

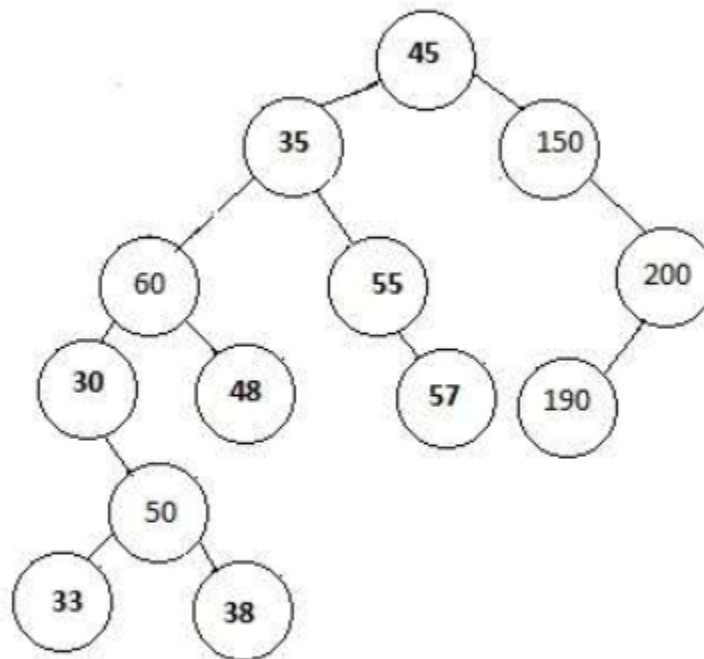
1) Dado a de tipo ABB, Indicar cómo queda en cada estado dando una breve explicación, sabiendo que comienza vacío y se hacen las siguientes operaciones:

- a. a.insertar(20)
- b. a.insertar(28)
- c. a.insertar(11)
- d. a.insertar(18)
- e. a.insertar(3)
- f. a.insertar(22)
- g. a.insertar(15)
- h. a.eliminar(28)
- i. a.insertar(16)
- j. a.insertar(30)
- k. a.eliminar(11)

2) Árbol B.

- a. Definir árbol B de orden m.
- b. Sea un árbol B de orden $m = 5$. Indicar cómo queda en cada paso al insertarse las siguientes claves: 8, 14, 2, 15, 3, 1, 16, 6, 5, 27, 37, 18, 25, 7, 13, 20, 22, 23, 24. El árbol comienza vacío.

3) Dado el árbol:



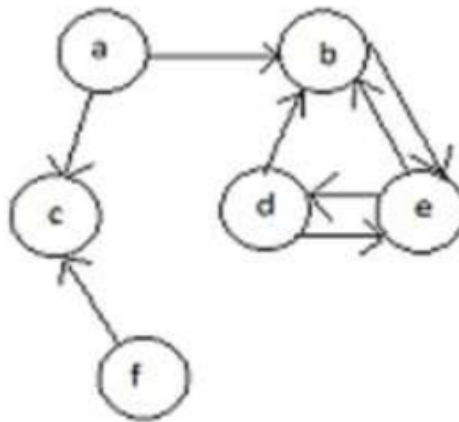
- a. Indicar si es un ABB. Justificar.
- b. Indicar cómo queda la salida con un recorrido en pre orden.

4) Sea el array de bits

1	0	0	1	1	0	1	1
---	---	---	---	---	---	---	---

- a. Indicar qué elementos se encuentran.
- b. Qué pasos y operaciones se deben hacer para
 - i. Dar de alta al valor 5.
 - ii. Dar de baja al valor 3.
 - iii. Consultar si está el valor 4.

5) Dado el grafo dirigido:



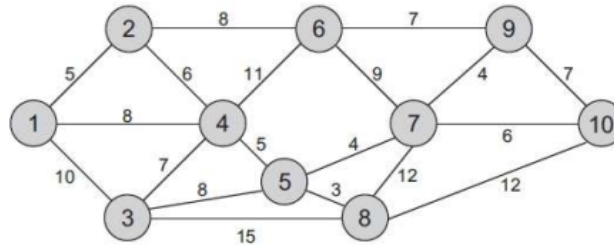
- a. Indicar cómo es la matriz de adyacencia.
- b. Indicar cómo queda el recorrido en profundidad primero. Imprimir los vértices y explicar el recorrido habiendo elegido la matriz de adyacencia o la lista de adyacencia.

6) Sea una estructura de tipo heap de mínima

- a. Indicar cuánto es el costo de encontrar el mínimo elemento.
- b. Indicar cuánto es el costo de insertar un elemento.

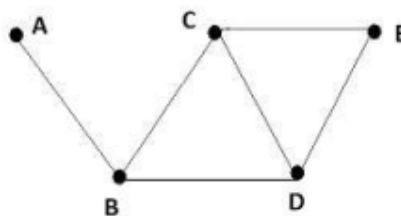
- 7) Sea el vector $v = [150, 31, 83, 325, 46, 186, 222]$
- Explicar cómo funciona el algoritmo de ordenamiento heap.
 - Ordenar el vector, indicando cómo queda en cada paso, aplicando dicho algoritmo.

- 8) Explicar la diferencia entre el algoritmo de Prim y el de Kruskal para encontrar un árbol de expansión mínimo.
- Aplicar ambos al siguiente grafo indicando cómo se construye paso por paso.



- 9) Nombre y explique brevemente las funciones de hashing que conoce.
- 10) ¿Qué es una colisión? ¿Qué formas de resolverla conoce? Nombrar y explicar.
- 11) Definir árbol de expansión mínimo de un grafo.

Dado el siguiente grafo



Encontrar todos los árboles de expansión mínimo (las aristas tienen peso 1).

- 12) Programar en C++ una función recursiva que calcule la suma de los nodos internos (no hojas) de un árbol binario de enteros.