

FEBRERO 2013 – SEMANA 1 (MODELO A)

1. Con respecto a la resolución de colisiones en las funciones Hash se puede afirmar que:

- a) El recorrido lineal permite mayor dispersión de las colisiones que el cuadrático.
- b) Si el factor de carga es 1 se puede resolver mediante hashing cerrado.
- c) El hashing abierto contempla un método de resolución conocido como "recorrido mediante doble hashing".
- d) Ninguna de las anteriores es correcta.

Explicación: El recorrido cuadrático permite mayor dispersión. Si el factor de carga es 1 se puede resolver doblando el tamaño de la tabla con hashing abierto. El doble hashing se da en hashing cerrado.

2. Durante la ejecución de un algoritmo de Ramificación y Poda se halla una solución que es mejor que la mejor solución existente en ese momento y que mejora la estimación optimista de la cima del montículo. Esto implica:

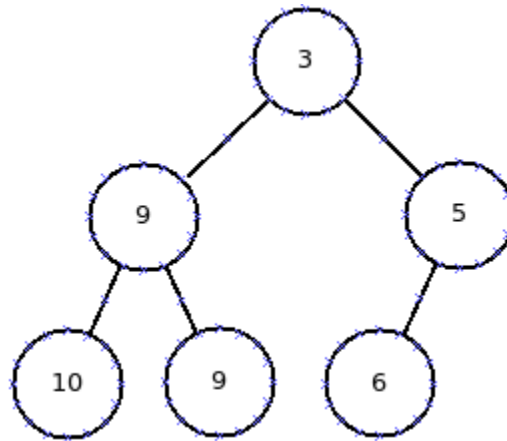
- a) Definitivamente el algoritmo ha encontrado la solución.
- b) Se actualiza la cota y se sigue con la exploración porque no hemos terminado.
- c) Se actualiza la cima del montículo y se sigue con la exploración porque no hemos terminado.
- d) Ninguna de las anteriores es correcta.

3. Se dispone de n cubos numerados del 1 al n . Cada cubo tiene impresa una letra distinta en cada una de sus caras, aunque distintos cubos pueden tener letras repetidas. Se pretende, dada una palabra de longitud n , colocar el total de cubos de manera consecutiva de forma que se forme la palabra dada. ¿Cuál de los siguientes esquemas algorítmicos es el más apropiado para resolver dicho problema?

- a) Programación dinámica.
- b) Divide y vencerás.
- c) Ramificación y poda.
- d) Vuelta atrás.

Explicación: Habiendo más de una solución hay que explorarlas todas, y eso lo hace Vuelta Atrás.

4. Dado el siguiente montículo de mínimos cuál de los siguientes vectores lo representa de forma correcta:



a) [3,9,10,9,5,6].

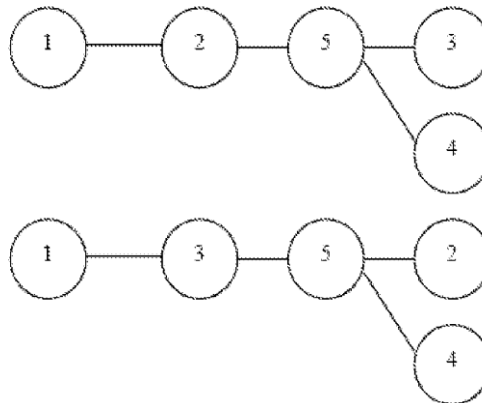
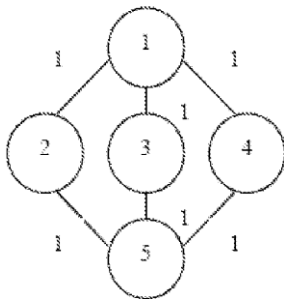
b) [10,9,9,6,5,3].

c) [3,9,5,10,9,6].

d) El montículo propuesto es un montículo de máximos y no de mínimos.

Explicación: Se leen las filas completas de arriba hacia abajo (recorrido en anchura).

5. Selecciona la afirmación más ajustada de las siguientes. Las siguientes tres figuras corresponden a:



a) La de la izquierda es un grafo y las de la derecha son dos posibles árboles de recubrimiento mínimo asociados a él.

b) Las tres corresponden a tres grafos no dirigidos sin ninguna relación entre ellos.

c) La de la izquierda es un grafo y a los grafos no dirigidos conexos con aristas de igual coste no se les puede asociar más de un árbol de recubrimiento mínimo.

d) Ninguna de las anteriores es correcta.

6. Considere un array $A[1..n]$ ordenado y formado por enteros diferentes, algunos de los cuales pueden ser negativos. Se busca un algoritmo recursivo que calcule en tiempo logarítmico un índice i tal que $1 \leq i \leq n$ y $T[i] = i$, siempre que este índice exista. Se supone que las operaciones elementales tienen coste unitario. ¿A cuál de los siguientes esquemas correspondería dicho algoritmo?

a) Esquema voraz.

b) Divide y vencerás.

c) Esquema de vuelta atrás.

d) Esquema de ramificación y poda.

Explicación: Divide y vencerás, se va dividiendo el vector y se comprueba el elemento central entre ambas mitades. En caso de $T[i] = x < l$ se elige la mitad derecha, sino la izquierda.

PROBLEMA (4 Puntos)

El tío Facundo poseen huertas, cada una con un tipo diferente de árboles frutales. Las frutas ya han madurado y es hora de recolectarlas. El tío Facundo conoce, para cada una de las huertas, el beneficio b_i que obtendría por la venta de lo recolectado. El tiempo que se tarda en recolectar los frutos de cada finca es así mismo variable (no unitario) y viene dado por t_i . También sabe los días d_i que tardan en pudrirse los frutos de cada huerta. Se pide programar un algoritmo para ayudar a decidir al tío Facundo qué debe recolectar para maximizar el beneficio total obtenido.

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).*
- 2. Descripción de las estructuras de datos necesarias (0.5 puntos solo si el punto 1 es correcto).*
- 3. Algoritmo completo a partir del refinamiento del esquema general (2,5 puntos solo si el punto 1 es correcto).*
- 4. Estudio del coste del algoritmo desarrollado (0.5 puntos solo si el punto 1 es correcto).*