SEPTIEMBRE 2014 – ORIGINAL (MODELO A)

1. Una filmoteca ha organizado un maratón de cortometrajes. Durante 24 horas se proyectarán cortos de cine (todos diferentes) en las n salas disponibles. Un cinéfilo ha conseguido la programación completa donde aparecen todas las películas que se van a proyectar durante el maratón, incluyendo el título, duración del corto, sala en la que se proyecta y hora de comienzo.

Si se quiere planificar el maratón del cinéfilo de forma que pueda ver el máximo número posible de cortos, ¿Cuál es el esquema más apropiado para hacer la planificación eficientemente?

(a) Esquema voraz.

(b) Divide y vencerás.

(c) Esquema de vuelta atrás.

(d) Esquema de ramificación y poda.

Respuesta A)

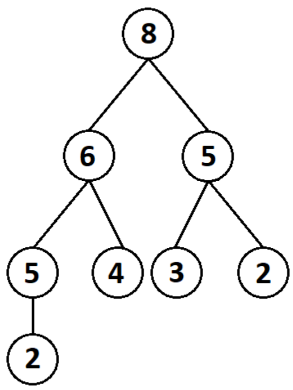
2. Considérese el vector v[1 .. n] = [8,6,5,5,4,3,2,2]. Indique cuál de las siguientes opciones es cierta:

(a) El vector v es un montículo de máximos.

(b) El vector v no es un montículo de máximos porque el elemento v[5] = 4 debe ser flotado.

(c) El vector v no es un montículo de máximos porque el elemento v[4] = 5 debe ser hundido.

(d) Ninguna de las anteriores.



a) Cierto. Cada hijo de los nodos es un valor menor al de su nodo padre.
b) Falso. Si ocurriera, no podría ser un montículo de máximos.
c) Falso. Si ocurriese, dejaría de ser un montículo de máximos.
d) Falso. No es posible, ya que la respuesta correcta es la A)

Respuesta A)

3. Sea el problema de la mochila, en el que tenemos una mochila de capacidad M, n objetos con beneficios $b_1, b_2, b_3, \ldots b_n$; y pesos $p_1, p_2, p_3, \ldots p_n$. El objetivo es maximizar el valor de los objetos transportados, respetando la limitación de la capacidad impuesta M. Indica de los esquemas siguientes cuál es el más adecuado en el caso de que cada objeto puede meterse en la mochila, no meterse, o meter la mitad del objeto obteniendo la mitad del beneficio.

(a) El esquema voraz.

> *Con este algoritmo no se obtiene la solución óptima si los objetos no son totalmente fraccionables.*

(b) El esquema divide y vencerás.

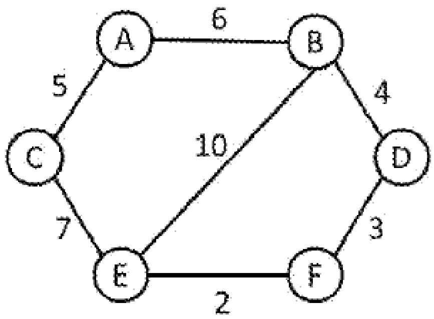
(c) El esquema de vuelta atrás.

(d) El esquema de ramificación y poda.

> *Este es el esquema óptimo tanto con objetos fraccionables como enteros.*

Respuesta D)

4. Sea el grafo de la figura:



Indique cuál sería el orden en que se seleccionan los nodos del conjunto de candidatos al aplicar el algoritmo de Dijkstra comenzando por el nodo A:

(a) A C B D E F
(b) A B C D E F
(c) A B D C F E
(d) Ninguna de las anteriores

Nodos que han salido	Nodos que no han salido	Vector Distancia especial [ ]					Predecesores				
		B	C	D	E	F	B	C	D	E	F
A	B, C, D, E, F	6	**5**	∞	∞	∞	A	A	A	A	A
A, C	B, D, E, F	**6**	5	∞	12	∞	A	A	A	C	A
A, C, B	D, E, F	6	5	**10**	12	∞	A	A	B	C	A
A, C, B, D	E, F	6	5	10	**12**	13	A	A	B	C	D
A, C, B, D, E	F	6	5	10	12	13	A	A	B	C	D

Respuesta A)

5. En relación a la representación de grafos mediante listas de adyacencia, indique cuál de las siguientes afirmaciones es <u>falsa</u>:

(a) Sean el número de nodos del grafo, necesitaremos n listas para representarlo.

(b) Si el grafo presenta pocas aristas, es más eficiente en espacio representarlo mediante listas de adyacencia que representarlo mediante una matriz de adyacencia.

(c) El coste asociado a la operación *Borrar Arista* es O(n+a), siendo n el número de nodos del grafo y siendo a el número de aristas.

(d) El coste asociado a la operación Añadir Vértice es 0(1).

Funciones Manipulación de Grafos	**Matriz de Adyacencia**	**Lista de Adyacencia**	**Descripción**
CrearGrafo	O(1)	O(1)	Devuelve un grafo vacío
AñadirArista	O(1)	O(1)	Añade una arista entre los vértices ‘u’ y ‘v’, y le asigna un peso ‘p’
AñadirVertice	O(n)	O(1)	Añade el vértice ‘v’ al grafo ‘g’
BorrarArista	O(1)	O(n)	Elimina la arista entre los vértices
BorrarVertice	O(n)	O(n + a)	Borra el vértice ‘v’ al grafo ‘g’ y todas las aristas que partan o lleguen a él
Adyacente? o esAdyacente	O(1)	O(n)	Comprueba si los vértices son adyacentes
Adyacentes	O(n)	O(1)	Devuelve una lista con los vértices adyacentes a ‘v’
Etiqueta	O(1)	O(n)	Devuelve la etiqueta o peso asociado a la arista que une los vértices

<u>Respuesta C)</u>

6. Indica cuál de las siguientes afirmaciones es <u>cierta</u>:

(a) Un árbol libre es un grafo acíclico, conexo y dirigido.
(b) Un ciclo es un camino simple que empieza y termina en el mismo vértice.
(c) Si un grafo tiene pocas aristas las listas de adyacencia resultan una estructura más costosa en espacio que la matriz de adyacencia.
(d) Ninguna de las definiciones anteriores es completa y cierta.

a) Árbol Libre: es un grafo acíclico, conexo y no dirigido. No tienen raíz y los nodos hijos no están desordenados.

b) Ciclo: También llamado circuito, es un camino simple que empieza y termina en el mismo vértice.

c) Si un grafo tiene pocas aristas, las listas de adyacencias resultan una estructura más costosa en espacio. Sin embargo, cuando se acercan a n^2 (máximo número de nodos posibles), el coste en espacio es del mismo orden. La matriz de adyacencia es más sencilla de tratar, ya que no utilizan punteros.

<u>Respuesta B)</u>

PROBLEMA (4 Puntos)

Desarrollar un programa que compruebe si es posible que un caballo de ajedrez, mediante una secuencia de sus movimientos permitidos, recorra todas las casillas de un tablero NxN a partir de una determinada casilla dada como entrada y sin repetir ninguna casilla.

La resolución de este problema debe incluir, por este orden:

1. Elección del esquema más apropiado, el esquema general y explicación de su aplicación al problema (0,5 puntos)

El esquema correcto es el de vuelta atrás. No hay heurísticas ni criterios de búsqueda y el problema no puede descomponerse en subproblemas de su misma naturaleza, por lo que se utiliza la backtraking para la exploración del árbol de búsqueda.

2. Descripción de las estructuras de datos necesarias (0,5 puntos solo si el punto 1 es correcto)

Las estructuras de datos son el tablero de NxN que contiene en cada casilla valores que indiquen que el caballo no ha pasado (cero) o que ha pasado (m) donde m es un natural que indica que en el movimiento m-ésimo, el caballo llega a dicha casilla.

El nodo contaría con dicho tablero, con el número de caballos puestos en el tablero hasta el momento, y con la posición actual del caballo (el último movimiento efectuado).

3. Algoritmo completo a partir del refinamiento del esquema general (2,5 puntos solo si el punto 1 es correcto)

El refinamiento del esquema precisa detallar cómo generar las compleciones y cómo realizar la recursión por el árbol implícito. Las compleciones son las siguientes:

```
Fun compleciones(ensayo)lista ← lista_ vacía

(i,j) → nodo.ultima_posición
n ← nodo.numero_movimientos
ultimo_tablero ← ensayo tablero

para i ← -2 hasta 2 hacer
     para j ← -2 hasta 2 hacer

          si |i|+|j|==3 AND nodo.tablero[i,j] == 0 entonces hacer

               nuevoTablero ← ultimoTablero
               nuevoTablero[i,j] ← n + 1
               nuevaPosicion ← (i,j)
               nuevoNodo ←  < nuevoTablero, nuevaPosición, n+1>
               lista.add(nuevoNodo)

          fsi
     fpara
fpara
dev lista
ffun
```

El algoritmo principal queda como sigue:

```
Fun saltoCaballo (nodo)

si solucion (nodo)       /* el numero de casillas libre es cero */
entonces escribe(nodo)    /* hemos terminado */
sino hacer

     lista ← compleciones (nodo) /* lista de los posibles movimiento
                                    válidos */
     mientras  ¬ lista.vacia() hacer
          w ← lista.primerElemento()
          saltoCaballo(w)
          lista ← lista.resto()
     fmientras
fsi
ffun
```

4. Estudio del coste <u>del algoritmo desarrollado</u> (0,5 puntos solo si el punto 1 es correcto)

Sin tener en cuenta las reglas de colocación del caballo, las n^2 casillas pueden numerarse de $(n^2)!$ maneras posibles, sin embargo sabemos que el número máximo de ramificaciones del árbol es 8 por ser éstas las alternativas de movimiento de un caballo de ajedrez.

Considerando esto, el tamaño del árbol puede acotarse en 8^{n^2} pero se puede precisar que no hay 8 ramas en todos los casos. De cada n^2 casillas, solo desde $n^2-8(n-2)$ es posible realizar 8 movimientos, y además los últimos movimientos no tendrán prácticamente ninguna alternativa.