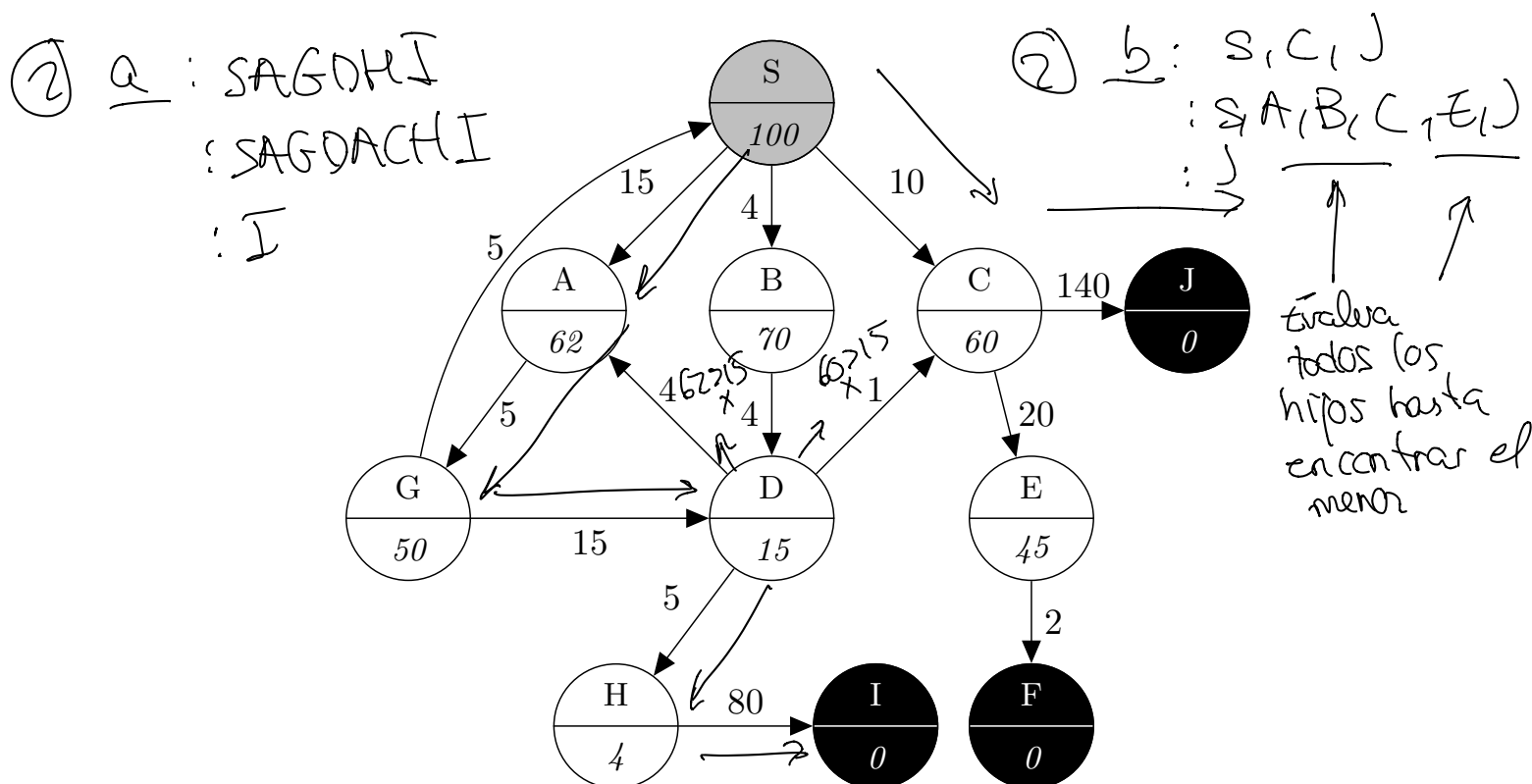


**Inteligencia Artificial**  
**Departamento de Ciencias de la Computación e Inteligencia Artificial**  
**Curso 2015/2016 (Convocatoria de Junio)**  
**Examen de Ejercicios**

Nombre:	
DNI:	
Titulación:	
Grupo:	Subgrupo Prácticas:

1. 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  $S$ . Los nodos objetivos son  $I$ ,  $J$  y  $F$ .



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.

① b)

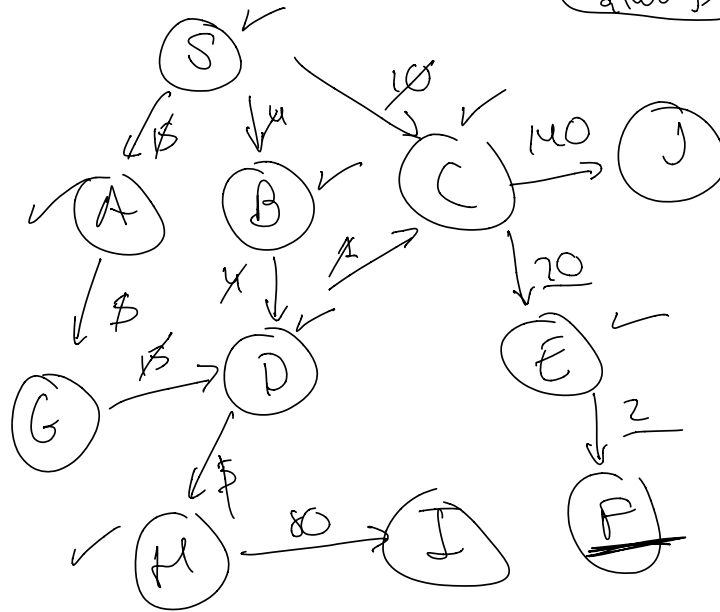
cheapest always

Visitados

S, B, D, C

A, H, G, E

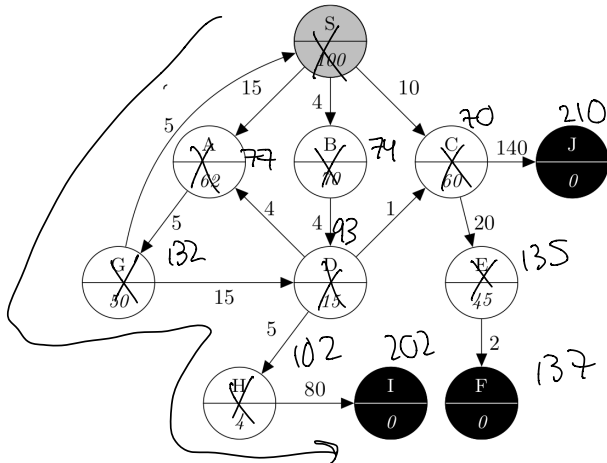
F



F, E, C, D, B, S → S, B, D, C, E, F

c) Abiertos: A, B, J, E, D, G, H, I, F

Cerrados: S, C, B, A, D, H, G, E, F



a) S, A, G, D, H, I  
I  
S, A, G, D, H, I

~~(a)~~ Algoritmo 1: Búsqueda en Profundidad (1 punto)

Secuencia de nodos:	S, A, G, D, H, I	Alternativa	S, C, J
Estado objetivo alcanzado:	I	Alternativa	J
Camino solución:	S A G D H I	Alternativa	S C J

\* ~~(b)~~

Algoritmo 3: Búsqueda Costo Uniforme (1,5 puntos)

↪ camino más económico

Secuencia de nodos:	S, B, D, C, A, H, G, E, F
Estado objetivo alcanzado:	F
Camino solución:	S B D C E F

\* ~~(c)~~

Algoritmo 4: Algoritmo A\* (2 puntos)

Secuencia de nodos:	S, C, B, D, H, A, G, E, F
Estado objetivo alcanzado:	F
Camino solución:	S B D C E F

2. Prueba los siguientes algoritmos de búsqueda sobre el grafo del ejercicio anterior generando los nodos siempre en orden alfabético y usando  $h$  como función heurística.

~~(a)~~

Algoritmo 1: Búsqueda en Escalada Simple (0,75 puntos)

Mayor costo camino

Secuencia de nodos (camino):	S, A, G, D, H, I
Estado objetivo alcanzado:	I
Secuencia de nodos (evaluados):	S, A, G, D, A, C, H, I

~~(b)~~

Algoritmo 2: Búsqueda en Escalada con Pendiente máxima (0,75 puntos)

Secuencia de nodos (camino):	S, C, J
Estado objetivo alcanzado:	J
Secuencia de nodos (evaluados):	S, A, B, C, E, J

3. (4 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 de 23 valores distintos para los 46 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 y 7 han sido podados en el proceso, y la lista refleja los valores heurísticos asignados a los nodos no 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.



