

# EXAMEN PRÁCTICAS INTELIGENCIA ARTIFICIAL CONVOCATORIA DE JUNIO

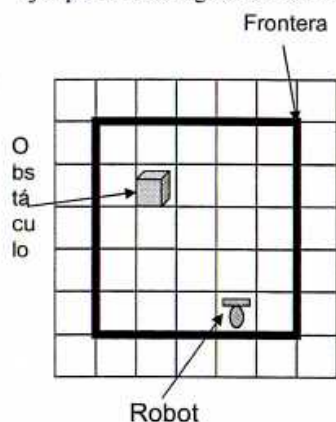
Segundo curso del Grado en Ingeniería Informática, Curso 2012-2013

Nombre:

DNI:

Grupo:

1. (3 puntos) Supongamos que un agente trabaja sobre un tablero formado por  $n \times n$  casillas. Sobre este tablero se definen dos zonas: una "zona interior" formada por un tablero de  $(n-2) \times (n-2)$  casillas inscrito en el tablero general, y una "zona exterior" formada por el resto de las casillas. Separando ambas zona aparece una línea gruesa negra denominada "Frontera". En la figura se muestra un ejemplo de la configuración de un tablero 7x7.



El cometido del robot consiste en llevar todos los obstáculos que se encuentren en la zona interior a la zona exterior. El robot siempre se debe encontrar en la zona interior, y no debe nunca traspasar la frontera.

Para realizar esta tarea, el robot dispone de 3 sensores, un sensor de choque "BUMPER" que le permite detectar el obstáculo, un sensor de infrarrojos "CNY70" que permite ver donde está la línea de la Frontera, y una brújula digital "Brújula" que le indica su orientación en el avance. Los dos primeros sensores se encuentran situados en la parte frontal del robot. La brújula sólo devuelve 4 valores: 0, 1, 2 y 3, representando respectivamente Norte, Este, Sur y Oeste.

Las acciones que puede realizar el robot son las siguientes:

- **Avanzar:** Avanza una casilla en la dirección que marca su brújula siempre que no tenga un obstáculo delante.
- **Retroceder:** Retrocede una casilla en la dirección contraria a la que indica su brújula, siempre que no tenga un obstáculo detrás.
- **Girar:** Gira sin moverse de la casilla en el sentido de la agujas del reloj.
- **Empujar:** Avanza una casilla en la dirección que marca su brújula. Para que esta acción tenga efecto, debe estar activado el sensor de choque.

Se pide:

- a) Definir las variables de estado y las reglas de producción necesarias para diseñar un agente reactivo con memoria que, partiendo de una casilla desconocida dentro de la zona interior de un tablero de dimensiones también desconocidas (nunca superiores a 100x100), sea capaz de encontrar el obstáculo y expulsarlo hacia la zona exterior.

- b) Añadir sobre el problema del apartado anterior, las variables de estado y las reglas de producción necesarias que permitan al robot expulsar el obstáculo hacia la zona exterior, arrastrándolo *por el camino más corto de casillas*.

2. (2 puntos) Considere el 8-puzzle cuyo estado inicial y estado meta se muestran en la siguiente figura:

2	4	6
1	3	5
8	7	

1	2	3
8		4
7	6	5

Desarrolle el árbol de búsqueda que expande el algoritmo de escalada por máxima pendiente, utilizando las siguientes heurísticas. El orden de aplicación de los movimientos es el siguiente: Arriba, Abajo, Izquierda, Derecha.

- a) Heurística 1:  $h1(n)$  = número de piezas mal colocadas.
- b) Heurística 2:  $h2(n)$  = suma de distancia Manhattan de cada pieza sobre su posición correcta. (Nota: La distancia Manhattan de una pieza es la suma de las distancias vertical y horizontal a su posición final.)

Indique en el desarrollo del árbol junto al tablero el valor de la heurística. En caso, de dos tableros con la misma heurística se expande el que primero se generó.

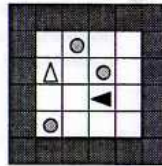
3. (2 puntos) Implementar el algoritmo MINIMAX en C++. Para su implementación, suponer que existen dos funciones:

- a) **double Valoration (const state & tablero, int jug)** que devuelve la valoración de un tablero desde el punto de vista de *jug* y
- b) **state Next\_Child(const state & tablero)** que genera el siguiente hijo, aún no generado, del tablero que se pasa como argumento.

Se puede suponer además la existencia de una constante **const state NULO** que es el valor que devuelve la función *Next\_Child* cuando ya no tiene más descendientes no generados del tablero que se pasa como argumento.

Si con la información anterior no fuese posible hacer la implementación, añadir tantas suposiciones como se crean necesarias.

4. (3 puntos) En la siguiente figura se muestra una situación concreta en el juego de las aspiradoras. El juego consiste en que dos aspiradoras (una blanca y otra negra) que compiten por llevarse la mayor cantidad de suciedad disponible en el tablero.



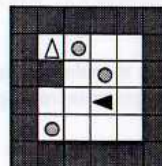
Los elementos que aparecen en el tablero son tres:

- Las aspiradoras, una blanca y otra negra.
- La suciedad, representada por casillas con un círculo inscrito.
- Obstáculos, marcados como casillas oscuras y representan casillas que no puede ocupar una aspiradora.

Las tres acciones que puede realizar cada aspiradora son las siguientes:

- **Avanzar:** la aspiradora avanza a la casilla adyacente según indica su orientación. El avance sólo es posible si la casilla a la que va a acceder no está ocupada por un obstáculo.
- **Girar\_Derecha:** la aspiradora se mantiene en su posición y cambia su orientación de avance hacia la derecha.
- **Girar\_Izquierda:** la aspiradora se mantiene en su posición y cambia su orientación de avance hacia la izquierda.

Como situación especial, cada vez que una aspiradora avanza, deja en la casilla origen un obstáculo, dejando dicha casilla como no transitable. Así, por ejemplo, el efecto de que la aspiradora de la figura anterior avance, el tablero resultante es el siguiente:



Se pide:

- Construir el árbol del juego hasta profundidad 4 (es decir, desde el nivel 0 hasta nivel 3), tomando este último tablero como nodo raíz, y suponiendo que el primero en jugar es la aspiradora negra.
- La función de valoración en los nodos hoja es la siguiente:

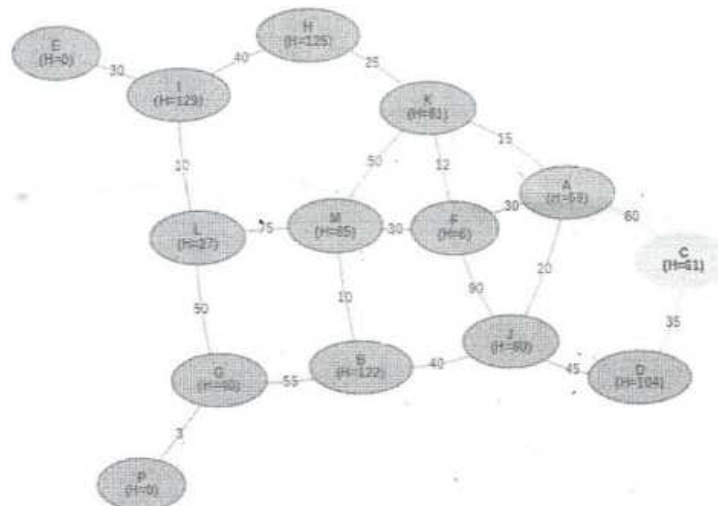
$$M(\text{Aspiradora Negra}) - M(\text{Aspiradora Blanca}) + Dsc(\text{Aspiradora Blanca})$$

Siendo  $M(x)$  una función que mide la suciedad obtenida hasta el momento por la aspiradora  $x$  que se pasa como argumento y  $Dsc(x)$  es la distancia de Manhattan de la aspiradora pasada como argumento a la casilla más cercana con suciedad.

- Calcular el valor MINIMAX de la raíz del árbol construido usando la poda ALFA-BETA. Se debe marcar con claridad que nodos son podados al realizar el proceso.

Inteligencia Artificial  
Departamento de Ciencias de la Computación e Inteligencia Artificial  
Curso 2016/2017 (Convocatoria de Junio)  
Examen de Ejercicios

1. Supongamos que el siguiente grafo no dirigido representa un espacio de estados de un problema real. Cada arco es de doble dirección, por ejemplo,  $A$  es padre de  $F$  y  $F$  es padre de  $A$ . El número dentro de cada nodo representa el valor de la función  $h$  del nodo y el número sobre el arco el coste del operador. El único nodo de inicio es  $C$ . Los nodos objetivos son  $E$  y  $P$ .



Debes probar distintos algoritmos de búsqueda sobre el problema anterior. En todos los casos ABIERTOS representará el conjunto de nodos que pueden ser explorados en la siguiente iteración del algoritmo de búsqueda. Cuando no importe el coste o la heurística, o cuando importe pero haya un empate, **utilizar siempre el orden alfabético para seleccionar los nodos.**

Para cada algoritmo se pedirán tres datos:

- El conjunto de nodos expandidos en el orden que salen de ABIERTOS.
- El objetivo alcanzado.
- El camino solución encontrado.

(a) Algoritmo 1: Búsqueda en Anchura (1 punto)

Secuencia de nodos:	
Estado objetivo alcanzado:	
Camino solución:	

(b) Algoritmo 2: Búsqueda en Profundidad (1 punto)

Secuencia de nodos:	
Estado objetivo alcanzado:	
Camino solución:	

(c) Algoritmo 3: Búsqueda Costo Uniforme (2 puntos)

Secuencia de nodos:	
Estado objetivo alcanzado:	
Camino solución:	

(d) Algoritmo 4: Algoritmo A\* (3 puntos)

Secuencia de nodos:	
Estado objetivo alcanzado:	
Camino solución:	



2. (3 puntos) En la siguiente página se muestra un árbol de búsqueda para un determinado juego. Los nodos blancos son nodos interiores del árbol, mientras los nodos grises representan estados en los que se debe aplicar la función heurística. Como se puede observar, estos últimos tienen asociado un número. Este número no es su valor heurístico, sino que simplemente es un identificador.

Por otro lado, en la parte inferior de la página, aparece una secuencia (Nodo, Valor). La parte de "Valor" está compuesta por una secuencia de valores para los nodos evaluables del árbol que corresponde con las distintas valoraciones heurísticas. Sin embargo, la parte "Nodo" está vacía.

La idea del ejercicio es aplicar el algoritmo de poda alfa-beta a dicho árbol sabiendo que sólo las hojas no podadas toman valores heurísticos de la secuencia inferior, y que dichos valores se asignan a los nodos secuencialmente en el orden en el que estos son evaluados.

Ejemplo: Supongamos un árbol con 9 nodos hoja, numerados del 1 a 9, y que tienen asociados la siguiente secuencia "Nodo"/"Valor":

Nodo									
Valor	5	7	-3	9	13	8	12	9	23

Empezamos a aplicar el método de la poda alfa-beta y llegamos a la primera hoja y ésta no se puede podar. Entonces se le asigna a dicha hoja el primer valor de la secuencia, el 5, y lo anotamos en la lista de la siguiente forma:

Nodo	1								
Valor	5	7	-3	9	13	8	12	9	23

Supongamos que la hoja 2 y 3 se podan, pero la 4 no. En este caso anotamos que el 7 es para la hoja 4:

Nodo	1	4							
Valor	5	7	-3	9	13	8	12	9	23

Este proceso se aplica sucesivamente hasta que se resuelva el árbol. Supongamos que el resultado final es el siguiente:

Nodo	1	4	5	8	9				
Valor	5	7	-3	9	13	8	12	9	23

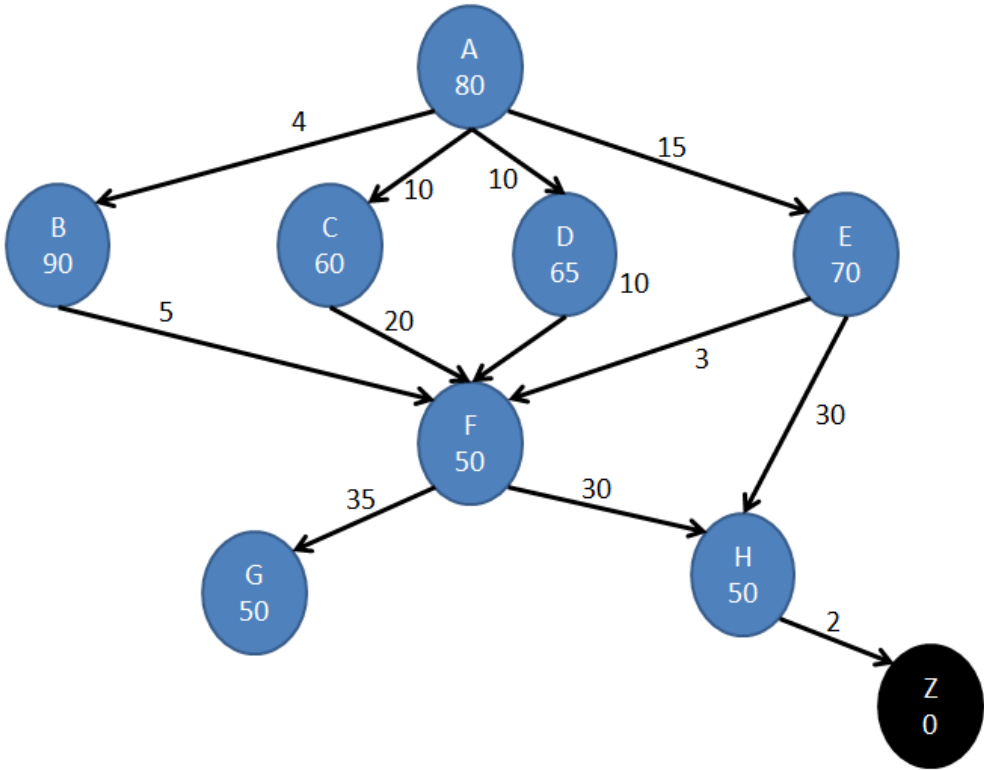
Como se puede observar, no todos los valores heurísticos han sido utilizados, ya que los nodos 2, 3, 6, 7, 7 han sido podados en el proceso, y la lista refleja los valores heurísticos asignados a los nodos podados.

Se pide resolver el árbol usando el algoritmo minimax con poda-alfa beta **completando la lista "Nodo"/"Valor"** resultante de su resolución donde aparezca la asignación de la función heurística a cada uno de los nodos hojas no podados, así como el **valor Minimax** asociado a la resolución del árbol.



Examen de ejercicios (Parte práctica)

1. Dado el siguiente grafo con los costes, determina:



Donde Z es el nodo objetivo, el valor que se encuentra dentro de cada nodo es la heurística y el coste para cada camino es el que se encuentra en cada unión.

Para A el valor de la heurística es 80, y el camino para ir a E cuesta 15

a) Búsqueda en anchura (1 punto)

Secuencia de cerrados	
Secuencia resultado	
Coste del camino	

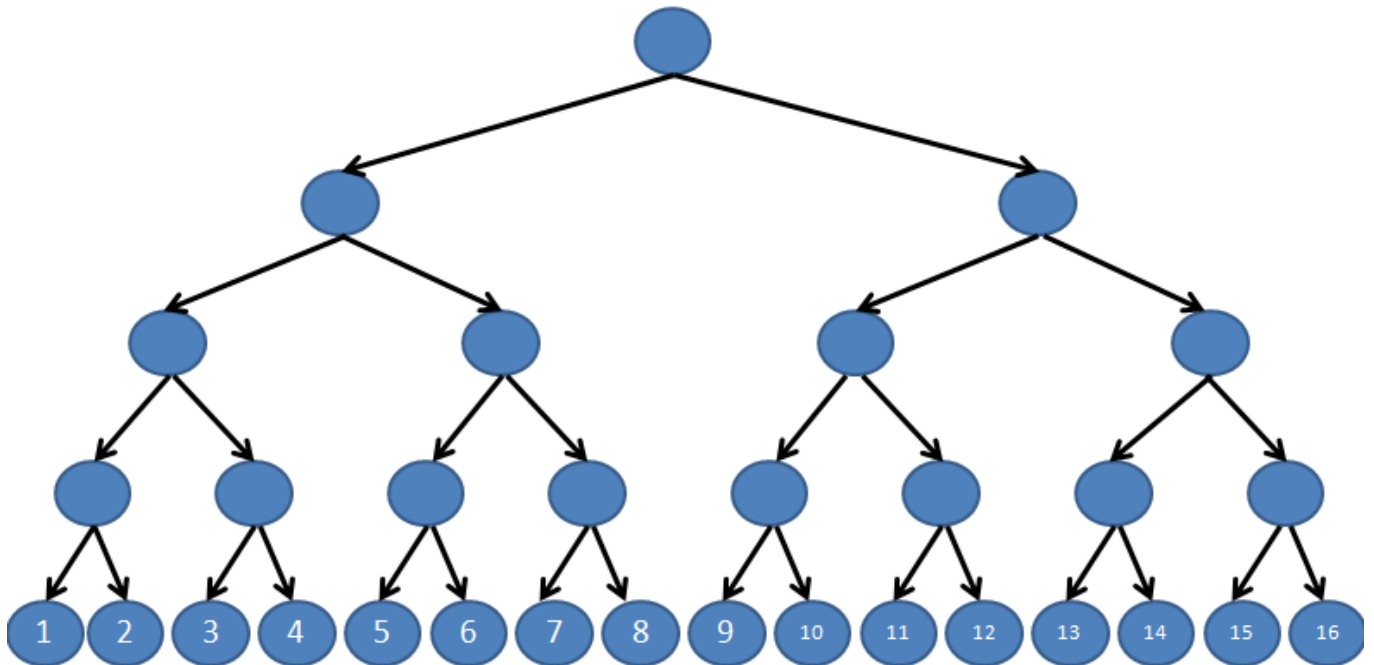
b) Coste uniforme (2 puntos)

Secuencia de cerrados	
Secuencia resultado	
Coste del camino	

c) Algoritmo A\* (3 puntos)

Secuencia de cerrados	
Secuencia resultado	
Coste del camino	

2. Dado el siguiente árbol, devuelve el resultado minimax, tras aplicar el algoritmo de poda Alfa-Beta (4 puntos)



Rellene la tabla con los nodos según el valor correspondiente:

Nodo																	
Valor	8	-1	6	2	8	4	-5	-10	20	5	4	6	4	9	-10	-3	1

El valor minimax es:



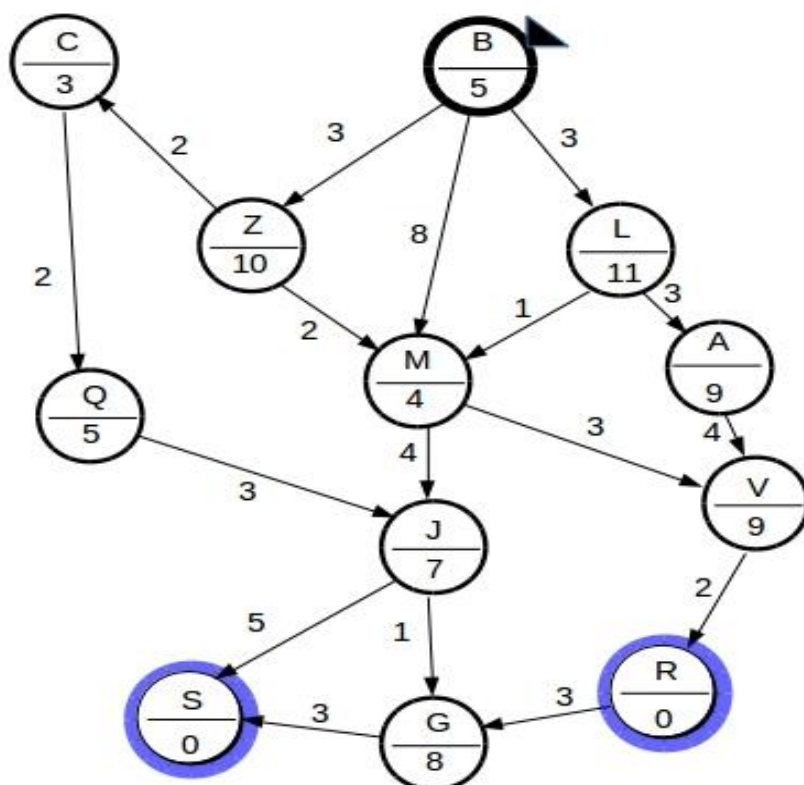


**Inteligencia Artificial**  
**Departamento de Ciencias de la Computación e Inteligencia Artificial**  
**Curso 2019/2020 (Relación de Problemas Entregable)**

Nombre:	-		
DNI:	-		
Grupo:	-	Subgrupo Prácticas	-

**Se deberá subir una copia de este documento completada y en formato PDF en PRADO desde el 4 de junio a partir de las 10:00 y con fecha límite el 6 de junio a las 10:00.**

**Ejercicio 1:** (6 Puntos) Supongamos que el siguiente grafo representa un espacio de estados de un problema real. El número dentro de cada nodo representa el valor de la función  $h$  del nodo y el número sobre el arco el coste del operador. El único nodo de inicio es **B**. Los nodos objetivos son **R** y **S**.



Usar los algoritmos de búsqueda especificados más abajo para resolver el problema anterior. En todos los casos ABIERTOS representará el conjunto de nodos que pueden ser explorados en la siguiente iteración del algoritmo de búsqueda. Y por supuesto, como en todos los algoritmos de búsqueda, el algoritmo termina cuando un nodo objetivo satisface la condición de terminación del propio algoritmo, y en ningún caso es necesario obtener todos los nodos objetivos.

**Importante:** En la búsqueda en anchura los nodos se deben expandir siguiendo el orden alfabético (por ejemplo, los sucesores del nodo B son L, M y Z en ese orden). En los otros dos algoritmos

cuando se presente alguna situación de empate se aplicará de nuevo el orden alfabético para seleccionar los nodos. Esto es fundamental para que el ejercicio se pueda valorar correctamente, de no usar el criterio, los problemas no estarán bien resueltos.

Para cada algoritmo se pedirán cuatro datos:

- Secuencia de nodos: corresponde a la secuencia de nodos en el orden en el que salen de ABIERTOS.
- El objetivo alcanzado.
- El camino solución encontrado: secuencia de nodos del camino empezando en B.
- El coste de la solución encontrada. En el algoritmo de búsqueda en anchura no se usa el coste, pero el camino final tiene un coste, reflejarlo en la casilla correspondiente. Los otros dos algoritmos si lo usan.

Algoritmo 1: Búsqueda en Anchura (1 punto)

Secuencia de nodos	B L M Z A J V C G S
Estado objetivo alcanzado	S
Camino solución	B M J S
Coste de la solución	17

**Insertar aquí** documentación gráfica de la resolución manual:

Búsqueda en anchura

<u>Abiertos</u>	<u>Cerrados</u>
<del>(B, -)</del>	(B, -)
<del>(L, B)</del>	(L, B)
<del>(M, B)</del>	(M, B)
<del>(Z, B)</del>	(Z, B)
<del>(A, L)</del>	(A, L)
<del>(J, M)</del>	(J, M)
<del>(V, M)</del>	(V, M)
<del>(C, Z)</del>	(C, Z)
<del>(G, J)</del>	(G, J)
<del>(S, J)</del>	(S, J) ← solución
(R, V)	
(Q, C)	

Algoritmo 2: Búsqueda Costo Uniforme (2 puntos)

Secuencia de nodos	B L Z M C A Q V J G R
Estado objetivo alcanzado	R
Camino solución	B L M V R
Coste de la solución	9

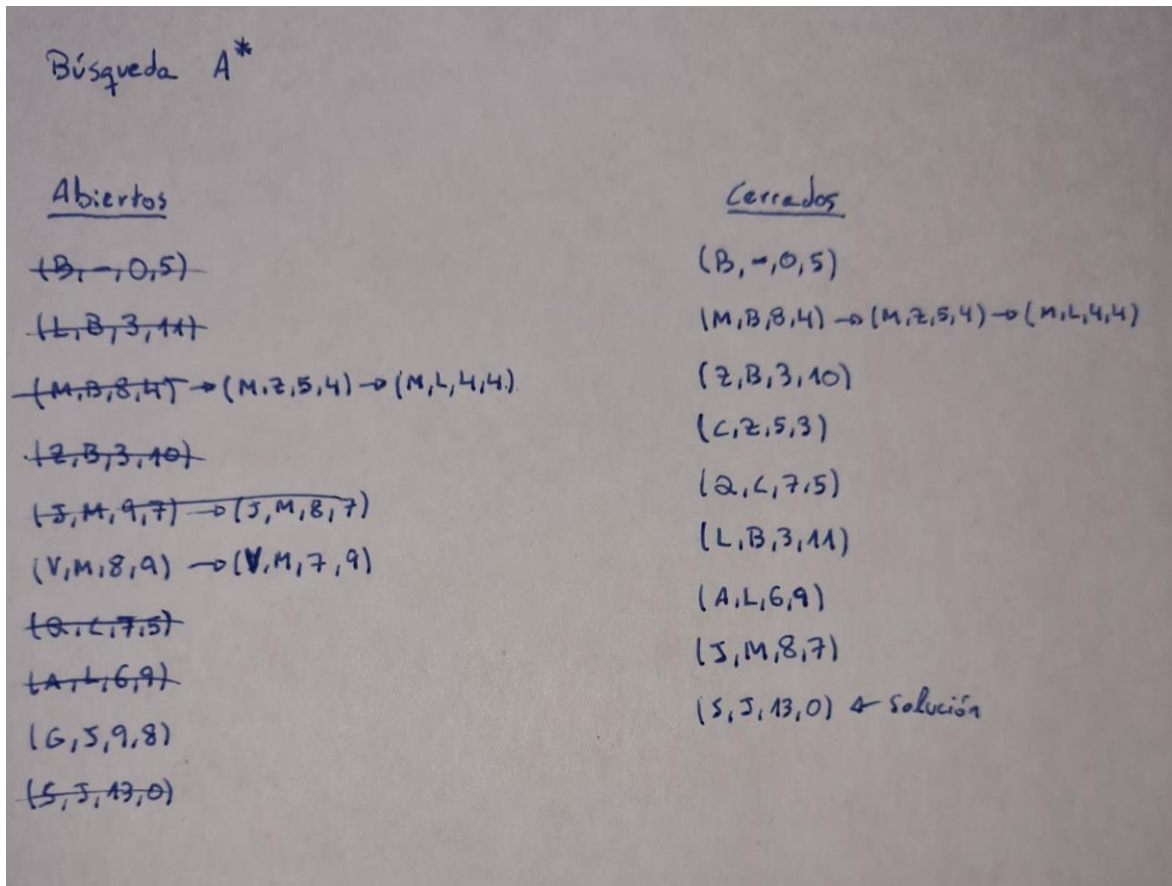
**Insertar aquí** documentación gráfica de la resolución manual:

<u>Abiertos</u>	<u>Cerrados</u>	Búsqueda costo uniforme {nodo, padre, coste acumulado}
<del>(B, -, -)</del>	(B, -, -)	
<del>(L, B, 3)</del>	(L, B, 3)	
<del>(M, B, 8) → (M, L, 4)</del>	(Z, B, 3)	
<del>(Z, B, 3)</del>	(M, L, 4)	
<del>(A, L, 6)</del>	(L, Z, 5)	
<del>(L, Z, 5)</del>	(A, L, 6)	
<del>(J, M, 8)</del>	(Q, L, 7)	
<del>(V, M, 7)</del>	(V, M, 7)	
<del>(Q, L, 7)</del>	(J, M, 8)	
<del>(R, V, 9)</del>	(G, J, 9)	
<del>(G, J, 9)</del>	(R, V, 9)	
<del>(S, J, 13) → (S, G, 12)</del>		

Algoritmo 3: Búsqueda A\* (3 puntos)

Secuencia de nodos	B M Z C Q L A J S
Estado objetivo alcanzado	S
Camino solución	B L M J S
Coste de la solución	13

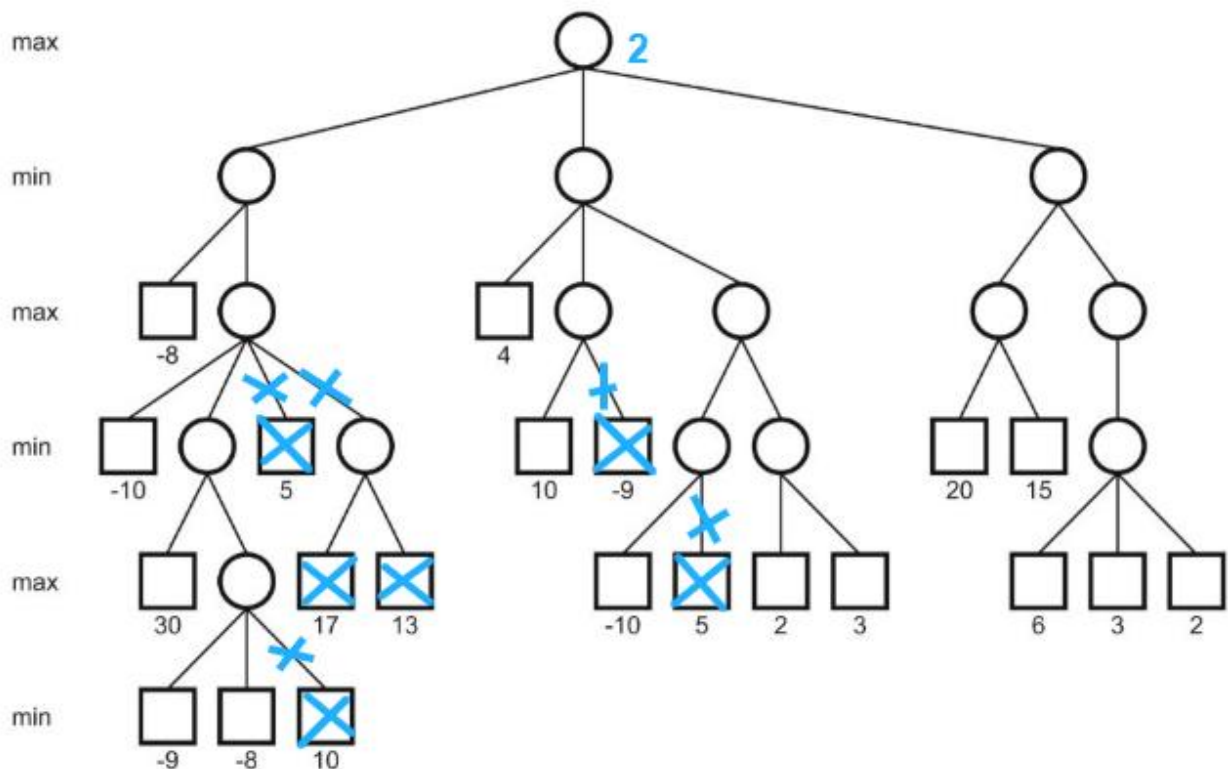
**Insertar aquí** documentación gráfica de la resolución manual:





**Ejercicio 2:** (4 puntos) En la siguiente página se muestra un árbol de búsqueda para un determinado juego. Las capas de nodos MAX y MIN están especificadas en el lateral izquierdo de la imagen. Los nodos en forma de círculo representan los nodos interiores del árbol. Los nodos en forma de rectángulo representan estados terminales del juego y en los que se debe aplicar la función de evaluación estática o heurística. El valor heurístico asociado a cada nodo terminal se indica debajo de dicho nodo.

Se pide resolver el juego usando el algoritmo de poda alfa-beta. Es muy importante que en su resolución aparezcan claramente marcados los nodos terminales podados (con forma de rectángulo) (**marcar con una "X" el interior del recuadro de aquellos nodos que no se evaluarán mediante su función heurística**), así como el **valor Minimax** asociado al nodo de inicio, indicar también las **podas que se hayan producido**, si hubiese alguna, marcando con una cruz sobre los arcos podados.



Indicar aquí el valor minimax obtenido= 2



**Curso 2021/22**

# **INTELIGENCIA ARTIFICIAL**

## **Relación de Problemas 1**

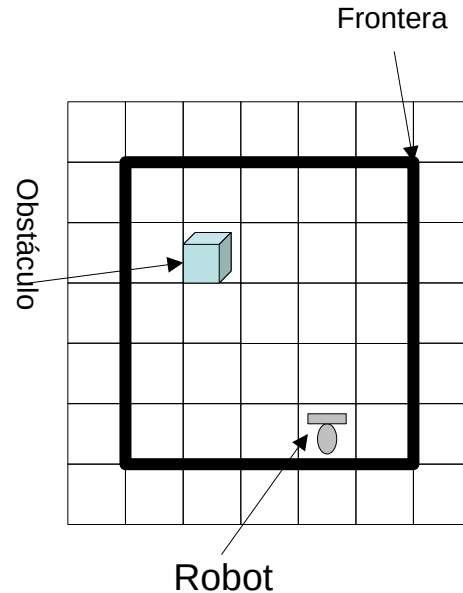
### **AGENTES REACTIVOS**

- 1.** Una hormiga artificial vive en un mundo bidimensional cuadriculado y desarrolla un comportamiento que le permite seguir un rastro de feromonas a lo largo de un conjunto de casillas previamente marcadas (el tamaño del rastro es de una casilla). La hormiga ocupa una sola casilla y puede encarar las casillas que se encuentran arriba, a la derecha, a la izquierda y debajo de la posición en la que se encuentra. La hormiga puede llevar a cabo tres acciones: moverse a una celda hacia adelante (actFORWARD), girar a la izquierda permaneciendo en la misma casilla (actTURN\_L) y girar a la derecha permaneciendo en la misma casilla (actTURN\_R). La hormiga puede percibir si la casilla que tiene delante (en el sentido del movimiento) tiene feromona.
  - a) Especificar un sistema de reglas para controlar el comportamiento de la hormiga en el seguimiento del rastro de la feromona. Suponer inicialmente a la hormiga en una casilla en la que puede percibir el rastro de feromona.
  - b) Resuelva el problema anterior con la siguiente restricción: la hormiga, una vez que llega a una nueva casilla, no puede girar más de 180 grados en el mismo sentido de giro.
- 2.** La avispa hembra del género *Sphex*, deja sus huevos dentro de un grillo que ha paralizado y ha llevado a su nido. Las larvas de la avispa salen del grillo y se alimentan de él. La avispa presenta el siguiente comportamiento: lleva el grillo paralizado a su nido, lo deja en el umbral del nido, entrar dentro del nido para ver si todo está correcto, sale, y entonces arrastra al grillo hacia su interior. Si el grillo se mueve cuando la avispa está en el interior haciendo la inspección preliminar, la avispa saldrá del nido, volverá a colocar el grillo en el umbral, pero no dentro, y repetirá el procedimiento de entrar en el nido para ver si todo está correcto. Si el grillo se mueve otra vez mientras la avispa está dentro del nido, ésta volverá a salir y colocar el grillo en el umbral, entrando de nuevo en el nido para realizar la inspección preliminar. En una ocasión, este procedimiento se repitió cuarenta veces. Define características y acciones para diseñar un agente reactivo que se corresponda con el comportamiento de la avispa.

3. Supongamos que un agente trabaja sobre un tablero formado por  $N \times N$  casillas. Sobre este tablero se definen dos zonas: una "zona interior" formada por un tablero de  $(N-2) \times (N-2)$  casillas inscrito en el tablero general, y una "zona exterior" formada por el resto de las casillas. Separando ambas zona aparece una línea gruesa negra denominada "Frontera". En la figura se muestra un ejemplo de la configuración de un tablero  $7 \times 7$ .

El cometido del robot consiste en llevar todos los obstáculos que se encuentren en la zona interior a la zona exterior. El robot siempre se debe encontrar en la zona interior, y no debe nunca traspasar la frontera.

Para realizar esta tarea, el robot dispone de 3 sensores, un sensor de choque "**BUMPER**" que le permite detectar el obstáculo, un sensor de infrarrojos "**CNY70**" que permite ver dónde está la línea de la Frontera, y una brújula digital "**Brujula**" que le indica su orientación en el avance. Los dos primeros sensores se encuentran situados en la parte frontal del robot. La brújula sólo devuelve 4 valores: 0, 1, 2 y 3, representando respectivamente Norte, Este, Sur y Oeste.



Las acciones que puede realizar el robot son las siguientes:

**Avanzar:** Avanza una casilla en la dirección que marca su brújula siempre que no tenga un obstáculo delante.

**Retroceder:** Retrocede una casilla en la dirección contraria a la que indica su brújula, siempre que no tenga un obstáculo detrás.

**GirarI:** Gira sin moverse de la casilla hacia la izquierda.

**GirarD:** Gira sin moverse de la casilla hacia la derecha.

**Nada:** No realiza ninguna acción

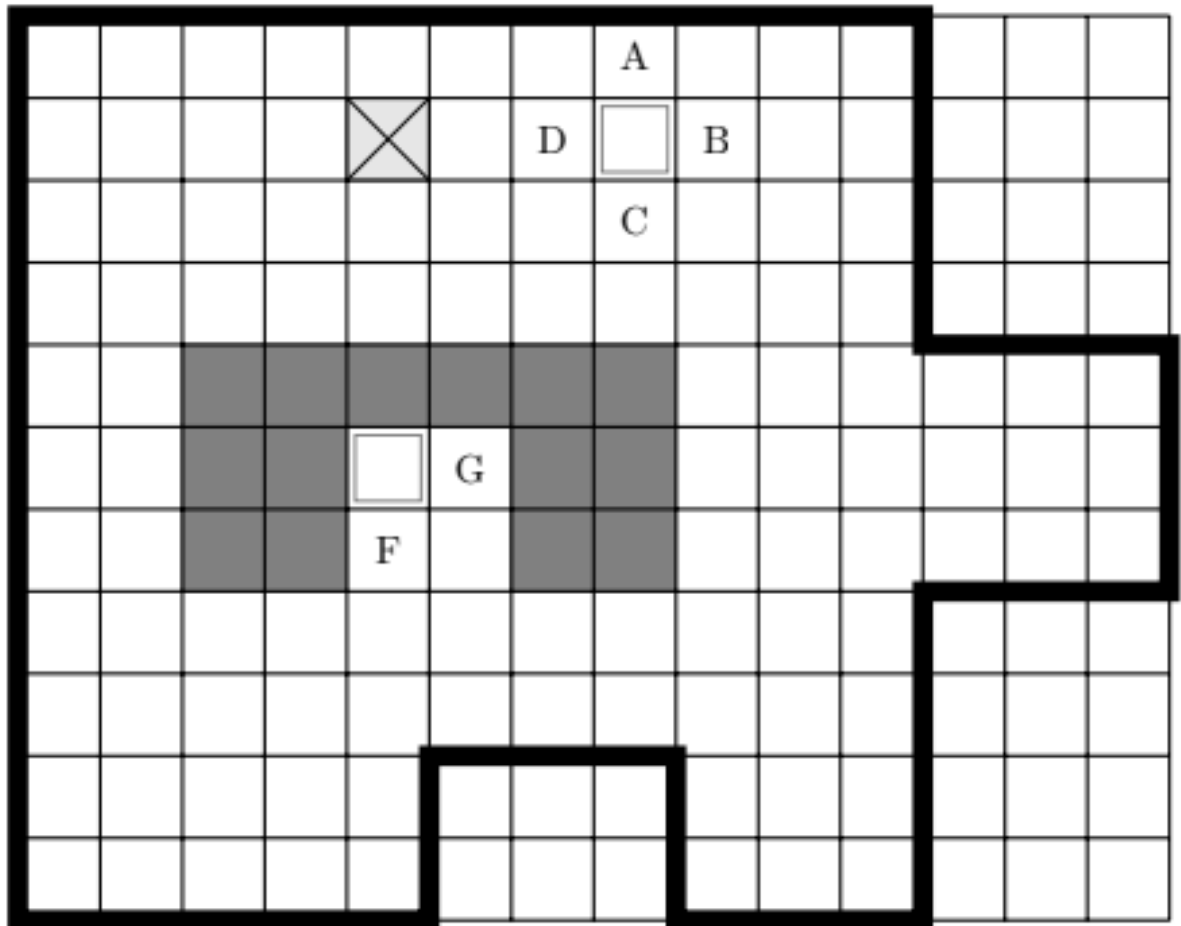
**Empujar:** Avanza una casilla en la dirección que marca su brújula. Para que esta acción tenga efecto, debe estar activado el sensor de choque.

Se pide:



- a) Definir las variables de estado (**nombre e descripción**) y las reglas de producción necesarias para diseñar un agente reactivo con memoria que partiendo de una casilla desconocida dentro de la zona interior de un tablero de dimensiones también desconocidas (nunca superiores a 99x99), sea capaz de calcular la dimensión de la zona interior, suponiendo que en el tablero no hay obstáculos.
  - b) Definir las variables de estado y las reglas de producción necesarias que permitan al robot localizar el obstáculo en el tablero.
  - c) Suponiendo que el robot se encuentra orientado hacia el obstáculo en una casilla adyacente (es decir, el sensor BUMPER está activado) y que el obstáculo se encuentra en una casilla interna del tablero que no es adyacente con ninguna casilla pegada a la frontera, definir las variables de estado y las reglas de producción necesarias que permitan al robot expulsar el obstáculo hacia la zona exterior, arrastrándolo por el camino más corto de casillas.
- 4.** Supongamos que tenemos un robot sobre un mapa bidimensional discreto de tamaño  $N \times M$ . El robot puede realizar las acciones de **Avanzar** y **Girar** en el sentido de las agujas del reloj. El robot posee un sistema de posicionamiento sobre el mapa que le devuelve sus coordenadas absolutas “(**robotX**, **robotY**)” dentro del mapa.
- Suponiendo que en el mapa hay obstáculos fijos (paredes), y que el robot se encuentra ubicado dentro de ese mapa en una posición concreta, definir un comportamiento reactivo para el mismo que le permita desplazarse hasta una coordenada objetivo “(**ObjX**, **ObjY**)”. Para ello, definir las variables de estado necesarias y el sistema de reglas de producción que reproducen el comportamiento requerido.

5. Idear una función de potencial artificial (con componentes repulsivos y atractivos) que pueda ser utilizada para guiar un robot desde cualquier casilla del mundo bidimensional cuadriculado de la figura siguiente, a la casilla objetivo que está marcada con una X (suponer que las posibles acciones que puede ejecutar el robot son ir al norte, sur, este y oeste). ¿Tienen las componentes repulsivas y atractivas algún mínimo local? Si es así, ¿dónde?







---

# INTELIGENCIA ARTIFICIAL

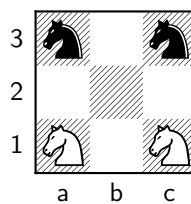
Curso 2021/22

Relación de Problemas 2

BÚSQUEDA

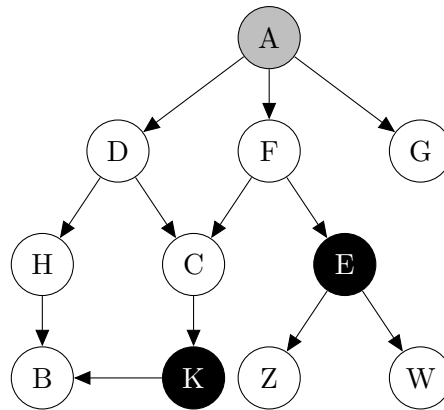
---

1. Tenemos un tablero de  $3 \times 3$  casillas como el siguiente:



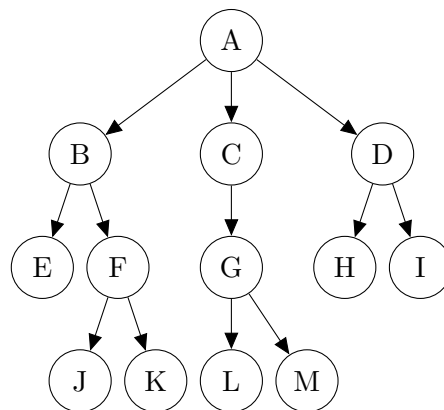
Deseamos intercambiar los caballos negros con los blancos.

- (a) Define que elementos forman:
- el estado inicial
  - le estado final
  - los operadores
- (b) ¿Importa el camino o sólo el estado final?
- (c) ¿La solución ha de ser óptima?
2. El grafo que se muestra a continuación determina un problema de búsqueda. Cada nodo representa un estado, mientras que los arcos modelan la aplicación de los operadores. Si A es el estado inicial y K y E son los estados meta:
- (a) Desarrolle el árbol de búsqueda en anchura
- (b) ¿Cuál de los nodos meta se expande primero?
- (c) La búsqueda en profundidad se diferencia de la búsqueda en anchura en que al expandir un nodo, los nodos hijo se insertan al inicio de la lista. Resuelva el problema usando este algoritmo.



3. Liste el orden en el que son visitados los nodos del árbol para cada una de las siguientes estrategias de búsqueda<sup>1</sup>. ¿Cuál es el número máximo de nodos que se ha requerido almacenar simultáneamente en memoria en cada caso?

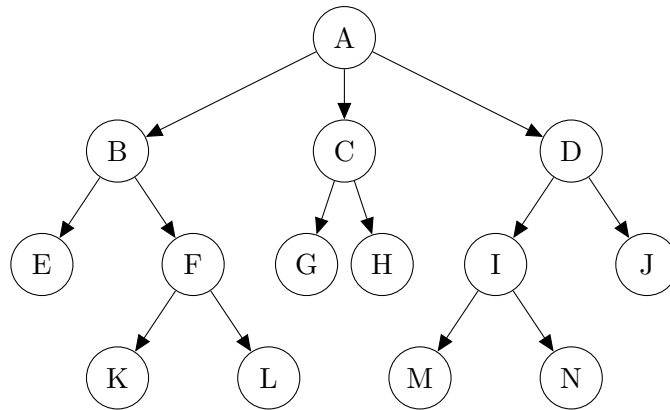
- (a) Búsqueda primero en anchura.
- (b) Búsqueda en profundidad.
- (c) Búsqueda retroactiva.
- (d) Descenso iterativo.



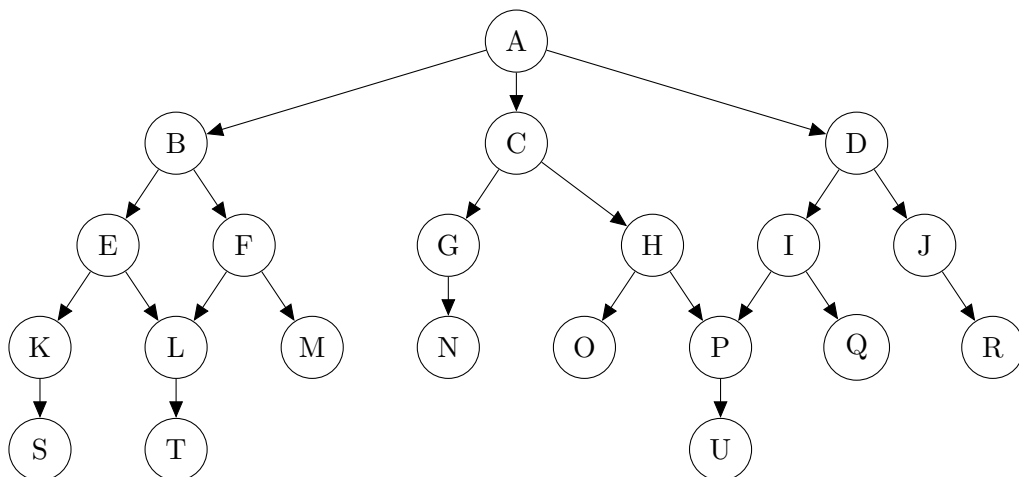
4. Dar el orden de búsqueda para los nodos mostrados en la siguiente figura para:

- (a) Búsqueda en anchura.
- (b) Búsqueda retroactiva con profundidad limitada ( $d = 2$ ).
- (c) Descenso iterativo (profundidad inicial = 1).

<sup>1</sup>Eligiendo siempre el nodo más a la izquierda



5. Ejecutar el algoritmo de búsqueda retroactiva sobre el grafo de la figura:

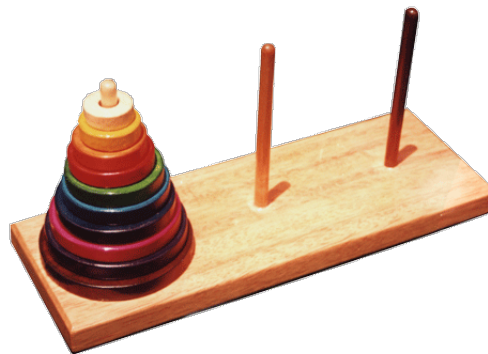


6. En la orilla de un río hay 3 misioneros y 3 caníbales. Todos ellos pretenden cruzar al otro lado. La barca que se utiliza para cruzarlo sólo tiene capacidad para dos personas, con lo que alguien ha de estar volviendo siempre a la orilla inicial mientras quede gente sin cruzar. Además, si en alguna ocasión y en cualquiera de las orillas se encuentran un número mayor de caníbales que de misioneros, los primeros se comerían a los segundos.

- ¿Cómo representarías los estados?
  - ¿Cuáles serían los operadores?
  - Dibujar el espacio de estados.
7. En una mesa se encuentran dos jarras, una con capacidad para 3 litros (llamada Tres), y la otra con capacidad para 4 litros (llamada Cuatro). Inicialmente, Tres y Cuatro están vacías. Cualquiera de ellas puede llenarse con agua del grifo G. Asimismo, el contenido de las jarras se puede vaciar en una pila P. También es posible verter el agua de una jarra en la otra. No se dispone de dispositivos de medición adicionales. Se trata de encontrar una secuencia de operadores que deje exactamente dos litros de agua en Cuatro.



- (a) Representar este problema como un problema de búsqueda
  - (b) Definir un estado inicial, el conjunto de estados meta, los operadores, así como el coste de cada operador.
  - (c) Encontrar una solución al problema utilizando un algoritmo de búsqueda.
8. “Las torres de Hanoi” es un juego matemático ideado en el siglo XVIII. Este juego consiste en pasar 64 discos de diámetro decreciente, de un poste a otro poste, utilizando un tercer poste auxiliar para los pasos intermedios.



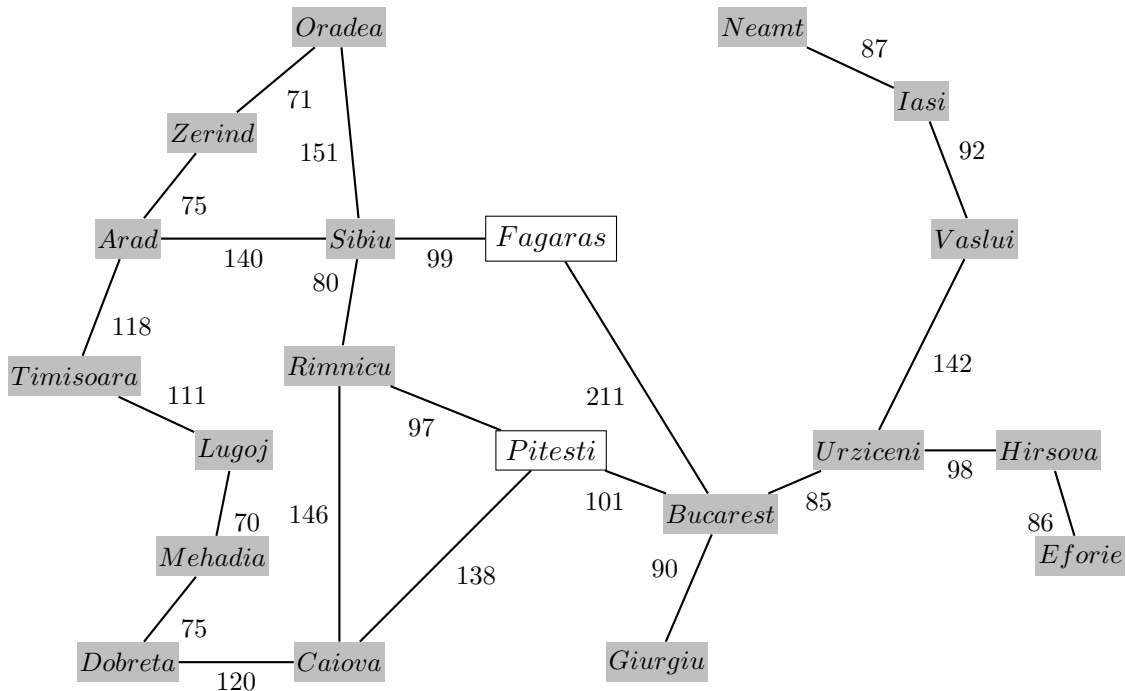
Cada vez sólo se puede mover un disco, los discos siempre deben estar en algún poste y no se puede colocar un disco sobre otro de menor tamaño.

- (a) ¿Cómo representarías los estados?
  - (b) ¿Cuáles serían los operadores?
9. Un acertijo consiste en dados 4 números y un resultado, determinar las operaciones de suma o resta que hay que realizar sobre los números para obtener ese resultado. Por ejemplo:
- Números: 1, 4, 3, 2
  - Resultado: 0
  - Solución:  $4 - 3 - 2 + 1$

Suponiendo que resolvemos el acertijo como un problema de búsqueda, responde las siguientes cuestiones:

- (a) Propón una representación de los estados y explica cómo se generarían los estados sucesores.
- (b) ¿Cuál sería el tamaño del espacio de estados?
- (c) ¿Qué tipo de algoritmo de búsqueda no informada sería mejor utilizar y por qué?

10. Aplica la búsqueda de coste uniforme para encontrar la ruta más corta de Pitesti a Fagaras.



- (a) Desarrolla el árbol de búsqueda generado por el algoritmo, asumiendo que se evitan ciclos simples.
- (b) Indica el valor  $g$  de cada nodo, así como el orden en que se expanden.
11. Se han descubierto  $A$  fuentes de contaminación en un parque natural y se quieren colocar  $B$  aparatos de descontaminación (donde  $B < A$ ) para mejorar la situación. Para ello se dispone de un mapa del parque que indica la posición de la estación de trenes donde se han almacenado todos los aparatos y de los  $A$  lugares donde se necesita colocar los aparatos de descontaminación. Además también se dispone del nivel de contaminación que hay alrededor de cada fuente, de un mapa de los desplazamientos posibles de los aparatos en el territorio y del coste de cada desplazamiento. Cada aparato puede eliminar por completo la contaminación de una fuente, independientemente de su nivel.

El objetivo es colocar los aparatos de manera que se minimice la contaminación total en el parque y el coste del recorrido (suma de desplazamientos) que harán los aparatos en el sentido “estación  $\rightarrow$  fuente de contaminación”. En los siguientes apartados se proponen diferentes alternativas para algunos de los elementos necesarios para plantear la búsqueda (solución inicial, operadores, función heurística). Hay que comentar la solución que se propone respecto a si es correcta, es eficiente, y es mejor o peor en comparación con otras alternativas, además de justificar todas las respuestas.

- (a) Se planea solucionar el problema mediante un método de escalada, partiendo de una solución inicial sin ningún aparato y con un operador que coloca un





ugr

Universidad de Granada  
Departamento de Ciencias de la Computación  
e Inteligencia Artificial

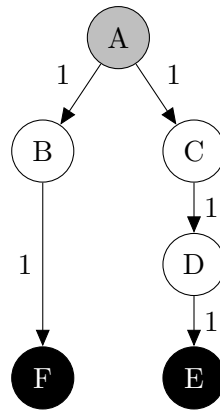


aparato en una fuente de contaminación determinada, controlando que el número de los aparatos colocados sea como máximo  $B$ .

- (b) Se plantea solucionarlo mediante un método de escalada, partiendo de una solución inicial con  $B$  aparatos colocados aleatoriamente, y utilizando como función heurística la suma de los costes de desplazamiento de la estación a cada una de las  $B$  fuentes.
  - (c) Se plantea solucionarlo mediante un método de escalada, partiendo de una solución inicial con  $B$  aparatos colocados aleatoriamente, y utilizando como función heurística la suma de los costes mínimos de los recorridos “estación  $\rightarrow$  fuente de contaminación” multiplicada por la suma de los niveles de contaminación correspondientes a los  $B$  aparatos.
  - (d) Se plantea solucionarlo mediante un método de escalada, partiendo de una solución inicial alcanzada colocando los  $B$  aparatos ordenadamente según el coste mínimo “estación  $\rightarrow$  fuente de contaminación” y empezando con el que tiene coste menor. Se plantea como operador mover un aparato a cualquier fuente cuyo producto de coste mínimo “estación  $\rightarrow$  fuente de contaminación” sea menor que el actual.
  - (e) Se plantea solucionarlo mediante algoritmos genéticos: se usan individuos de  $A$  bits y como población inicial se generan  $n$  individuos donde en cada uno hay exactamente  $B$  bits a 1. La función de idoneidad es la suma de los costes mínimos “estación  $\rightarrow$  fuente de contaminación” más la contaminación total residual del parque multiplicada por una constante. Como operadores se usan los habituales de cruce y mutación.
12. Sobre el problema de misioneros y caníbales tomar como función heurística el número total de personas en la orilla del río que hace de salida. ¿Qué resultado darían método de escalada en la resolución de este problema?
13. El algoritmo  $A^*$  no termina mientras no se selecciona un nodo objetivo para su expansión. Sin embargo, es posible encontrar un camino al objetivo mucho antes.
- (a) ¿Por qué no se termina en el momento en que se encuentra un nodo objetivo?
  - (b) Ilustre la respuesta anterior con un ejemplo.
14. El grafo que se muestra en la figura describe un problema de búsqueda. Suponga que  $A$  es el estado inicial y que  $F$  y  $E$  son estados meta. Los arcos están etiquetados con el coste real de los operadores y  $h$  es una función heurística cuyos valores son:

	A	B	C	D	E	F
$h$	8	6	6	5	0	0

- (a) Desarrolle el árbol de búsqueda que genera el algoritmo  $A^*$ .
- (b) Indique el orden en que se expanden los nodos.
- (c) ¿Cuál de los nodos meta se encuentra primero?
- (d) ¿La función heurística es admisible? Argumente su respuesta.



15. Considere el problema de búsqueda del ejercicio precedente.
- Asigne los valores del coste real de los operadores y de la función heurística  $h$ , de modo que ésta resulte ser admisible.
  - Desarrolle el árbol de búsqueda que genera el algoritmo  $A^*$ .
  - ¿Se ha obtenido la solución óptima?
16. Tenemos cinco monedas dispuestas de la siguiente forma:

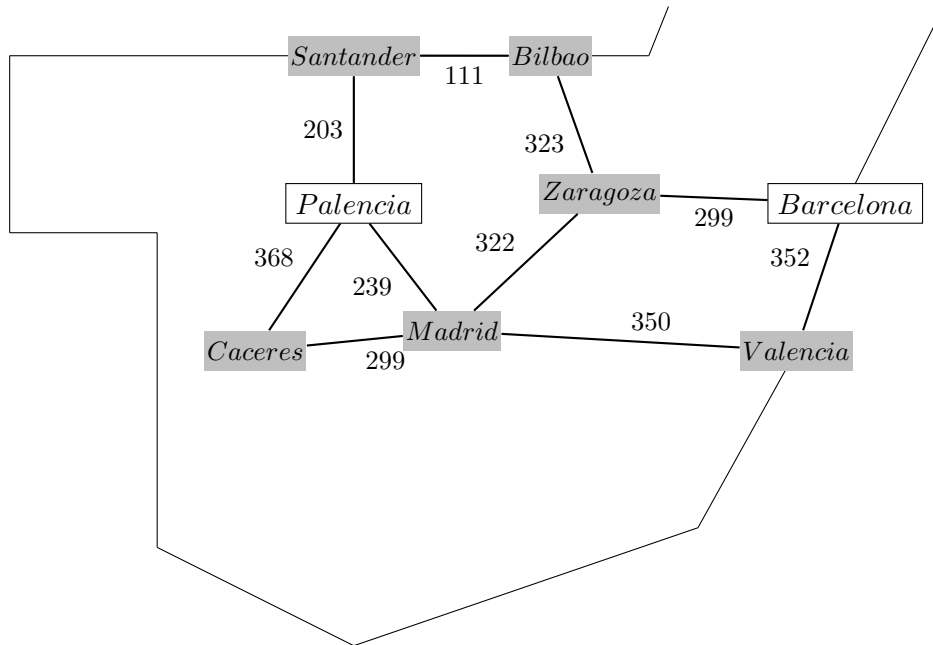
$A R A R A$

El anverso de la moneda está representado por A y el reverso por R. Se considera un movimiento (de coste 1) el dar la vuelta a dos monedas contiguas. Deseamos obtener la situación final siguiente:

$R R R A R$

Dada la función heurística  $h(n) = \text{número de monedas mal colocadas}$ .

- Resolver el problema aplicando  $A^*$ . Indica claramente el orden de expansión de los nodos, el tratamiento de nodos duplicados, los valores de las funciones, el camino obtenido y su coste.
  - ¿La heurística usada es admisible? ¿Por qué?
17. Dado el siguiente mapa de carreteras en el que los caminos entre cada dos ciudades están etiquetados con sus distancias en kilómetros:

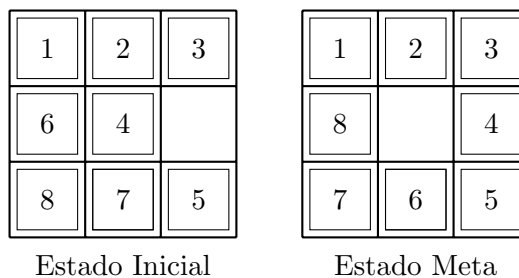


El objetivo es encontrar el camino más corto entre Palencia y Barcelona. Para ello:

- Aplica el algoritmo A\* utilizando como estimación del coste del camino entre la ciudad actual y el objetivo el valor 0 para todos los nodos.
- Aplica el algoritmo A\* utilizando ahora como estimación del coste del camino entre la ciudad actual y el objetivo las distancias siguientes:

	Bilbao	Cáceres	Madrid	Palencia	Santander	Valencia	Zaragoza
Barcelona	502	850	550	580	605	303	275

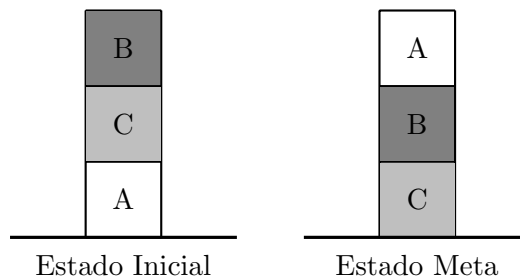
18. Considere el 8-puzzle cuyo estado inicial y estado meta se muestran en la siguiente figura:



Desarrolle el árbol de búsqueda que expande el algoritmo A\* utilizando las siguientes heurísticas. Evite ciclos generales, indique el orden de expansión de los estados y muestre en cada paso los valores de  $f$ ,  $g$  y  $h$ . Suponga que el coste de cada operador es 1.



- (a) Heurística  $h_1(n)$  = número de piezas mal colocadas.
  - (b) Heurística  $h_2(n)$  = suma de distancia Manhattan. La distancia Manhattan de una pieza es la suma de las distancias vertical y horizontal a su posición final.
  - (c) ¿Cuál de las heurísticas expande menos nodos y por qué?
  - (d) ¿Puede sacar una conclusión general con respecto a la calidad de las funciones heurísticas?
19. Considere el problema de los bloques cuyo estado inicial y estado meta se muestran en la siguiente figura:

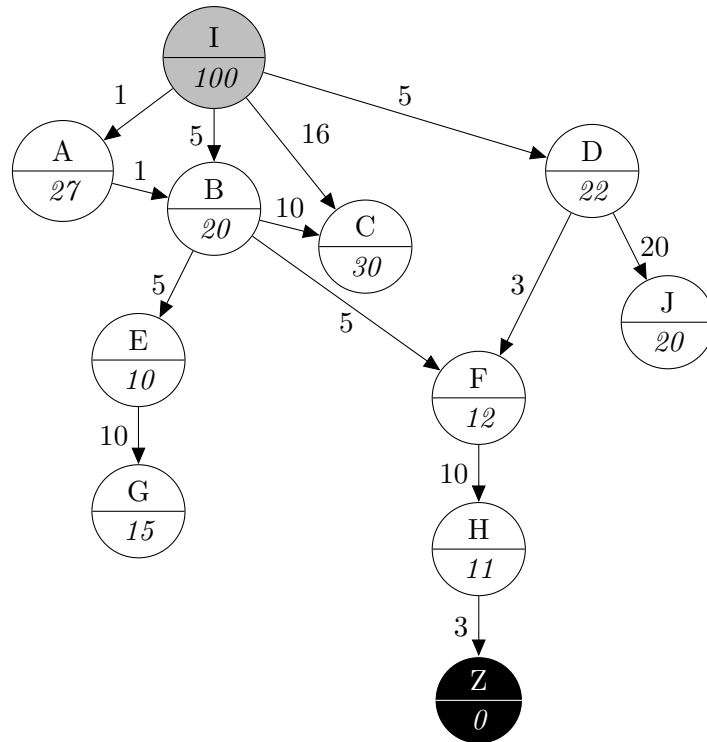


Desarrolle el árbol de búsqueda que expande el algoritmo A\* utilizando la siguiente heurística:

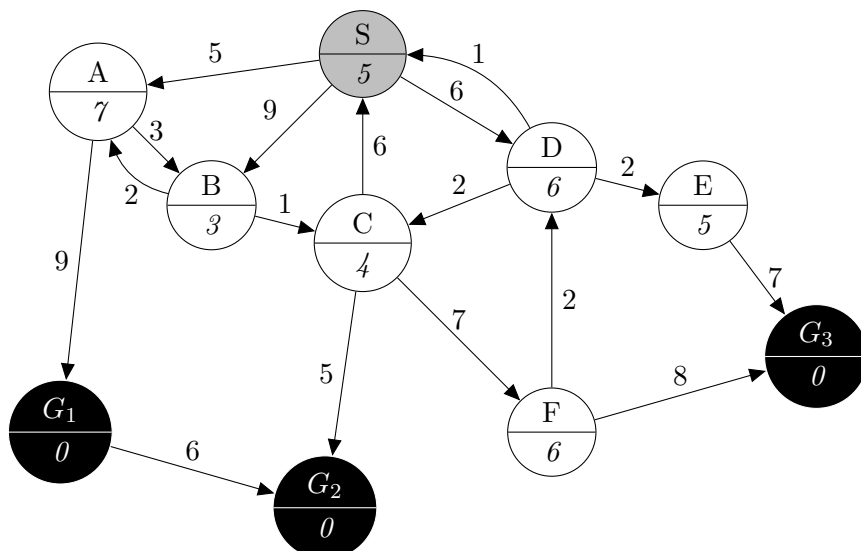
$$h(n) = \text{número de bloques mal colocados}$$

Filtre los ciclos simples, indique el orden de expansión de los estados y muestre en cada paso los valores de  $f$ ,  $g$  y  $h$ . Suponga que el coste de cada operador es 1.

20. Dado el siguiente grado representando un espacio de estados, utilizar el algoritmo A\* para obtener la solución al problema. En dicho grafo considerar como función  $h$  la que se indica dentro de cada nodo, y para el cálculo de la función  $g$  considerar el coste que aparece en el arco. El nodo inicial es  $I$  y hay un solo nodo meta  $Z$ .



- (a) Mostrar el gráfico final con la solución obtenida, los valores  $g$  y  $h$  finales para cada nodo, junto con los enlaces a los hijos y enlace al mejor padre.
- (b) ¿Es la función heurística utilizada en el problema admisible?
21. Supongamos que el siguiente grafo representa un espacio de estados de un problema real. El número dentro de cada nodo representa el valor de la función  $h$  del nodo y el número sobre el arco el coste del operador.







$S$  es el nodo de inicio y  $G_1$ ,  $G_2$  y  $G_3$  son nodos que satisfacen la condición objetivo. En todos los casos ABIERTOS representará el conjunto de nodos que puede ser explorados en la siguiente iteración del algoritmo de búsqueda. Utilizar distintos algoritmos de búsqueda para resolver el problema, para cada algoritmo obtener los siguientes datos:

- El conjunto de nodos expandidos en el orden que salen de ABIERTOS.
- El objetivo alcanzado.
- El camino solución encontrado.

Utilizar los siguientes algoritmos:

- Búsqueda en anchura.
- Profundizaje iterativo
- Método de escalada simple (usando solo la función  $h$ ).
- Método de escalada por la máxima pendiente (usando solo la función  $h$ ).
- Algoritmo  $A^*$  (usando la función  $f = g + h$ ).

22. Dado el grafo del problema anterior, supongamos que estamos utilizando el algoritmo de enfriamiento simulado (usando solo la función  $h$ ). Nos encontramos en el nodo  $S$ , y el algoritmo tiene que decidir si cambia al nodo  $A$  como siguiente estado. Si suponemos que la temperatura es en ese momento 10:

- ¿Cuál es la probabilidad de aceptar  $A$  como siguiente estado?
- Repite lo anterior tomando como nodo actual  $A$  y nodo candidato a siguiente estado  $B$  y la misma temperatura.

23. Aplicar el algoritmo  $A^*$  para hallar el camino con menor distancia que una las ciudades 1 y 8. Las distancias por carretera entre las distintas ciudades vienen especificada por la siguiente tabla:

	1	2	3	4	5	6	7	8
1		200						
2	200		150	250	450			
3		150			400	225		
4		350			300			
5		450	400	300			250	
6			225				450	
7					250	450		125
8							125	

Además se dispone de la distancia aérea en línea recta que existe entre todas las ciudades con la ciudad de destino:



ugr

Universidad de Granada  
Departamento de Ciencias de la Computación  
e Inteligencia Artificial



1	2	3	4	5	6	7
800	650	500	650	325	375	125

24. Considérese una modificación del algoritmo  $A^*$  en el que en lugar de usar  $f = g + h$  para ordenar la cola de abiertos se usa  $f_w(n) = (1 - w) * g(n) + w * h(n)$ , siendo  $w$  un número real constante entre 0 y 1.

- (a) ¿Qué algoritmo de búsqueda estaríamos aplicando si tomamos  $w = 0$ ?
- (b) ¿Y con  $w = 0.5$ ?
- (c) ¿Y con  $w = 1$ ?



# INTELIGENCIA ARTIFICIAL

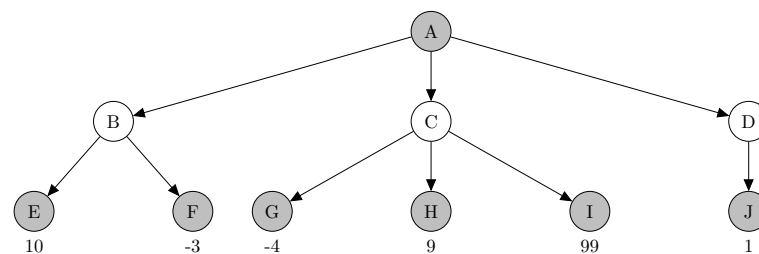
Curso 2021/22

## Relación de Problemas 3

### BÚSQUEDA CON ADVERSARIO: JUEGOS

1. Aplicar el algoritmo minimax para el árbol de juego de la siguiente figura, donde el primero que juega es el jugador Max.

- Indicar el valor del juego para Max y la mejor jugada a realizar.
- ¿Qué nodos no necesitan ser explorados si los descendientes de un nodo se visitan de izquierda a derecha y se aplica la poda alfa-beta?



2. Considérese un juego de cartas en el que dos jugadores por turnos van quitando una carta de una fila de cartas, bien por el extremo izquierdo o bien por el derecho. Inicialmente la fila contiene cinco cartas etiquetadas con las letras A o B, de la siguiente manera:

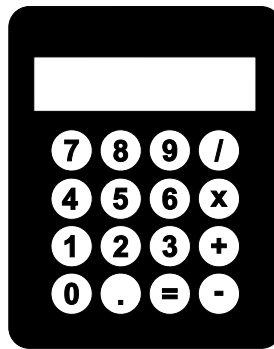


El jugador que quita la última carta gana si ésta está etiquetada con A, y pierde en otro caso. Se trata de probar que el segundo jugador siempre puede ganar.

- Dibujar el árbol del juego e identificar una estrategia ganadora para el segundo jugador.



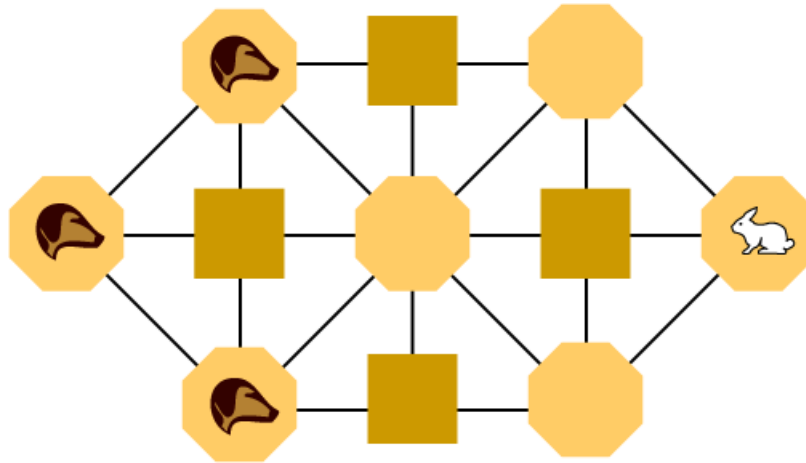
- Justificar qué técnica de resolución de juegos se está utilizando. No se considerará válida una solución que no esté correctamente formalizada.
3. Considérese el juego bipersonal en el que los dos jugadores van sumando cifras, con la ayuda de una calculadora, hasta alcanzar el número 31 ó superior. El juego consiste en ir pulsando en cada turno una de las teclas numéricas del 1 al 9 de la calculadora y el signo “+”, teniendo en cuenta que la tecla numérica que se puede pulsar no puede ser la última que pulsó el oponente y que tiene que estar en la misma fila o en la misma columna que ésta. El jugador que en su turno suma 31 ó más pierde la partida.
- Establecer y discutir la representación.
  - Estudiar cómo se podría resolver este juego.



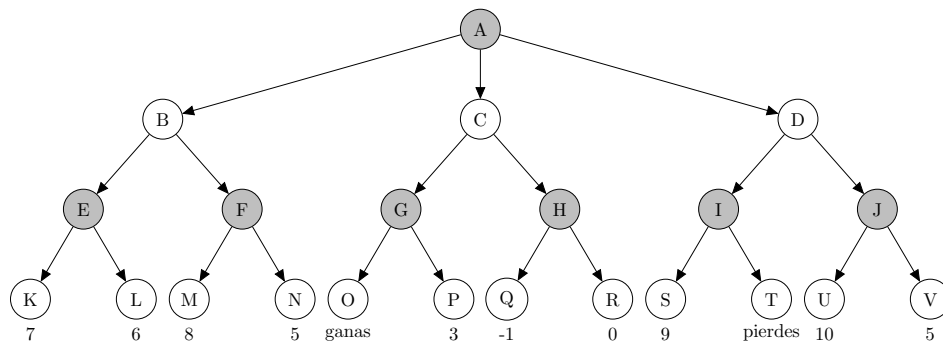
4. Hare & Hounds (Presa & Cazador) <sup>[1]</sup> es un juego de estrategia para 2 jugadores <sup>[2]</sup>. La presa decide los movimientos de la liebre, mientras que el cazador decide los movimientos de los perros. La presa gana el juego si consigue llegar de un extremo a otro del tablero. En cambio, el cazador gana si logra atrapar a la liebre, es decir, si se alcanza una situación del juego en la que la liebre no puede realizar ningún movimiento. En cada turno cada jugador puede mover sólo una ficha y siempre a una de las casillas adyacentes. Las fichas del cazador no pueden retroceder. El primer turno corresponde siempre al cazador. Se pide:
- Indicar posibles representaciones de los estados.
  - Describir los operadores del juego.
  - ¿Qué funciones de evaluación se te ocurren para programar un jugador automático inteligente del juego?

<sup>1</sup>[https://en.wikipedia.org/wiki/Hare\\_games](https://en.wikipedia.org/wiki/Hare_games)

<sup>2</sup>En <http://www.appletonline.com/JavaApplets/HoundsAndHare/> puede encontrarse un applet para jugar al juego.



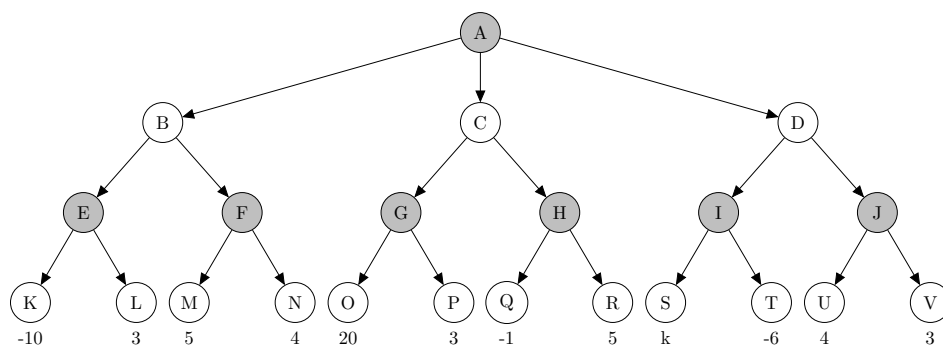
5. Dado el siguiente árbol de un juego, aplicar el algoritmo de poda alfa-beta para obtener la jugada minimax del mismo.



Representa apropiadamente en el algoritmo el “ganas” y “pierdes” y una vez aplicado el algoritmo:

- Especificar que nodos terminales NO es necesario evaluar por el algoritmo
- Dar el valor minimax del juego

6. Dado el siguiente árbol de un juego:





- Determinar el rango completo de valores del parámetro  $k$  que verifican que aplicando la poda alfa-beta se podan por lo mínimo cuatro nodos (terminales o no terminales).
- Especificar cuáles son los nodos que se han podado (la exploración del árbol se debe realizar de forma habitual utilizando la poda alfa-beta y explorarnos los nodos de izquierda a derecha).

7. Consideremos el siguiente juego. Hay dos monedas sesgadas  $M_1$  y  $M_2$ . La moneda  $M_1$  tiene probabilidad 0,75 de salir cara y 0,25 de salir cruz. La moneda  $M_2$  tiene probabilidad 0,10 de salir cara y 0,90 de salir cruz. Supongamos que el jugador  $J_1$  puede elegir una moneda y lanzarla. Una vez visto el resultado, el jugador  $J_2$  puede elegir cualquiera de las dos monedas y la lanza. Al final el jugador  $J_1$  obtiene un beneficio que viene dado por la siguiente tabla:

$J_1 \ M_1$								$J_1 \ M_2$							
Ca				Cr				Ca				Cr			
$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$	$J_2 \ M_1$	$J_2 \ M_2$
Ca	Cr	Ca	Cr	Ca	Cr	Ca	Cr	Ca	Cr	Ca	Cr	Ca	Cr	Ca	Cr
8	9	7	3	4	2	0	3	7	5	9	7	1	6	8	0

Donde la primera fila es la moneda elegida por  $J_1$ , la segunda fila es el resultado de esa moneda, la tercera la moneda elegida por  $J_2$ , la cuarta el resultado de esa moneda y la quinta es el resultado obtenido por  $J_1$ .

- Resolver el problema, calcular el valor del juego y la estrategia óptima para el jugador  $J_1$ .

