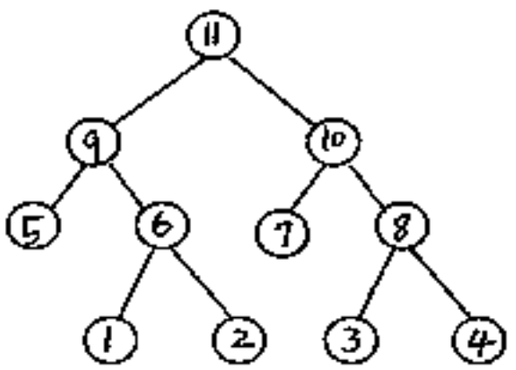
**1. (10%)** What are the sequences of nodes encountered when traveling the following tree in *Preorder*, *Inorder*, and *Postorder* ?




**2. (10%)** Consider the following two orders for traveling binary tree:

(a) (1) Traverse the right subtree
(2) Visit the root
(3) Traverse the left subtree

(b) (1) Visit the root
(2) Traverse the right subtree
(3) Traverse the left subtree

Are there sample relationships between the sequences of nodes encountered following these orders and those generated by the *Preorder*, *Inorder*, and *Postorder* ?

**3.(10%)** The Fibonacci numbers are defined by the recurrence relation

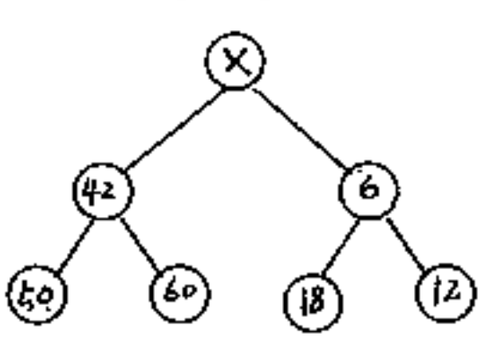
$fib_{n+1} = fib_n + fib_{n-1}$ for $n > 0$ and $fib_1 = 1$, $fib_0 = 0$.

A direct, native approach leads to the following program

```
Function  Fib(n: integer):integer
Begin if n = 0  then  Fib := 0 else
        if n = 1  then  Fib := 1 else
        Fib := Fib (n-1) + Fib (n-2)
End
```

Such a recursion program is clearly impractical due to the total number of calls grows exponentially. Please design an interactive scheme to compute the Fibonacci numbers without using the recursion.

**4.(10%)** What is *Heap* ? For the following heap with seven elements, what is the new heap when x = 44 ?



5.(10%) The B-trees are defined as follows; where $n$ is said to be the order of the B-tree.

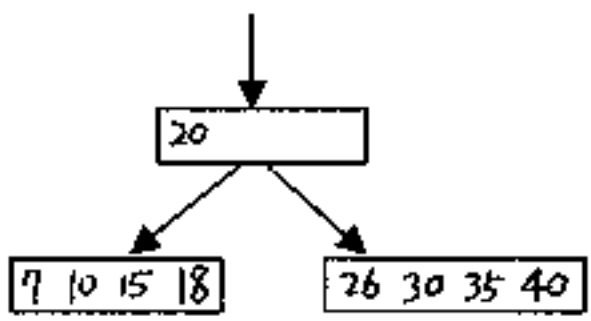
(a) Every page contains at most $2n$ items (keys).

(b) Every page, except the root page, contains at least n items.

(c) Every page is either a leaf page, i.e., has no descendants or it has $m$+1 descendants, where $m$ is its number of keys.

(d) All leaf pages appear at the same level.

For example, the following figure shows a B-tree of order 2 with 2 levels.




What is the new B-tree when a key 22 is inserted into the above B-tree ?

6. (10%) Let L, R, and X denote respectively the operations of inserting an element at the left, inserting an element at the right, and emitting an element from the left, of an output-restricted double-ended queue. Find a way to define the concept of an admissible sequence of the symbols L, R, and X in such a way that each admissible sequence performs a meaningful sequence of operations.

7. (10%) Design a method to represent circular lists inside a computer in such a way that the list can be traversed efficiently in both directions, yet only one link field is used per node.

8. (5%) Show that if we are given the preorder and the inorder of the nodes of a binary tree, the binary tree structure may be constructed.

9. (15%) Find all binary trees whose nodes appear in exactly the same sequence in both
   (a) Preorder and inorder
   (b) Preorder and postorder
   (c) Inorder and postorder

10. (10%) Given a large number of distinct 30-bit binary numbers, $x_1$, $x_2$, ..., $x_N$, design an efficient method to find all complementary pairs $(x_i, x_j)$ that are present. Do not compare all pairs of $(x_i, x_j)$ exhaustively. (Two binary numbers are complementary when one has 0 whenever the other has 1 at the same bit position.)