Inteligencia Artificial Hoja 1	Publicación: 2015/01/30 Límite entrega: 2015/02/10/11:00 Resolución en clase: 2015/02/10 (grupo mañana), 2015/02/11 (grupo tarde)

1. El 23 de junio de 2013 se cumplieron 101 años del nacimiento de uno de los científicos más importantes del siglo XX: Matemático, lógico, pionero de la informática, formó parte de un equipo de ingenieros, científicos y técnicos que descifraron códigos utilizados por los alemanes durante la segunda guerra mundial. Hizo asimismo contribuciones seminales en el campo de la Inteligencia Artificial y de la morfogénesis. Posteriormente fue condenado judicialmente por ser homosexual. La pena consistió en su castración mediante un tratamiento con hormonas. Murió poco tiempo después de haber finalizado el tratamiento; probablemente por suicidio.

   El objetivo de este ejercicio es ayudar a un ordenador a encontrar el nombre de pila de este gran matemático e informático a partir de las letras 'aAln'. **Las mayúsculas y las minúsculas son distintas**: La primera letra del nombre es una mayúscula. Para ello, vamos a aplicar el algoritmo A* con las siguientes especificaciones:
   - Inicialmente las letras se encuentran en orden alfabético.
   - Las únicas acciones permitidas son **permutaciones de la primera letra con otra letra de la secuencia**. El **orden** de las acciones corresponde a realizar **permutaciones de letras de izquierda a derecha**.
   - *Coste* de las acciones:
     - **Permutaciones que solo involucran minúsculas** tienen **coste 2**.
     - **Permutaciones que involucran alguna mayúscula** tienen **coste 1**.
   - La **heurística** es el **número de letras que no están en la posición correcta**.

   Responde las siguientes cuestiones:

   a. ¿Cómo se formalizan los estados de búsqueda?
   Cada estado corresponde a una permutación de las letras 'aAln'

   b. ¿Cuál es el estado inicial?
   El estado 'aAln'

   c. ¿Cuál es el test objetivo?
   ¿Coincide la permutación de letras del estado en cuestión con la secuencia 'Alan'?

   d. ¿Cuál es la profundidad máxima de la solución?
   4!-1 = 23

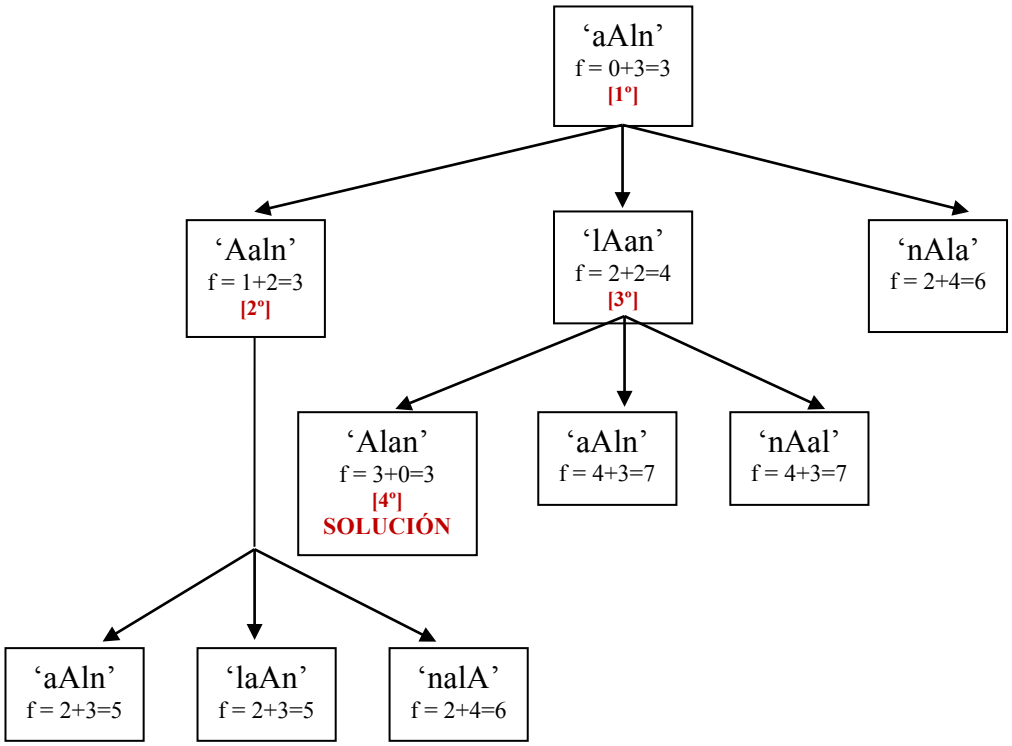
   e. ¿Es la heurística admisible? Explicad la razón.
   La heurística no es admisible. Por ejemplo, el coste de 'lAan' a 'Alan' es 1, cuando h('lAan') es 2.

   f. ¿Es la heurística monótona? Explicad la razón.
   Al no ser admisible no es monótona.
   Se puede comprobar directamente que no cumple
   $\forall n, n'$(sucesor de n) $h(n) \leq \Gamma(n \rightarrow n') + h(n')$
   En concreto

n = ‘nlaA’ [h(n) = 2]
n’ = ‘Alan’ [h(n’) = 0]
Γ(n→n’) = 1
con 2 ≥ 1 + 0

g. ¿Garantiza A* con esta heurística y eliminación de estados repetidos encontrar la solución óptima? Explicar la razón.
No, ya que no es monótona.

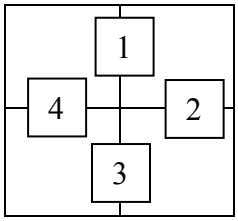
Detalla el árbol generado por A* con eliminación de estados repetidos. Indica para cada nodo los valores g + h = f y el orden en el que el intento de exploración se realiza (es decir, los nodos repetidos y la meta también reciben numeración). En caso de que haya empates, se elegirá primero en la exploración el nodo que haya sido generado antes.



h. ¿Encuentra A* la solución óptima en este ejemplo? Justifica tu respuesta, poniéndola en relación con tu respuesta al apartado (g).

En este caso encuentra la solución óptima.
El resultado no está en contradicción con la respuesta a (g): En este caso, dado que no ha sido necesario eliminar estados repetidos, la admisibilidad de la heurística garantiza encontrar el óptimo.
Incluso con una heurística no admisible podríamos haber encontrado la solución óptima en ejemplos particulares.

2. Consideremos la siguiente disposición de 12 palillos:



El objetivo es **eliminar palillos** de entre los que se encuentran **en el centro del cuadrado** de superficie 2×2, los cuales están numerados del 1 al 4 en el sentido de las agujas del reloj, de forma que queden únicamente dos cuadrados. En la figura actual hay 5 cuadrados: los 4 cuadrados adosados de superficie 1×1 y el cuadrado externo de superficie 2×2.

Vamos a aplicar el algoritmo A* para resolver el puzle:

- El estado inicial es el estado indicado en la figura.
- Las acciones permitidas son **eliminar un palillo de entre los 4 etiquetados**. El **orden de las acciones** es el dado por el **valor de la etiqueta**.
- Coste de las acciones:
  - El coste de **eliminar un palillo** cuya etiqueta es **impar es 1**
  - El coste de **eliminar un palillo** cuya etiqueta es **par es 0.5**
- El valor de la **heurística** en un estado de búsqueda es el **valor absoluto de la diferencia entre el número de cuadrados de la configuración de palillos en cuestión y 2**.

Responde las siguientes cuestiones:

a. ¿Cómo se formalizan los estados de búsqueda?
Los estados son disposiciones de 12-n palillos, donde n es la profundidad del nodo correspondiente en el árbol de búsqueda. Cada estado se identifica mediante una lista con las etiquetas de los palillos que permanecen en la disposición en cuestión

b. ¿Cuál es el estado inicial?
El estado inicial es '1234'

c. ¿Cuál es el test objetivo?
¿Es el número de cuadrados en la configuración actual igual a 2?

d. ¿Cuál es el factor de ramificación del árbol de búsqueda? En caso de que dependiera de la profundidad, indicad explícitamente esta dependencia

El factor de ramificación es 4-n, donde n es la profundidad del nodo que se expande.

e. ¿Es la heurística admisible? Explicad la razón.

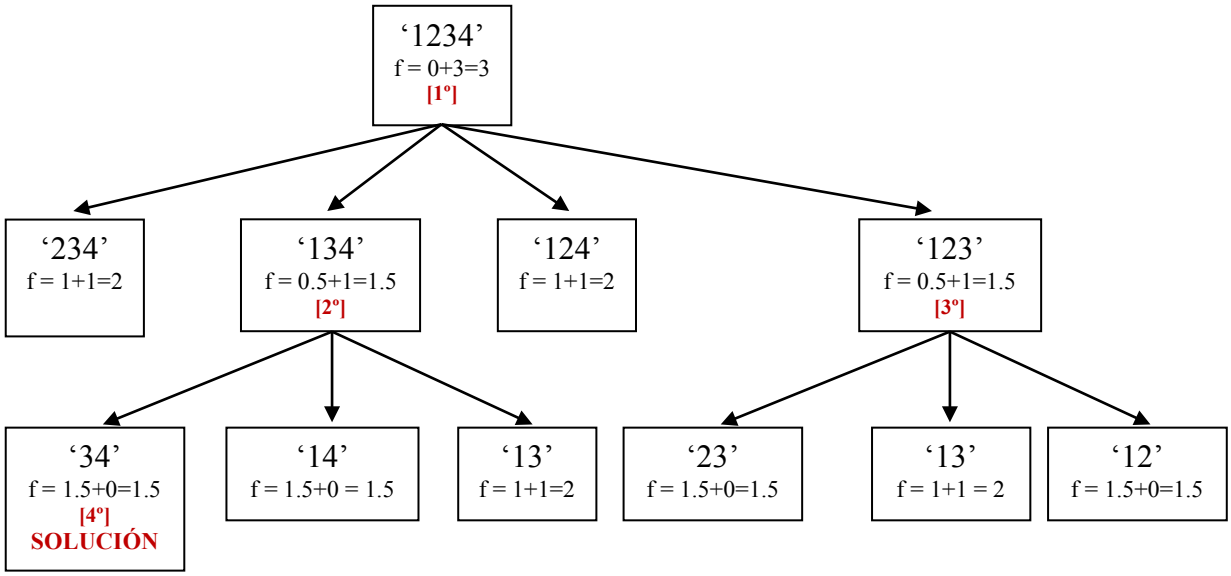
La heurística no es admisible. Por ejemplo, en este caso el coste de la solución óptima es 1.5. Sin embargo, el valor de la heurística para el nodo inicial es 3, mayor que el coste óptimo.

f. ¿Es la heurística monótona? Explicad la razón.
No, ya que no es admisible.

g. ¿Garantiza A* con esta heurística, sin eliminación de estados repetidos encontrar la solución óptima? Explicar la razón.

No, ya que no es admisible.

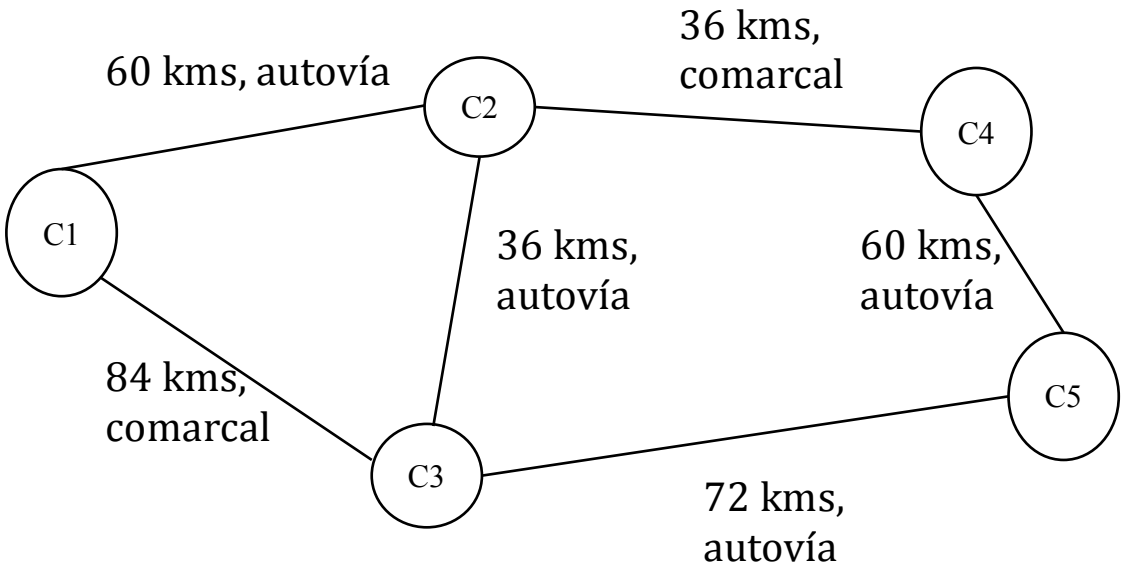
h. Detalla el árbol generado por **A* sin eliminación de estados repetidos**. Indica para cada nodo los **valores g + h = f** y el **orden** en el que el intento **de exploración** se realiza (es decir la meta también recibe numeración). En caso de que haya **empates**, se elegirá en la exploración el **nodo** que haya sido **generado antes**.



i. ¿Encuentra A* la solución óptima en este ejemplo? Justifica tu respuesta, poniéndola en relación con tu respuesta al apartado (g).

En este caso encentra la solución óptima.
El resultado no está en contradicción con la respuesta a (g): En este caso, dado que no ha sido necesario eliminar estados repetidos, la admisibilidad de la heurística garantiza encontrar el óptimo.
Incluso con una heurística no admisible podríamos haber encontrado la solución óptima en ejemplos particulares.

3. Consideremos el siguiente mapa de ciudades, donde cada nodo representa una ciudad, y cada arco una carretera que conecta directamente dos ciudades (asociados a cada arco tenemos la longitud de dicha carretera y el tipo de carretera):



Se pretende llegar a la ciudad objetivo (**ciudad 5**) con el menor coste posible en euros.
Hay dos tipos de carreteras: comarcales y autovías. En cada una de estas se estima que el coste promedio va a ser diferente. En autovías se estima un coste de combustible de 5 euros cada 100 Kms, y en comarcales de 10 euros cada 100 Kms. Además, en cada autopista hay un coste por peaje de 5 euros por trayecto.

Como información adicional disponemos de la distancia **en línea recta** de cada una de las ciudades a la objetivo:

Ciudad	**Distancia a C5 (Kms)**
1	120
2	90
3	60
4	42
5	0

Responde a las siguientes cuestiones:

- Define una heurística admisible para este problema y muestra los valores de esta h

- Determina, **para cada estado**, el coste para ir de dicho estado a cada uno de sus posibles sucesores.

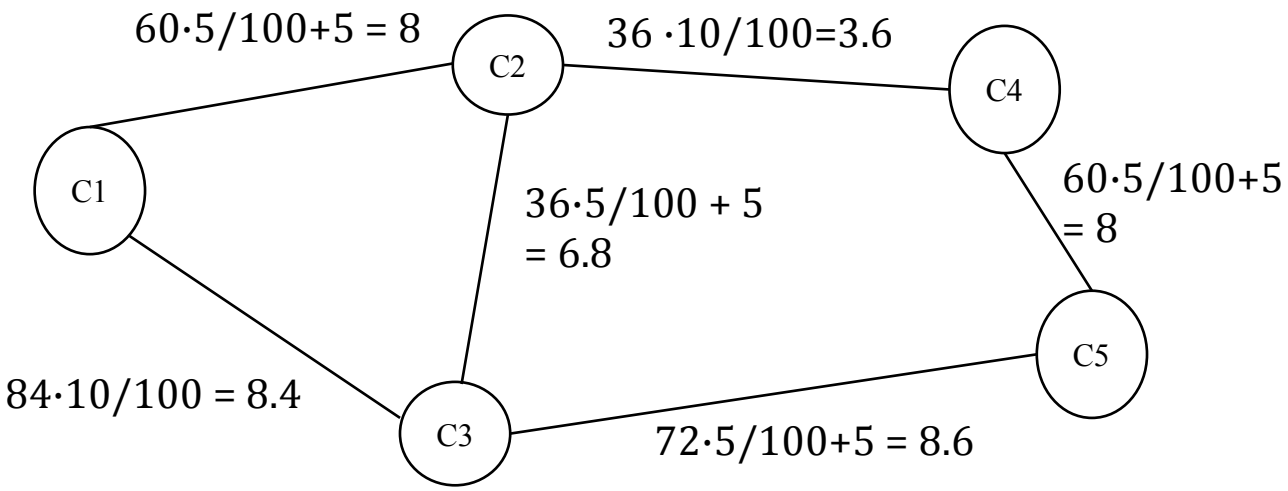
- Si el **estado inicial es la ciudad 1**, ¿A qué solución llegaría búsqueda-en-árbol (es decir, sin eliminar estados repetidos) usando A* y esa heurística?
  Muestra paso a paso qué nodos expandiría el algoritmo, cuáles genera, y en qué orden.
  ¿Qué solución encuentra? ¿es la óptima? ¿cuántos euros en total costaría llegar en nuestro coche a la ciudad objetivo por esta trayectoria?

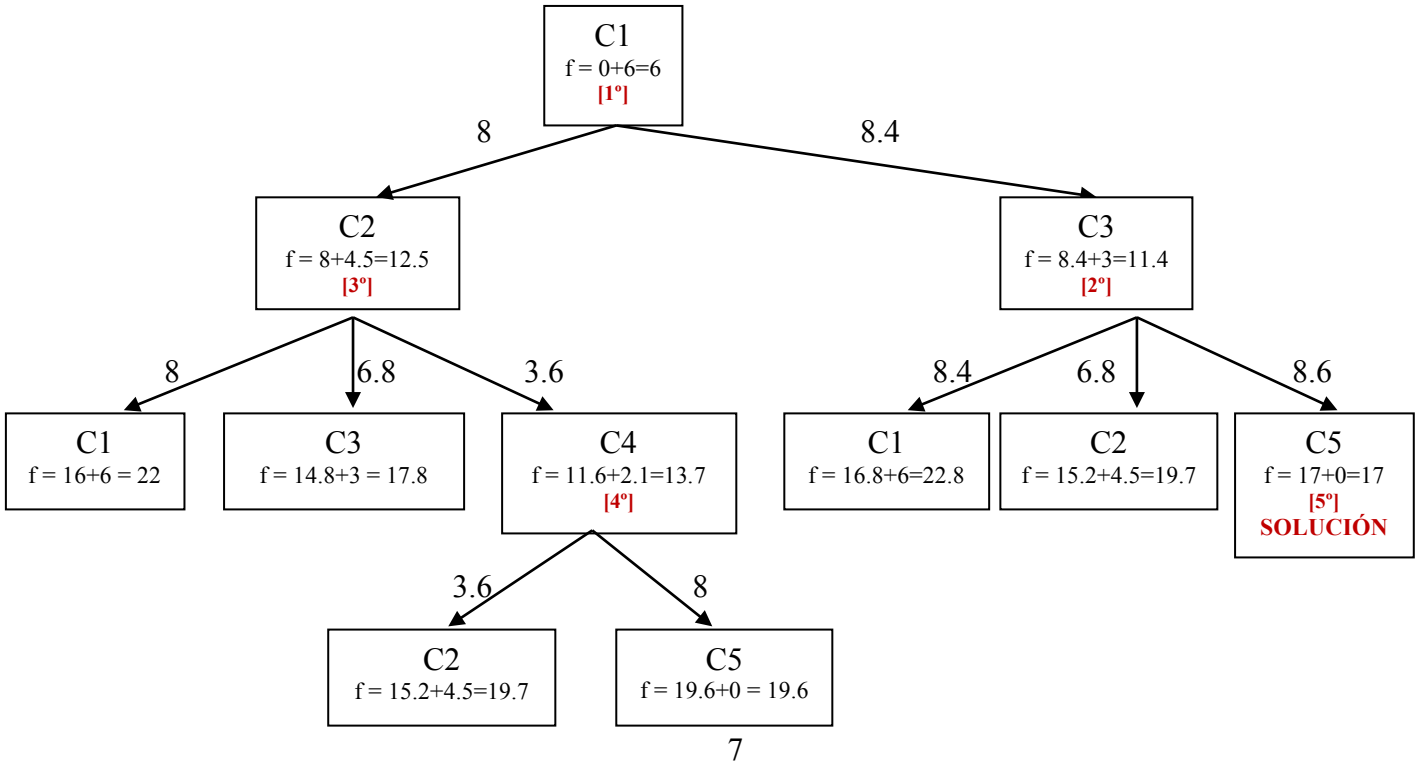
SOLUCIÓN:

h(n) = (distancia en línea recta a 5) · mínimo(5/100, 10/100) = (distancia en línea recta a 5)·5/100
Es admisible porque el verdadero coste h*(n) no puede ser menor que el coste de ir en línea recta a la ciudad objetivo con un gasto constante de 5 euros cada 100 kms (el menor) y sin ningún peaje.

Ciudad	Distancia a 5 en línea recta (kms)	h (euros)
1	120	120·5/100= 6
2	90	90·5/100 = 4.5
3	60	60·5/100 = 3
4	42	42·5/100 = 2.1
5	0	0

Todos los costes están en euros:

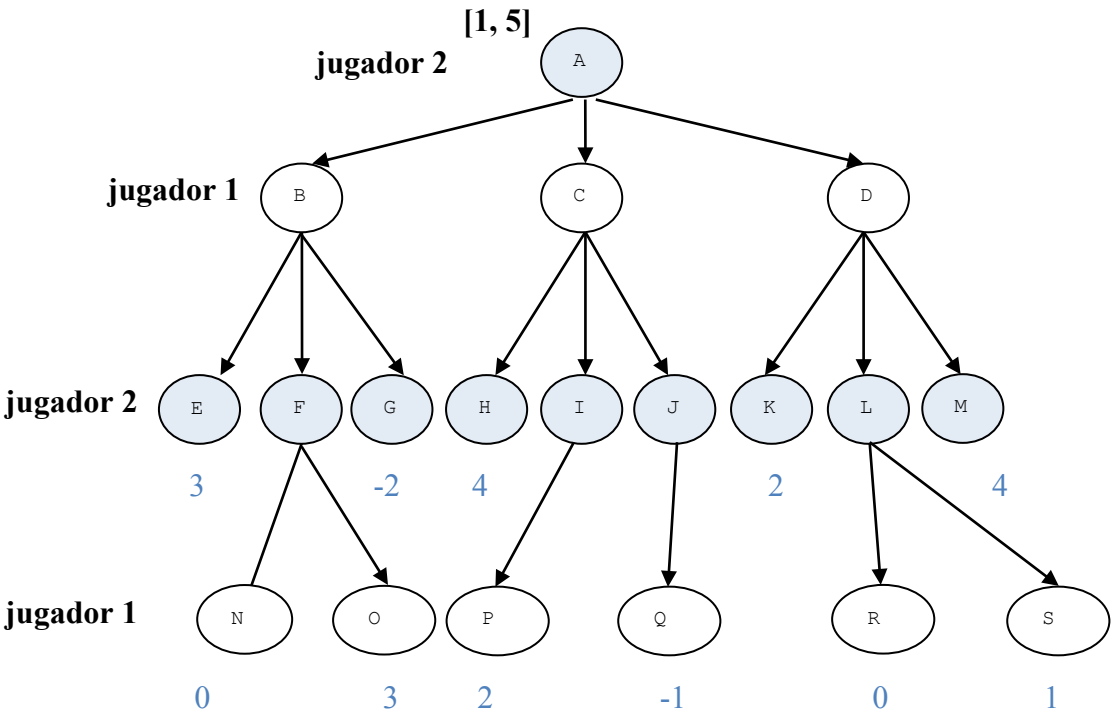




Se encuentra la solución C1 → C3 → C5.
Es óptima porque se ha usado A* con una heurística admisible y búsqueda-en-árbol (sin eliminar estados repetidos).
El coste total de esa solución es 8.4 + 8.6 = 17 euros.

4. Se está aplicando el MINIMAX con poda alfa-beta para analizar qué movimiento es el óptimo para el **jugador 1**, que tiene el turno (luego jugador 1 es **MAX**). En el proceso del análisis se llega al siguiente nodo, al cual llegan los valores alfa-beta [1,5]

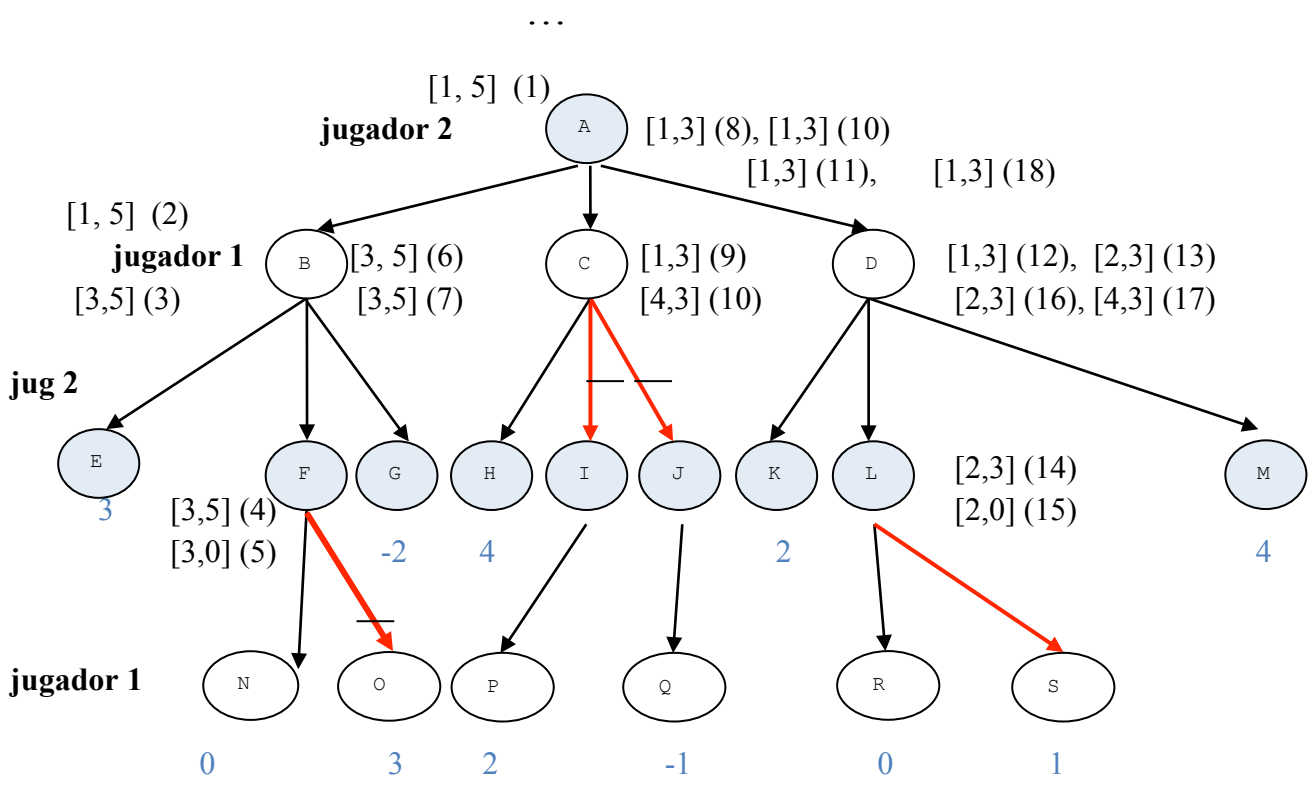
. . .



Inteligencia Artificial Hoja 1	Publicación: 2015/01/30 Límite entrega: 2015/02/10/11:00 Resolución en clase: 2015/02/10 (grupo mañana), 2015/02/11 (grupo tarde)

Realiza el árbol asociado al análisis MINIMAX con poda alfa-beta, señalando en cada paso los valores alfa-beta en cada momento.

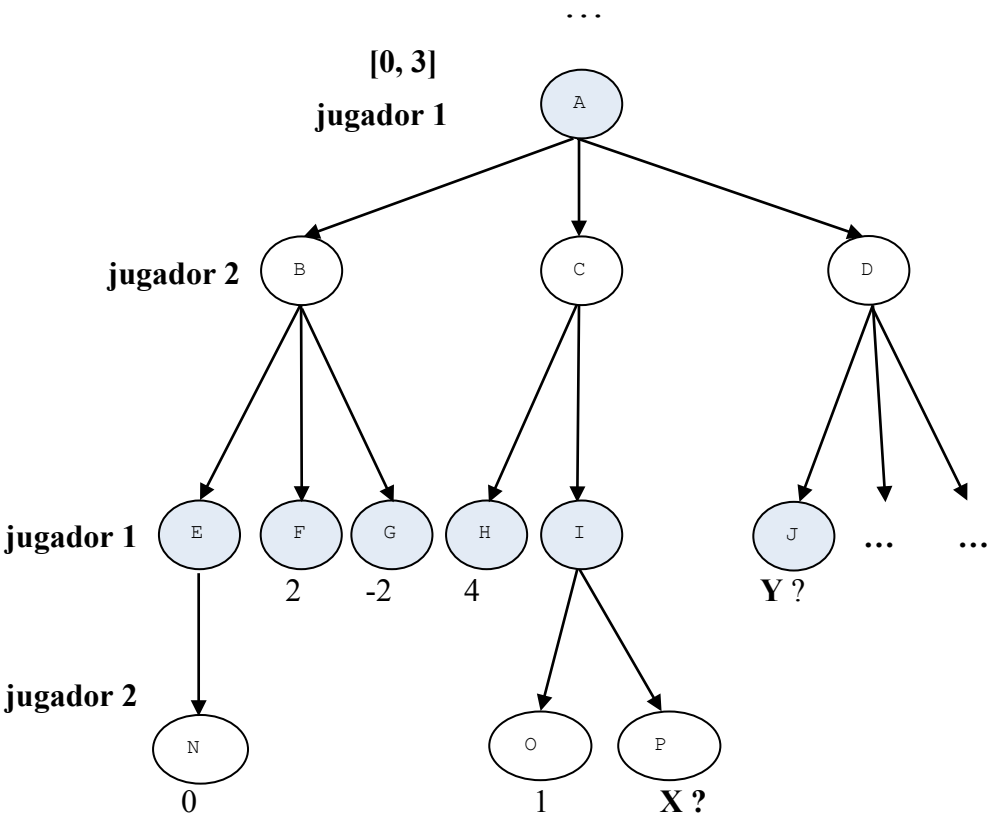
a) ¿Qué ramas del subárbol son podadas? Las marcadas en rojo.



b) ¿Qué valor de utilidad se propaga hacia arriba desde el nodo A?
3

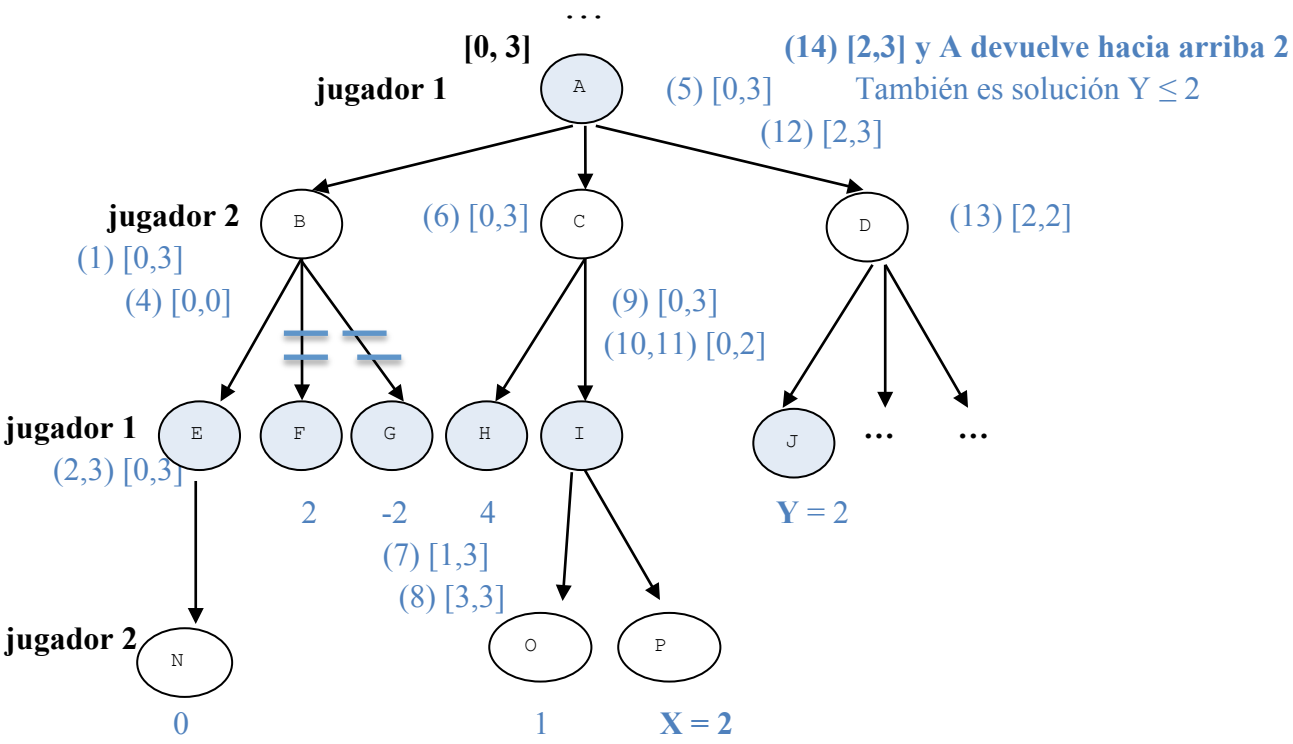
c) Cambia el valor de la utilidad del nodo terminal **K** de forma que sean podadas las tres ramas D-L-R, D-L-S y D-M.
El nodo **D** va a recibir el valor de la utilidad de **K,** lo que va a hacer cambiar su alfa ya que **D** es un nodo MAX. El intervalo alfa-beta en **D** justo antes de recibir la utilidad de **K** es [1, 3], por lo que basta que la utilidad de **K** sea 3 o mayor que 3 para activar la poda.

5. Se está aplicando el algoritmo MINIMAX con poda alfa-beta para analizar qué movimiento es el óptimo para el **jugador 1**, que tiene el turno (luego jugador 1 es **MAX**). En el proceso del análisis se llega al siguiente nodo, al cual llegan los valores alfa-beta [0,3]



¿Qué valores hay que dar a **X** e **Y** para que se llegue a explorar el nodo J pero se poden las ramas segunda, tercera etc. que cuelgan de D, y que al final del análisis se propague desde el nodo A un valor de 2?

Realiza el árbol asociado al análisis MINIMAX con poda alfa-beta, señalando claramente los valores alfa-beta en cada paso y las ramas que se van podando.


[0, 3]
jugador 1
A
(14) [2,3] y A devuelve hacia arriba 2
(5) [0,3]
También es solución Y ≤ 2
(12) [2,3]
jugador 2
B
(6) [0,3]
C
D
(13) [2,2]
(1) [0,3]
(4) [0,0]
(9) [0,3]
(10,11) [0,2]
jugador 1
(2,3) [0,3]
E
F
G
H
I
J
2
-2
4
Y = 2
(7) [1,3]
(8) [3,3]
jugador 2
N
O
P
0
1
X = 2

Inteligencia Artificial Hoja 1	Publicación: 2015/01/30 Límite entrega: 2015/02/10/11:00 Resolución en clase: 2015/02/10 (grupo mañana), 2015/02/11 (grupo tarde)

6. Consideremos el juego que se desarrolla en el siguiente tablero

Celda 1	Celda 2
Celda 3	Celda 4

Jugador I: Empresa de telecomunicaciones (elige frecuencias de celdas impares)
Jugador II: Teleterrorista (elige frecuencias de celdas pares)

El objetivo de la empresa de telecomunicaciones es diseñar una red de comunicaciones móviles basada en la segmentación del territorio en el que se desea proporcionar cobertura en celdas cuadradas (normalmente, en el mundo real, serían hexagonales). Los emisores / receptores de la señal electromagnética portadora de la información (voz y datos) necesaria para posibilitar la comunicación entre los interlocutores se colocan en los centros de cada una de las celdas. El problema es que nuestra compañía ha comprado los derechos para utilizar únicamente 3 bandas de frecuencia (centradas en 900, 1300 y 1800 MHz), que representaremos mediante colores: R, G y B, respectivamente. Para evitar interferencias, las frecuencias (colores) de celdas adyacentes deben ser diferentes.

El objetivo del teleterrorista es inutilizar la red de comunicaciones favoreciendo las interferencias. Desde el punto de vista de la empresa de telecomunicaciones, la puntuación del tablero cuando todas las celdas han recibido la asignación de frecuencia es el número de adyacencias en las que los colores de las celdas adyacentes no son coincidentes.

Suponiendo que abre el juego la empresa de telecomunicaciones eligiendo la frecuencia de la celda 1, y se continúa con la celda 2 (elige frecuencia el teleterrorista), la celda 3 (elige la empresa), etc.

(i) Formula este juego como un problema búsqueda entre adversarios.
(ii) ¿Es un juego de suma cero?
(iii) ¿Se puede utilizar el algoritmo MINIMAX para encontrar el movimiento inicial óptimo para la empresa de telecomunicaciones?
(iv) ¿y para los siguientes?
(v) En caso de la respuesta a (iii) sea positiva, determinar el movimiento inicial óptimo utilizando MINIMAX con poda alpha-beta. En caso de que dicha respuesta sea negativa, diseñar un algoritmo que garantice encontrar el movimiento inicial óptimo.

SOLUCIÓN:

(i) Formula este juego como un problema búsqueda entre adversarios.

Estado inicial: Celdas sin frecuencias asignadas
Jugadas: (1) Empresa asigna frecuencia a Celda 1
(2) Teleterrorista asigna frecuencia a Celda 2
(3) Empresa asigna frecuencia a Celda 3
(4) Teleterrorista asigna frecuencia a Celda 4
Estado final: Todas las celdas tienen las frecuencias asignadas
Función de utilidad: número de adyacencias en las que los colores de las celdas adyacentes no son coincidentes

(ii) ¿Es un juego de suma cero?
Sí. Los objetivos de la empresa y del teleterrorista son contrapuestos, por lo que el valor la utilidad en un estado terminal desde el punto de vista del teleterrorista es menos el valor de la utilidad desde el punto de vista de la empresa.

(iii) ¿Se puede utilizar el algoritmo MINIMAX para encontrar el movimiento inicial óptimo para la empresa de telecomunicaciones?
Sí.

(iv) ¿y para los siguientes?
Sí.

(v) En caso de la respuesta a (iv) sea positiva, determinar la secuencia óptima utilizando MINIMAX con poda alpha-beta. En caso de que dicha respuesta sea negativa, diseñar un algoritmo que garantice encontrar la secuencia de jugadas óptimas.

Solo es necesario explorar una de las jugadas (por ejemplo: la empresa asigna R a la celda 1).
Por simetría, las otras asignaciones conducen a configuraciones equivalentes desde el punto de vista de diseño para la red de telecomunicaciones. Las tres posibles asignaciones son óptimas.
En rojo se marcan las podas.
En verde en el árbol se marca la secuencia de jugadas óptima.