leq (1+a) h^*(n)$, siendo a una constante positiva.
- 4) $f_4(n) = g(n) + h_4(n) / h_4(n) \leq h_1(n)$

Responder muy brevemente (1-2 líneas por cuestión), justificando la respuesta:

- a) Cuáles de los anteriores métodos es admisible?
- b) Quién tendrá mayor factor de penetrabilidad, f_1 o f_4 ?
- c) Qué podemos esperar sobre el coste de la solución que obtenga $f_3(n)$? Y de la que obtenga f_2 ?
- d) Ante un problema muy complejo, qué método sería más adecuado, en cada uno de estos casos:
 - d.1) Se requiere una respuesta en un tiempo muy corto, siendo esto prioritario al coste de la misma o, incluso, al poder encontrarla.
 - d.2) Se requiere una respuesta en un tiempo corto, pero con un buen coste.
 - d.3) Se requiere una respuesta de coste óptimo.
- e) Para un problema, en general, es posible conocer:
 - e.1) El valor de $h^*(n)$?
 - e.2) Si una función heurística $h(n)$ cumple $h(n) \leq h^*(n)$?
 - e.3) Si una función $h(n)$ cumple la condición de monotonía?

Ejercicio 7

Para la resolución de un problema mediante un proceso de búsqueda en un espacio de estados se han empleado tres operadores O_1 , O_2 y O_3 de coste 10, 20 y 30 respectivamente. El programa solicita al usuario un nivel máximo de profundidad de expansión del árbol, y devuelve el nivel dónde ha encontrado la solución, el número de nodos generados y el coste de la solución. Se sabe que para un nivel máximo de profundidad $D=5$ y una estrategia de búsqueda en anchura, la solución se ha encontrado en el nivel 2 y el coste de la solución es de 50 (en el nivel 2 sólo existe una solución). De acuerdo a esta información

- a) ¿Es esta solución encontrada la solución óptima de menor coste? Razona la respuesta. En caso negativo, ¿qué estrategia utilizarías para encontrar la solución óptima?
- b) Si el usuario selecciona $D=8$ y estrategia de búsqueda en anchura, ¿cómo será la solución encontrada? Compárala con la solución que se presenta en el enunciado.
- c) Si el usuario selecciona $D=8$ y estrategia de búsqueda por profundización iterativa, ¿cómo será la solución encontrada? Compárala con la solución que se presenta en el enunciado.

Ejercicio 8

Para un determinado problema de búsqueda, se ejecutan las siguientes estrategias: anchura, profundidad, un algoritmo A^* con $h_1(n)$, otro algoritmo A^* con $h_2(n) > h_1(n)$ y un algoritmo no A^* . El usuario limita el nivel de profundidad en la generación del árbol, y se obtienen los resultados que se muestran en la tabla. Explica razonadamente a qué estrategia de búsqueda corresponde cada línea de la tabla.

Nivel usuario	Nivel solución	Num. nodos
15	13	264
18	13	860
18	18	565
20	13	325
25	20	205

Ejercicio 9

Tenemos un problema de búsqueda cuyos operadores tienen distinto coste. Si aplicamos la estrategia BPI (Búsqueda por Profundización Iterativa) para resolver el problema, ¿qué características generales tendrá la solución encontrada?. Determina si BPI encontrará solución y en caso que así sea define la solución en términos de optimalidad, coste temporal, espacial, etc.

Ejercicio 10

La siguiente configuración del 8-puzzle:

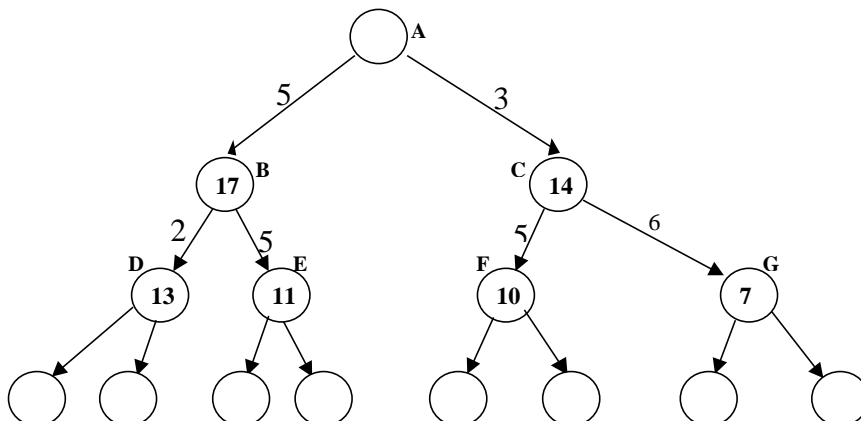
2	3	4
1	6	5
8	7	

la resolvemos con un algoritmo de búsqueda que emplea la función de evaluación $f(n)=g(n)+\text{Distancias-Manhattan}(n)$, y con otro que emplea la función de evaluación $f(n)=g(n)+\text{Piezas descolocadas}(n)$. ¿Qué proceso de búsqueda generará menos nodos para dicha configuración? ¿Porqué?

Ejercicio 11

Dado el espacio de búsqueda de la figura, **listar el orden** en que serán expandidos los nodos (A, B, C, D, E, F y G), los nodos terminales no se expanden, siguiendo las siguientes estrategias de búsqueda (eligiendo siempre el nodo de más a la izquierda). Los valores en las ramas es $g(n)$. En los nodos es $h(n)$.

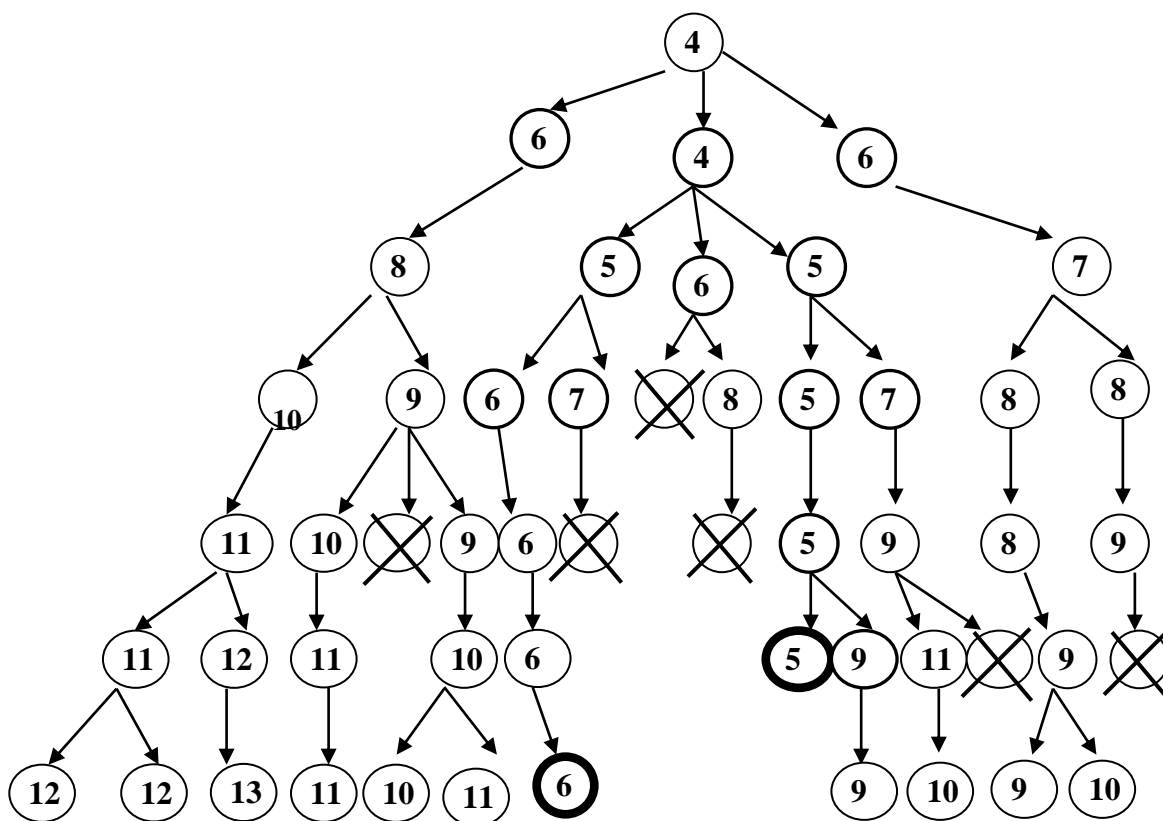
- Primero en profundidad
- Primero en anchura
- Profundización iterativa (aumento en 1 la profundidad límite de cada iteración)
- Coste uniforme
- Búsqueda voraz
- Algoritmo A



Ejercicio 12

Sea el siguiente árbol que representa un proceso de búsqueda en un espacio de estados. Los valores de cada nodo son el resultado de aplicar una función $f(n)=g(n)+h(n)$, $h(n) \leq h^*(n)$ (heurística admisible ó A^*), y el coste de aplicación de cada operador es 1. Los nodos que aparecen en **negrita** son dos posibles soluciones al problema, y los nodos que aparecen pintados con una aspa indican que no se generan nodos sucesores. Contesta a las siguientes preguntas:

- si se realiza una búsqueda heurística aplicando la función $f(n)$ mencionada anteriormente, ¿qué solución encontraría el proceso de búsqueda? ¿cuál es el coste de dicha solución? ¿cuántos nodos se generarían para encontrar dicha solución?
- a la vista de los valores $f(n)$ de cada nodo, ¿es el algoritmo aplicado un A* monótono? ¿Porqué? Justifica brevemente la respuesta.

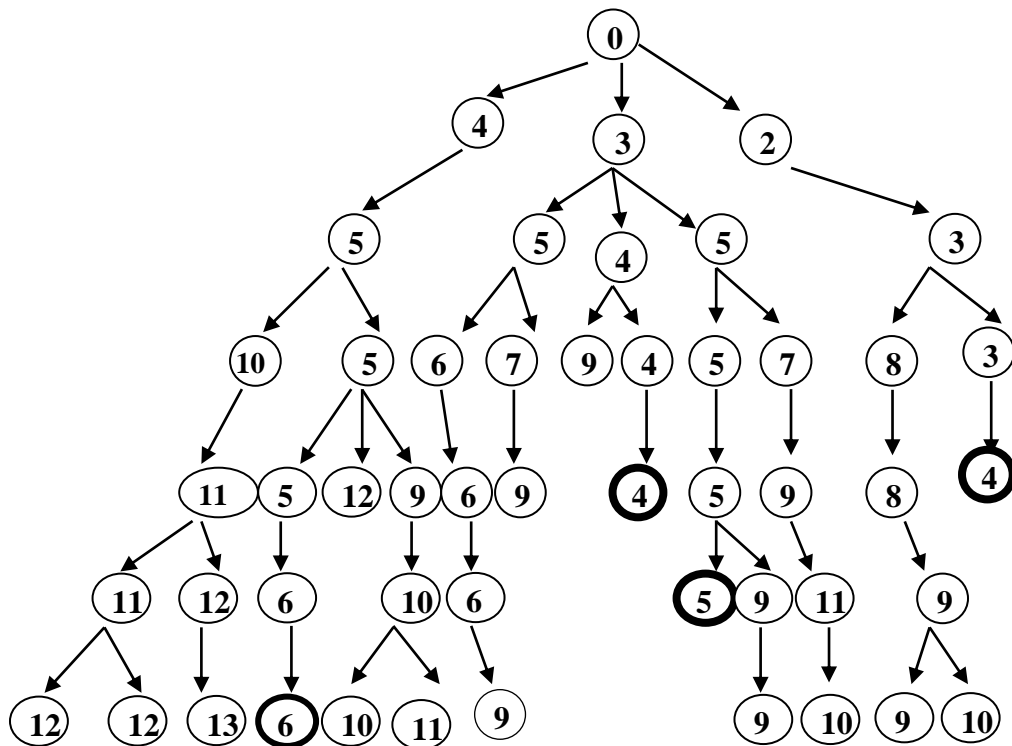


Ejercicio 13

Sea el siguiente árbol que representa el espacio de soluciones de un problema determinado. Los valores de cada nodo son el resultado de aplicar una función $f(n)=g(n)+h(n)$, donde $h(n)$ es una función heurística que da lugar a una búsqueda de tipo A* ($h(n) \leq h^*(n)$). El coste de aplicación de cada operador es uniforme (1). Los nodos que aparecen en **negrita** son estados solución del problema. Contesta a las siguientes preguntas:

- si se realiza una búsqueda heurística aplicando la función $f(n)$ anterior, ¿qué solución encontraría el proceso de búsqueda?, ¿cuántos nodos se generarían para encontrar dicha solución?
- Si sobre dicho espacio de soluciones se realiza una búsqueda por profundización iterativa ¿qué solución encontraría el proceso de búsqueda?, ¿son iguales las dos soluciones, en caso contrario cual es mejor?
- ¿Qué solución se encontraría si se realiza una búsqueda en profundidad?

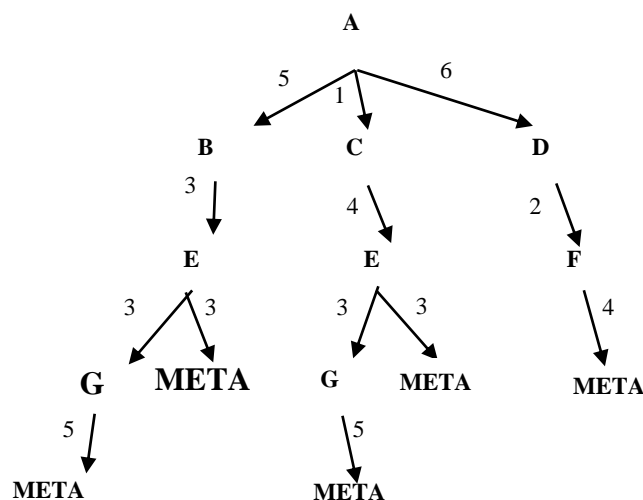
- d) Utilizando una función de evaluación $f_1(n) = g(n) + h_1(n)$, tal que $h^*(n) \geq h_1(n) > h(n)$, ¿la senda solución que se obtenga con $f_1(n)$, será mejor/peor/igual que la que se obtenía con $f(n)$? ¿qué función de evaluación realizará más/menos/igual búsqueda?



Ejercicio 14

Dado el siguiente árbol de búsqueda (donde el valor del coste de cada paso se indica en las ramas) y donde los valores $h(n)$ de cada nodo son: $h(A)=5$, $h(B)=4$, $h(C)=3$, $h(D)=5$, $h(E)=2$, $h(F)=1$, $h(G)=3$, establecer la búsqueda que se realizaría mediante:

- 1) Una búsqueda voraz (guiada por $h(n)$)
- 2) una búsqueda A
- 3) en base a los valores indicados, ¿es una búsqueda A*?



Ejercicio 15

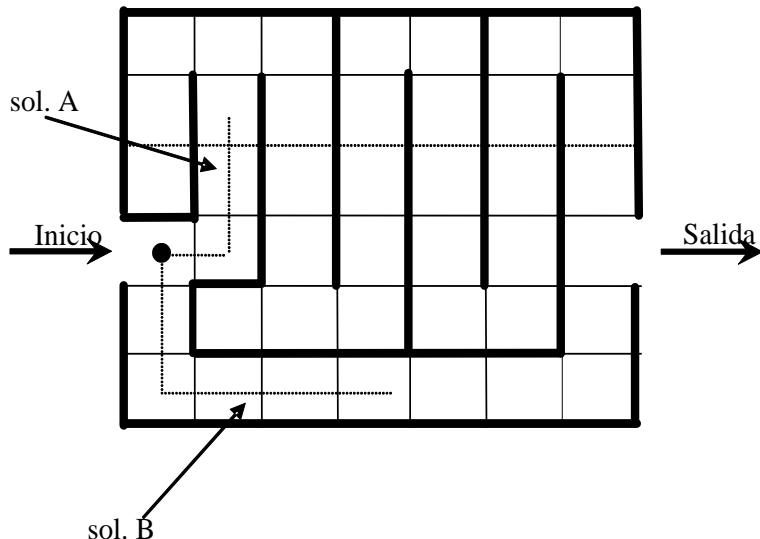
Responder brevemente a las siguientes cuestiones:

- a) Considerar un espacio de estados donde el estado inicial es el número 1 y la función sucesor (operador) para cualquier estado 'n' devuelve dos estados $2n$ y $(2n+1)$.
1. Dibujar el espacio de búsqueda, solo para los estados del 1 al 15.
 2. Suponiendo que el estado objetivo es el 11. Representar el espacio de estados, y enumerar el orden en que se expanden los nodos, en el caso en que se desarrolle una búsqueda primero en anchura y la búsqueda de profundización iterativa.
- b) Dada la siguiente función de evaluación $f(n) = (2-a) \cdot g(n) + h(n)$ tal que $h(n) \leq h^*$,
- ¿Para que valores de 'a' está garantizado que se encuentra la solución óptima?.
 - ¿Para $a=2$ se encontrará la solución óptima?.

Ejercicio 16

Sea el laberinto que se muestra en la siguiente figura donde las líneas de trazo grueso indican obstáculos; el objetivo consiste en encontrar la Salida empezando desde la posición marcada como Inicio. Los posibles operadores son ARRIBA, ABAJO, DERECHA e IZQUIERDA, los cuales solo se pueden aplicar si no se atraviesa una línea de trazo grueso (obstáculo) entre dos casillas. Como se puede observar en la figura, existen dos posibles soluciones para este problema.

Considera la heurística Distancias de Manhattan para este problema (igual aplicación que para el 8-puzzle desde Inicio hasta Salida). Asumimos que todos los operadores tienen un coste de una unidad. Asumimos que en el proceso de búsqueda se controlan los ciclos directos. Contesta a las siguientes preguntas:

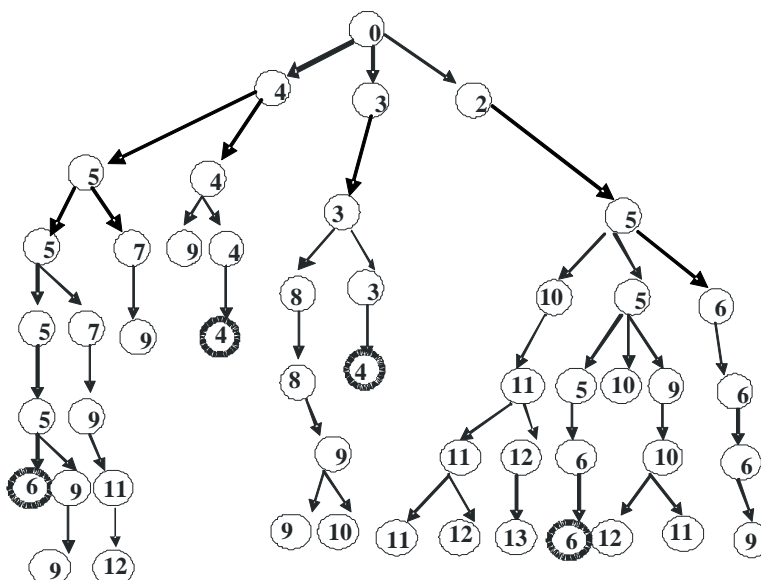


- 1) Genera el árbol que resulta de aplicar un algoritmo de tipo A ($f(n)=g(n)+h(n)$) mostrando los 7 primeros nodos que se expanden. ¿Qué solución encontrará este algoritmo? ¿Es una búsqueda A*? Razona las respuestas.
- 2) Si el coste del operador DERECHA fuera 2 en lugar de 1, la solución encontrada por el algoritmo A en el apartado 1) ¿sería la misma (misma calidad y mismo coste computacional)? ¿Por qué?
- 3) Si aplicamos una búsqueda primero-el-mejor, ¿qué solución se encontrará? Justifica tu respuesta.
- 4) Si aplicamos una búsqueda en anchura, ¿qué solución se encontrará? Justifica tu respuesta.
- 5) Supongamos que queremos aplicar una búsqueda en profundidad. Especifica qué parámetros sería necesario establecer garantizar que se encuentra una solución e indica cuál sería dicha solución.

Ejercicio 17

Sea el siguiente árbol que representa el espacio de soluciones de un problema determinado. Los valores de cada nodo son el resultado de aplicar una función $f(n)=g(n)+h(n)$, donde $h(n)$ es una función heurística que da lugar a una búsqueda de tipo A* ($h(n) \leq h^*(n)$). El coste de aplicación de cada operador es uniforme (1). Los nodos que aparecen en **negrita** son estados solución del problema. Contesta a las siguientes preguntas:

- a) Si se realiza una búsqueda heurística aplicando la función $f(n)$ anterior, ¿qué solución encontraría el proceso de búsqueda?, ¿cuántos nodos se generarían para encontrar dicha solución?
- b) Si sobre dicho espacio de soluciones se realiza una búsqueda por profundización iterativa ¿qué solución encontraría el proceso de búsqueda?, ¿son iguales las dos soluciones, en caso contrario cual es mejor? ¿Por qué?
- c) Utilizando una función de evaluación $f_1(n) = g(n) + h_1(n)$, tal que $h(n) < h_1(n) \leq h^*(n)$, ¿la senda solución que se obtenga con $f_1(n)$, será mejor/peor/igual que la que se obtenía con $f(n)$?, ¿qué función de evaluación realizará más/menos/igual búsqueda?
- d) Si se realiza una búsqueda por coste uniforme, ¿qué solución se encontraría?, ¿Qué nodos se generarían para encontrar la solución?. ¿Será mejor o peor que la encontrada en el apartado a)? ¿Por qué?



Ejercicio 18

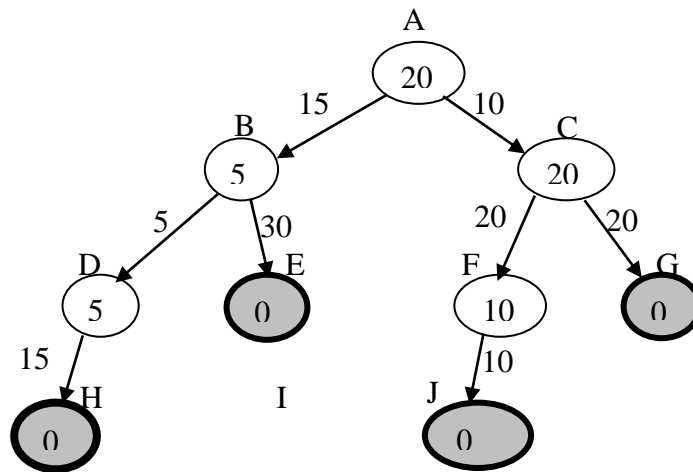
Dado el siguiente árbol de búsqueda, que representa el espacio de búsqueda completo a partir de nodo inicial A, y en el que se indica:

- en cada una de las ramas, el coste de aplicación de la regla, $g(n_i \rightarrow n_j)$,
- en cada uno de los nodos, la estimación del coste a meta, $h(n)$,
- los nodos meta, como nodos sombreados

Efectuando una búsqueda de tipo-A, a partir del nodo inicial A, responder:

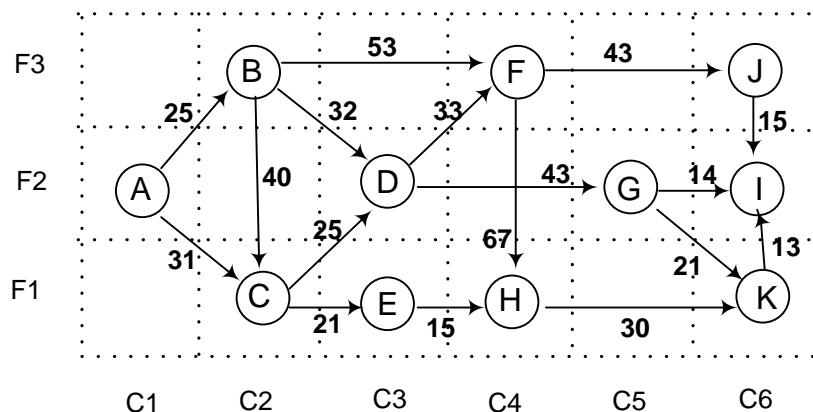
- ¿Es una búsqueda tipo A*?
- Indicar *claramente* el orden en el que se expandirían los nodos, por ejemplo:
“Paso-i: Se expande el nodo X, y genera los nodos Y y Z”
- Indicar el estado que devolvería el algoritmo como estado-meta.
- Devuelve el proceso de búsqueda la solución óptima. ¿Por qué?
- Es monótona la función $f(n)=g(n)+h(n)$ utilizada?

- f) Indicar el orden de expansión de los nodos, si se hiciera una búsqueda 'voraz' y el estado meta que devolvería



Ejercicio 19

Sea el grafo que se muestra en la figura, donde los nodos representan ciudades y los números de los arcos representan la distancia real en km. entre pares de ciudades. El grafo aparece representado en un mapa con los diferentes cuadrantes donde se sitúa cada nodo ciudad.



Considera la heurística Distancias de Manhattan para este problema, cuya aplicación es exactamente igual que para el problema del 8-puzzle desde un nodo inicio a un nodo final a través de los cuadrantes del mapa (por ejemplo, la Distancia Manhattan entre la ciudad A y E es 3; la distancia Manhattan entre la ciudad E y la I es 4).

Sea $h(n) = \text{Distancias_Manhattan}(n) * 10$; el coste de los operadores es la distancia real en km. con la que aparecen etiquetados los arcos entre ciudades.

Contesta a las siguientes preguntas justificando las respuestas. En cada una de las preguntas, indica y detalla claramente la lista de nodos ABIERTOS que se genera en cada iteración de los procesos de búsqueda.

- 1) Se quiere encontrar una solución para ir de la ciudad A a la ciudad I. Muestra el árbol que resulta de aplicar un algoritmo de tipo A ($f(n) = g(n) + h(n)$). ¿Qué solución encontrará este algoritmo? ¿Es una búsqueda A*?
- 2) Sea el mismo problema de ir de la ciudad A a la ciudad I. Muestra el árbol de búsqueda que resulta de aplicar una búsqueda primero en profundidad con un límite máximo de profundidad = 4. Asume que se expanden primero los nodos alfabéticamente anteriores. ¿Qué solución encontrará este proceso? ¿cuántos nodos se generan y cuántos se expanden en esta búsqueda?

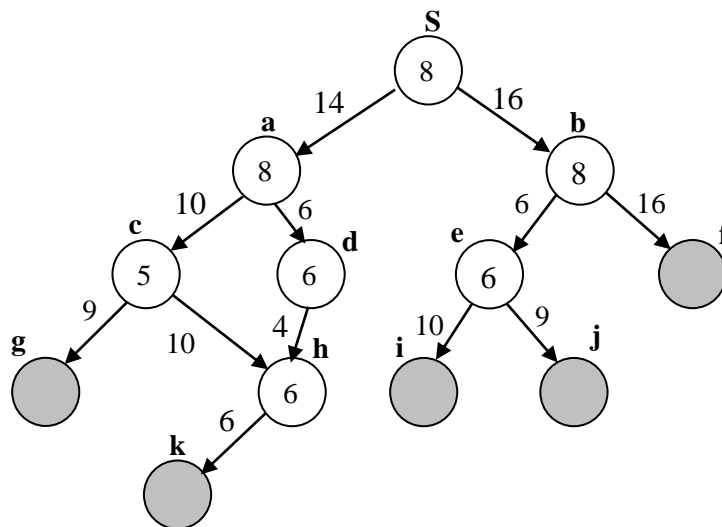
- 3) Sea el problema de ir de la ciudad A a cualquier ciudad de la columna C6. Si aplicamos una búsqueda voraz, ¿qué ciudad es la que se alcanza primero? ¿qué coste tiene esta solución? Muestra el árbol que resulta de aplicar esta búsqueda.

Ejercicio 20

Dado el siguiente espacio de búsqueda, con los valores de $g(n)$ indicados en las ramas, de $h(n)$ en los nodos, y los nodos meta sombreados, realizar un proceso de búsqueda A y un proceso de búsqueda de coste uniforme. En cada caso, indicar:

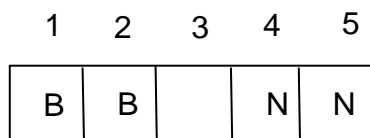
- El orden en el que se expanden los nodos y el nodo meta obtenido.
- ¿Devuelve la senda de coste óptimo? ¿Por qué?

Nota: A igualdad de la evaluación para la expansión, se expande siempre el de menor nivel.

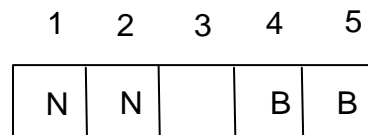


Ejercicio 21

La figura (a) muestra el estado inicial de un puzzle lineal que consta de 5 casillas: las dos de la izquierda contienen 2 fichas blancas (B), la del medio es un espacio vacío, y las dos de la derecha contienen 2 piezas negras (N). La figura (b) muestra el estado final al que se desea llegar.



(a) Estado inicial



(b) Estado final

Las acciones que se pueden aplicar en este problema son:

- A1. Una ficha se puede mover al espacio vacío si está al lado del espacio vacío (coste = 1)
- A2. Una ficha se puede mover al espacio vacío saltando sobre otra ficha de cualquier color (coste =1)
- A3. Una ficha se puede mover al espacio vacío saltando sobre dos fichas de cualquier color (coste =2)

Además, hay que tener en cuenta que las fichas B solo se pueden mover hacia la DERECHA, y las fichas N solo se pueden mover hacia la IZQUIERDA.

Sea la siguiente heurística basada en la distancia de una ficha a su zona objetivo. Se define “zona objetivo” de una ficha a las dos casillas de destino posibles de dicha ficha. Así, la zona objetivo de las fichas B son las casillas 4 y 5, y la

zona objetivo de las fichas N son las casillas 1 y 2. La función 'h' aplicada sobre una ficha que está en una posición determinada (1, 2, 3, 4 ó 5) devuelve una estimación del coste de mover esa ficha a su zona objetivo. Los valores de 'h' sobre cada ficha y cada posición son:

POSICIONES	1	2	3	4	5
funcion $h(B_i)$:	2	1	1	0	0
funcion $h(N_i)$:	0	0	1	1	2

Por ejemplo, la estimación de una ficha B que está en la posición 1 tiene un coste de 2 ($h(B1)=2$), o la estimación de una ficha N que está en la posición 1 es cero porque ya está en su zona objetivo ($h(N1)=0$). La función heurística $h(n)$ aplicada sobre un estado del problema es la suma de las heurísticas para cada ficha. Por ejemplo, dado el estado BB_NN, el cálculo de $h(n)$ sería:

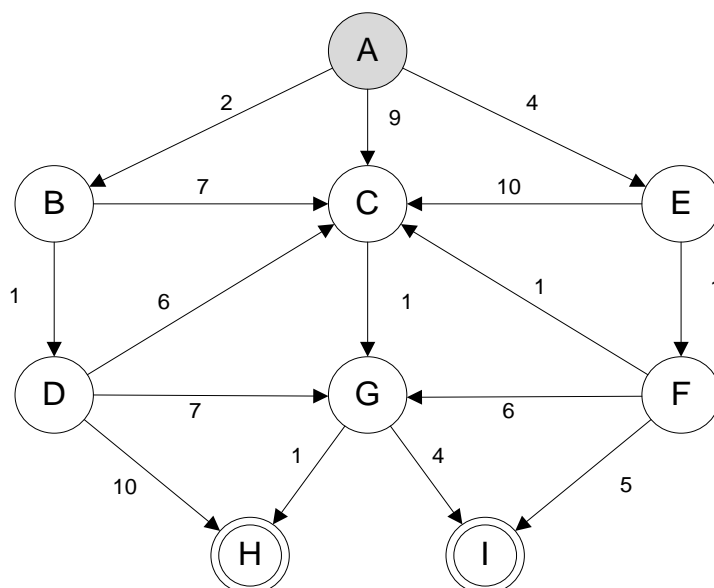
$$h(BB_NN)=h(B1)+h(B2)+h(N4)+h(N5)=2+1+1+2=6$$

Contesta a las siguientes preguntas justificando las respuestas:

- 1) Genera y dibuja el árbol que resulta de aplicar un algoritmo de tipo A ($f(n)=g(n)+h(n)$). Muestra claramente el valor $f(n)$ de cada nodo, el orden en que se expanden y el estado de la lista OPEN en cada iteración. ¿Qué solución encontrará este algoritmo y cuál es el coste de dicha solución? ¿Es una búsqueda A*? Razona las respuestas.
 - a. Nota: Para generar el árbol aplicar este orden de movimientos: primero un movimiento A1 sobre fichas B, luego A2 sobre B, A3 sobre B, A1 sobre N, A2 sobre N, y finalmente A3 sobre N.
 - b. Nota: Si dos nodos tienen mismo valor $f(n)$ expandir antes el nodo **más profundo**. Ante igualdad en el mismo nivel de profundidad, expandir primero **el más antiguo** creado.
 - c. Nota: Evitar nodos repetidos.
- 2) La aplicación de un algoritmo búsqueda voraz, ¿qué solución encontraría? ¿sería una solución óptima? Razona las respuestas.
- 3) Si se quiere aplicar una búsqueda en profundidad, ¿sería necesario limitar el nivel de profundidad máximo del árbol para garantizar que se encuentra una solución? ¿Por qué?. En caso afirmativo determina cuál sería el valor de este límite. Justifica las respuestas.

Ejercicio 22

El siguiente grafo representa un problema de espacio de estados. Los nodos del grafo son los estados del problema, las aristas conectan cada estado con sus sucesores, y el valor numérico de cada arista representa el coste de pasar de un estado al sucesor correspondiente. El estado inicial del problema es el nodo A y los estados finales son H e I. La función heurística $h(n)$ se indica en la tabla:



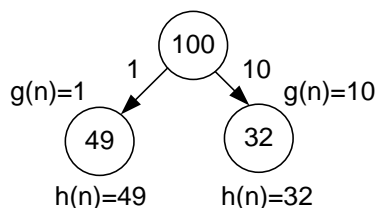
Nodo	$h(n)$
A	7
B	7
C	2
D	7
E	4
F	9
G	1
H	0
I	0

Realiza una búsqueda por coste uniforme y con un algoritmo A, indicando en cada caso, las listas de nodos abiertos y cerrados, la solución obtenida y su coste. Se recomienda controlar los nodos repetidos. ¿Se obtiene la solución óptima en ambos casos? ¿La función $h(n)$ es admisible? ¿Por qué?

Nota: Si dos nodos tienen mismo valor $f(n)$ expandir antes el nodo **más profundo**

Ejercicio 23

Considerar un espacio de soluciones donde el estado inicial es el número 100 y la función sucesor (operador) para cualquier estado ' n ' devuelve dos estados $\text{int}(n/2)-1$ (el coste de este operador es 1) y $\text{int}(n/3)-1$ (el coste de este operador es 10). Supongamos que se aplica una función de evaluación $h(n)$ para cada nodo, tal que $h(n)$ es igual al valor del nodo n . Véase el ejemplo para el primer nivel:

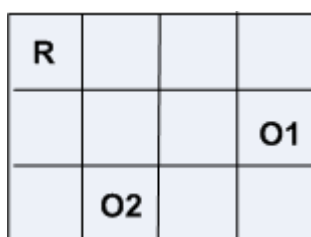


Obtener la senda solución hasta un estado meta (**la condición de nodo meta es que su valor sea negativo**) y el **coste de la senda solución obtenida**, aplicando:

- Una búsqueda de tipo A, indicando los valores $g(n)$ y $h(n)$ de cada nodo generado.
- Una búsqueda 'voraz'.
- Responder justificadamente si la función $h(n)$ aplicada es admisible.

Ejercicio 24

Queremos encontrar la ruta que tiene que seguir un robot (R) para recoger los objetos de una habitación (O1 y O2). La situación es la que se muestra en la figura. El robot solo puede moverse horizontal y verticalmente, y en cada movimiento solo puede avanzar una casilla. El robot recoge un objeto cuando llega al cuadrado donde está dicho objeto.



Sea la siguiente función heurística: $h(n)=\sum d_j$ donde d_j es la distancia (número de movimientos) que tiene que realizar el robot para alcanzar el objeto ' j '. Es decir, la función heurística es el sumatorio de la distancia de Manhattan del robot a cada objeto pendiente de recoger en la habitación. Cuando el robot recoge un objeto, asumiremos que la distancia a dicho objeto es 0, y que el objeto se mueve con el robot a partir de ese estado. El coste de cada movimiento es 1.

- Genera y dibuja el árbol que resulta de aplicar un algoritmo de tipo A ($f(n)=g(n)+h(n)$) para encontrar el camino que tiene que seguir el robot para recoger los dos objetos. Muestra claramente el valor $f(n)$ de cada nodo, el orden en que se expanden y el estado de la lista OPEN en cada iteración. ¿Qué solución encontrará este algoritmo y cuál es el coste de dicha solución? La función ' h ', ¿es una heurística A*? ¿Por qué? Razona las respuestas.
- Partiendo del árbol desarrollado en el apartado 1, la aplicación de un algoritmo búsqueda voraz, ¿qué solución encontraría? ¿sería una solución óptima? ¿Por qué? Justifica las respuestas.

NOTAS para realizar la expansión de los árboles:

1. Utilizar el siguiente orden de aplicación de movimientos: primero un movimiento ARRIBA, segundo ABAJO, tercero DERECHA, cuarto IZQUIERDA.
2. Si dos nodos tienen mismo valor $f(n)$ expandir antes el nodo **más profundo**. Ante igualdad en el mismo nivel de profundidad, expandir primero **el más antiguo** creado.
3. Evitar nodos repetidos.

Ejercicio 25

Tenemos cinco monedas dispuestas en una fila tal como indica la figura (a), donde **A** representa el anverso de la moneda, y **R** el reverso. La única operación que se puede realizar en este problema es dar la vuelta (de **A** a **R** ó de **R** a **A**) a cualquier **par de monedas contiguas** (moneda1-moneda2, moneda2-moneda3, moneda3-moneda4 ó moneda4-moneda5), y dicha operación tiene un coste de 1. Deseamos obtener la situación que se muestra en la figura (b).

Función heurística. Para calcular la función heurística $h(n)$ se analizan las monedas en dos fases, en la primera fase se analizan de izquierda a derecha y en la segunda fase de derecha a izquierda. En cada fase, se analizan los siguientes pares de monedas contiguas:

- de izda. a dcha.: se mira moneda 1 con moneda 2 y moneda 3 con moneda 4
- de dcha. a izda.: se mira moneda 5 con moneda 4 y moneda 3 con moneda 2

1	2	3	4	5
A	R	A	R	A

(a) Estado inicial

1	2	3	4	5
R	R	R	A	R

(b) Estado final

Los valores de 'h' para cada pareja de monedas son:

- $h(\text{moneda X colocada, moneda Y colocada}) = 0$ puntos
- $h(\text{moneda X descolocada, moneda Y descolocada}) = 1$ punto
- $h(\text{moneda X colocada, moneda Y descolocada}) = 2$ puntos
- $h(\text{moneda X descolocada, moneda Y colocada}) = 2$ puntos

En resumen, si las dos monedas contiguas están colocadas entonces sumamos 0 puntos. Si las dos monedas contiguas están descolocadas entonces sumamos 1 punto. Si de las dos monedas contiguas una está colocada y la otra no entonces sumamos 2 puntos.

La función $h(n)$ para un estado 'n' del problema se calcula del siguiente modo:
 $h(n) = \min(h_{\text{izq_der}}(n), h_{\text{der_izq}}(n))$ (mínimo valor entre recorrer las fichas de izquierda a derecha y recorrer las fichas de derecha a izquierda) donde:

$$\begin{aligned} h_{\text{izq_der}}(n) &= h(\text{moneda1, moneda 2}) + h(\text{moneda3, moneda4}) \\ h_{\text{der_izq}}(n) &= h(\text{moneda5, moneda 4}) + h(\text{moneda3, moneda2}) \end{aligned}$$

Por ejemplo, para la configuración de la figura (a), el valor de $h(n)$ sería:

$$h(\text{ARARA}) = \min(2+1, 1+2) = 3$$

Contesta a las siguientes preguntas justificando las respuestas:

- 1) Genera y dibuja el árbol que resulta de aplicar un algoritmo de tipo A ($f(n) = g(n) + h(n)$) al estado inicial de la figura ¿a). Muestra claramente el valor $f(n)$ de cada nodo, el orden de expansión de los nodos y el estado de la lista OPEN en cada iteración. ¿Qué solución encontrará este algoritmo y cuál es el coste de dicha solución? ¿La solución obtenida es de coste óptimo?

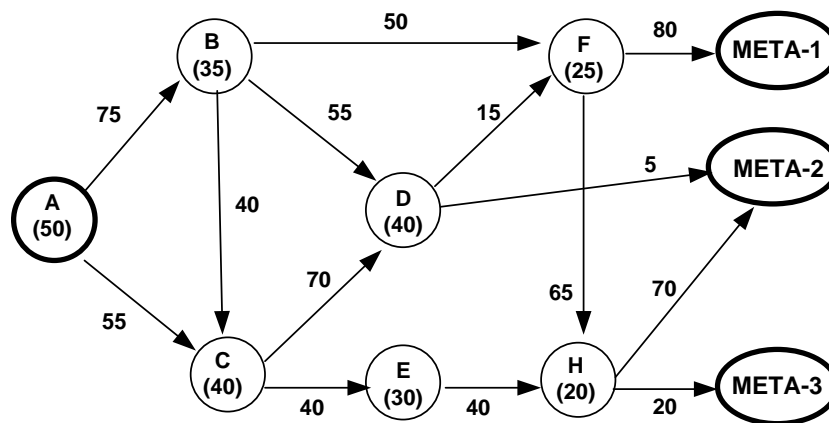
- 2) La heurística utilizada, ¿es admisible? Es decir, ¿se trata de una búsqueda A*? ¿por qué? Razona la respuesta.
- 3) Asumiendo que un usuario selecciona $m=5$ como máximo nivel de profundidad del árbol, ¿qué solución devolverá la estrategia de profundidad y cuál es el coste de dicha solución?.

NOTAS

- a) Para generar los árboles aplicar este orden de movimientos: primero moneda 1 con moneda 2, luego moneda 2 con 3, luego moneda 3 con 4, y finalmente moneda 4 con 5.
- b) Si dos nodos tienen mismo valor $f(n)$ expandir antes el nodo **más profundo**. Ante igualdad en el mismo nivel de profundidad, expandir primero **el más antiguo** creado.
- c) Aplicar SIEMPRE control de nodos repetidos.

Ejercicio 26

Sea el siguiente espacio de estados, donde se representa el estado inicial (A), los estados meta (Meta-1, Meta-2 y Meta-3), y los estados accesibles desde cada estado. En los arcos se etiqueta el coste del arco y en los nodos el valor de una función $h(n)$.



a) Obtener la senda solución desde el estado A hasta un estado meta, aplicando los siguientes métodos de búsqueda, indicando claramente, para cada uno de ellos:

- El árbol de búsqueda que genera, remarcando la senda solución que obtiene
- Indicar, de forma clara, el orden en el que se han expandido los nodos.
- Indicar, de forma clara, la meta alcanzada y coste de la senda solución encontrada.

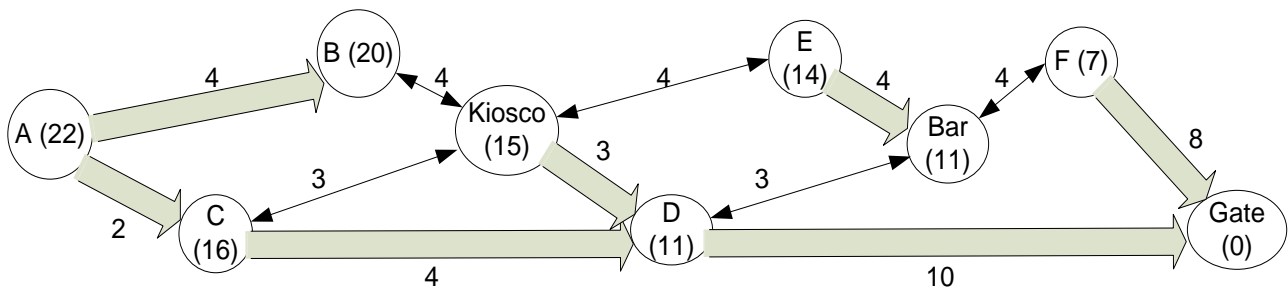
Métodos a aplicar (a igualdad de criterio, se expanden los nodos en orden alfabético):

- a.1) Búsqueda voraz
- a.2) Algoritmo A
- a.3) Coste uniforme

b) Es posible conocer, con alguno de los métodos aplicados, la senda óptima? La función $h(n)$ es admisible?

Ejercicio 27

Un viajero se encuentra en un aeropuerto (cuyo mapa simplificado se muestra en la figura) y debe llegar por el camino más corto desde el punto A a la puerta de embarque (Gate), pasando antes por el kiosco para comprar el periódico y por el bar para tomar un café. Las líneas gruesas representan cintas transportadoras de pasajeros (en un solo sentido) mientras que las líneas finas representan pasillos para ir de un punto a otro (en ambos sentidos). Los números sobre cada línea representan el coste de tomar esta cinta o pasillo para ir de un punto X a un punto Y. Los valores dentro de los nodos indican el coste estimado para ir desde el punto X a la puerta de embarque.



Se pide:

a) Obtener la senda solución desde el punto A hasta la puerta de embarque, aplicando un algoritmo A con control de nodos repetidos. Un estado de este problema consta de la siguiente información: el punto donde se encuentra el pasajero, si éste ya ha pasado por el kiosco o no, y si el pasajero ha pasado ya por el bar o no. Por tanto, dos estados se consideran repetidos si contienen exactamente la misma información respecto a estos tres elementos, independientemente de la senda utilizada para llegar a este estado. Es decir, el estado alcanzado tras el recorrido $A \rightarrow B \rightarrow \text{Kiosco} \rightarrow D$ y el estado alcanzado tras el recorrido $A \rightarrow C \rightarrow \text{Kiosco} \rightarrow D$, se consideran estados repetidos ya que en ambos estados el viajero está en el punto D y ha visitado el kiosco, pero no el bar. Por tanto, una senda solución no lo será únicamente por alcanzar la puerta de embarque, sino que se deberá tener en cuenta que, en este camino, se ha visitado el kiosco y el bar. Se debe indicar claramente:

- El árbol de búsqueda que genera el algoritmo, remarcando la senda solución que obtiene.
- El orden en el que se han expandido los nodos.
- El coste de la senda solución encontrada.
- El número de nodos generados y expandidos.

Nota: En el caso del mismo valor de $f(n)$, se debe expandir en primer lugar el nodo más profundo.

b) ¿Es admisible la heurística utilizada? Justificar la respuesta.

c) Aplicar de nuevo el algoritmo A con la siguiente heurística: dado un estado 'e', donde la posición actual del pasajero es n , el coste estimado para alcanzar la puerta de embarque será el valor indicado en el nodo n ($h(n)$) más 10, si no se ha visitado el kiosco, más 10, si no se ha visitado el bar. Por ejemplo, dado un estado 'e' con el recorrido $A \rightarrow B \rightarrow \text{Kiosco} \rightarrow D$, $h(e) = 11$ (por estar en D) + 10 (por no haber visitado el bar) = 21.

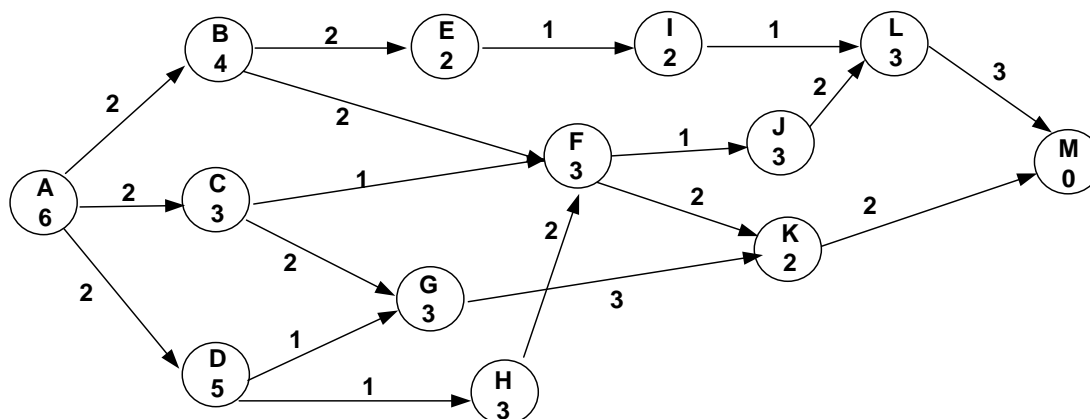
- Se debe indicar claramente el árbol de búsqueda que genera el algoritmo, remarcando la senda solución que obtiene; el orden en el que se han expandido los nodos; el coste de la senda solución encontrada y el número de nodos generados y expandidos.
- Compara esta heurística con la utilizada en el apartado (a). ¿Es admisible esta nueva heurística? Justifica la repuesta.

Ejercicio 28

Sea el grafo que se muestra en la figura, donde el estado inicial es A y el estado final que se quiere alcanzar es M. Los valores dentro de los nodos indican el valor $h(n)$ de cada nodo. Los valores de las aristas representan el coste para ir de un nodo a otro.

- a) Aplicar un algoritmo A con control de nodos repetidos. Mostrar el árbol de búsqueda que genera el algoritmo, indicar claramente el valor $f(n)$ de cada nodo y el orden en el que se expanden los nodos. Señala cuál es el coste del camino encontrado, el número de nodos generados y expandidos. **NOTA:** Si dos nodos tienen el mismo valor $f(n)$, expandir antes el nodo alfabéticamente anterior.

- b) La solución encontrada por el algoritmo A, ¿es la solución óptima? ¿La función $h(n)$ es admisible? Justifica las respuestas.
- c) La función $h(n)$ es no monótona. Muestra los valores del grafo y/o árbol que indican que $h(n)$ es no monótona y explica las consecuencias de la ausencia de esta propiedad en el árbol desarrollado.



- d) Muestra el árbol de búsqueda que resulta de aplicar una búsqueda primero en profundidad con un límite máximo de profundidad $m=4$ y con control de nodos repetidos. Muestra el estado de las listas OPEN y CLOSED en cada iteración. ¿Qué solución devolverá la estrategia de profundidad y cuál es el coste de dicha solución? **NOTA:** Si dos nodos están en el mismo nivel de profundidad, expandir antes el nodo alfabéticamente anterior.

Ejercicio 29

Dada la siguiente configuración inicial de un tablero lineal, donde hay dos piezas negras (N), dos piezas blancas (B) y una casilla vacía:

1	2	3	4	5
N	N	B	B	

Estado inicial

Considera las siguientes acciones:

- Una pieza se puede mover a la casilla vacía adyacente con un coste igual a 1
- Una pieza puede saltar sobre otra (solo sobre una) para colocarse en la casilla vacía con un coste igual a 1. Si la pieza que salta y la pieza sobre la que se salta son de diferente color entonces la pieza sobre la que se salta cambia de color.

El objetivo es que todas las piezas del tablero sean de color negro pudiendo estar en cualquier posición. Es decir, la posición final de la casilla vacía es indiferente.

- 1) Considera la siguiente función heurística: $h(n) = \sum_{i=1}^5 b(i) * c(i)$ donde:

- $b(i)=1$, si la casilla i contiene una pieza B. $b(i)=0$, en cualquier otro caso
- $c(i)=1$, si una de las casillas adyacentes $(i-1)$ o $(i+1)$ es la casilla vacía. $c(i)=2$, en cualquier otro caso

En otras palabras, la heurística solo considera las piezas B, anotando un punto de coste si la casilla vacía es adyacente a la pieza B ó 2 puntos de coste en caso contrario.

- Dibuja el árbol de búsqueda que resulta de aplicar un algoritmo A con $f(n)=g(n)+h(n)$.
- Indica sobre el árbol el *orden de expansión* de los nodos.
- Muestra la solución encontrada y el coste de dicha solución.

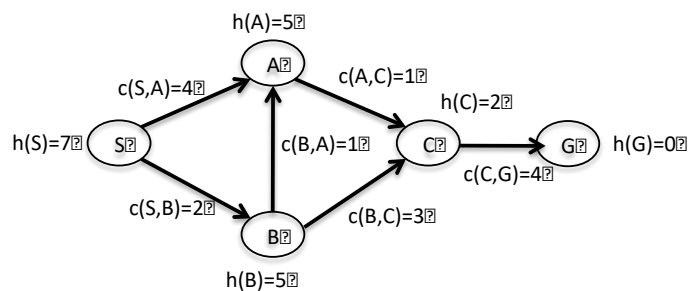
NOTAS: (a) Evitar nodos repetidos. (b) Si dos nodos tienen el mismo valor de $f(n)$ expandir antes el nodo más profundo.

- 2) La función $h(n)$ descrita en 1), ¿es admisible? Razona la respuesta.
- 3) Sin necesidad de desarrollar un árbol, ni realizar ninguna búsqueda adicional, contesta a las siguientes preguntas razonadamente:
 - a) ¿Cuántas iteraciones (árboles de búsqueda) tendría que realizar la estrategia Búsqueda por Profundización Iterativa para encontrar una solución?
 - b) Si el coste de la operación de salto fuera 2 en lugar de 1, ¿encontraría el algoritmo A del apartado 1 la misma solución?

Ejercicio 30

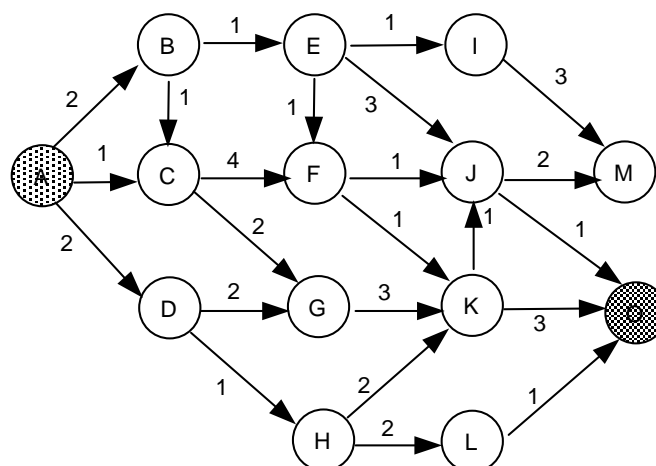
Dado el espacio de estados de la figura desde un nodo-origen (S) a nodo-meta (G), donde $h(x)$ denota el valor de aplicar la función heurística 'h' al estado 'x', $c(x,y)$ denota el coste de la acción que permite ir del estado 'x' al estado 'y'. Se pide:

- a) Realizar una búsqueda tipo A. Obtene la senda óptima de (S) a (G)?
- b) La heurística $h(n)$ es admisible? Y consistente?



Ejercicio 31 (Examen 2013)

Sea el siguiente grafo donde cada arco indica su coste y la tabla indica la estimación del coste 'h' hasta la solución. El nodo 'A' es el estado inicial y el nodo 'O' es el estado final.

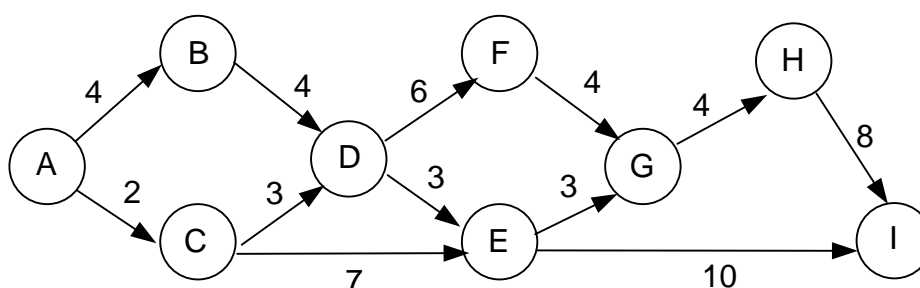


n	A	B	C	D	E	F	G	H	I	J	K	L	M	O
h(n)	6	5	6	6	3	5	5	4	8	2	2	1	5	0

1. 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. Si dos nodos tienen el mismo valor de $f(n)$, expandir antes el nodo alfabéticamente anterior.
2. De acuerdo a 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.
3. Sin desarrollar un árbol de búsqueda, contesta a las siguientes preguntas y justifica la respuesta:
 - a. ¿Qué estrategia utilizarías si queremos encontrar la solución que atraviese el menor número de nodos? Indica una solución que encontraría esta estrategia.
 - b. La aplicación de un algoritmo de profundización iterativa, ¿en qué nivel del árbol encontraría la solución? ¿por qué?
 - c. Si aplicamos un algoritmo en profundidad y no establecemos un límite máximo de profundidad, ¿encontraría el algoritmo una solución? ¿por qué?

Ejercicio 32 (Examen 2013)

Sea el siguiente grafo donde cada arco indica su coste y la tabla indica la estimación del coste 'h' hasta la solución. El nodo 'A' es el estado inicial y el nodo 'I' es el estado final.



n	A	B	C	D	E	F	G	H
h(n)	15	14	13	12	11	10	9	8

- 1) Asumiendo que se aplica un algoritmo en anchura, que ante el mismo valor de la función ' $f(n)$ ' se expande antes el nodo alfabéticamente anterior y que se realiza control de nodos repetidos (descartar los nodos más profundos o nodos expandidos con anterioridad en caso del mismo nivel de profundidad), contesta a las siguientes preguntas:
 - a) Escribe los nodos del camino solución desde el nodo A hasta el nodo I.
 - b) ¿Cuántos nodos se han generado en total y cuántos nodos se han expandido en el árbol?
- 2) Asumiendo que se aplica la versión grafo de un algoritmo A con control de nodos repetidos, contesta a las siguientes preguntas:

- a) Escribe los nodos del camino solución desde el nodo A hasta el nodo I y el coste de dicho camino solución; el camino encontrado, ¿es la solución óptima?
 - b) ¿Cuántos nodos se han generado en total y cuántos nodos se han expandido en el árbol?
 - c) Indica los nodos expandidos y su orden de expansión.
- 3) Responde brevemente a las siguientes preguntas justificando las respuestas:
- a) La función heurística de este problema, ¿es admisible? ¿por qué?
 - b) La función heurística de este problema, ¿es consistente? ¿por qué?
 - c) Si aplicamos un algoritmo de profundización iterativa en este problema, ¿qué solución encontraría? Indica los nodos del camino solución así como el número total de nodos generados.