1-An IT professional received a digital file, from the company's database sector, containing 1,048,575 different numeric keys, to be stored in a binary search tree type data structure. After creating a program to perform the task and inserting all the keys into the structure, it was found that the resulting tree was full. Given this result, it is correct to say that the height of this tree is

a- 10
b-15
c-20
d-25
e-30

2-Considering the data structure called tree,

a- its height is defined as the average depth of all its vertices.
b- a vertex with one or two children is called a leaf.
c- each node has at least two children in a binary tree.
d- the leaves of a complete binary tree can have different depths. and - the depth of a vertex in a tree is defined as the length from the root of the tree to that vertex.

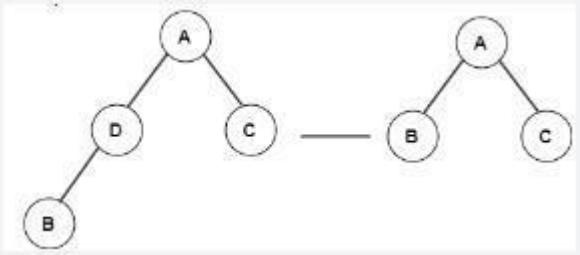
3-The number of edges from the root to the node is called the _____ of the tree.

a- height
b- Depth
c- Length
d - Width

4- The number of edges from the node to the deepest leaf is called _____ of the tree.

a- height
b- Depth
c- Length
d - Width

5- What operation does the following diagram represent?



a) inserting a leaf node
b) inserting an internal node
c) delete a node with 0 or 1 children
d) delete a node with 2 children

6- What is the average time complexity in the case to find the height of the tree binary?

a-h = O(log n)
b- h = O(n log n)
c-h = O(n)
d- h = $O(n^2)$

7- Which of the following is not an advantage of trees?

a- Hierarchical structure
B- Faster search
c- Router algorithms
d- Uses little memory to build the data structure

8- Uses little memory to build the data structure

a-Each node has exactly zero or two children
b-Each node has exactly two children
c-All sheets are on the same level
d-Each node has exactly one or two children

9-What does this piece of code do

```
public void func(Tree root)
{
        func(root.left());
        func(root.right());
        System.out.println(root.data());
}
```

a) Pre-order routing
b) Internal routing
c) Post-order routing
d) transversal level order

10-What does the code do?

empty publicfunc(Tree root) { System.Fora.println(root.data());
func(root.left()); func(root.right()); }

a) Pre-order routing
b) Internal routing
c) Post-order routing
d) transversal level order

11-Regarding binary trees, it is INCORRECT to state:

a- A binary tree is a finite collection of n>0 nodes that cannot be null.

b- A binary tree, whose root stores the element R, is called a binary tree. binary search if every element stored in the left subtree is smaller than R, no element stored in the right subtree is smaller than R and the subtrees left and right are also binary search trees.

c- It is a special case of tree in which no node has a degree greater than 2, that is, no node has more than two children.

d- There is a special node called root and the other nodes are partitioned in T1 and T2 disjoint binary tree structures. T1 is called subtree left and T2 right subtree of the root.

e- It is a tree that can be null.

12-In accordance with CORMEN et al., consider a data structure linked, in which each node is an object. In addition to a key and satellite data, each node contains attributes “left”, “right” and “p”, which point to the nodes corresponding to his son on the left, his son on the right and his father, respectively. This structure refers to:

a-Binary search trees.

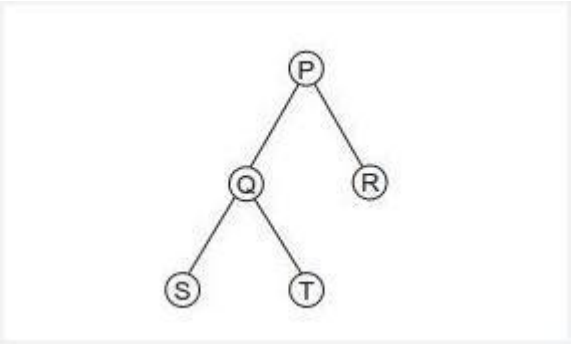
b-Stack.

c-Queues.

d-Linked lists.

e-Arrays.

13- The sequence that represents the path of the tree of the figure in post-order is



a- PQSTR

b- STQPR

c- PQRST

d- RPQTS

e- STQRP

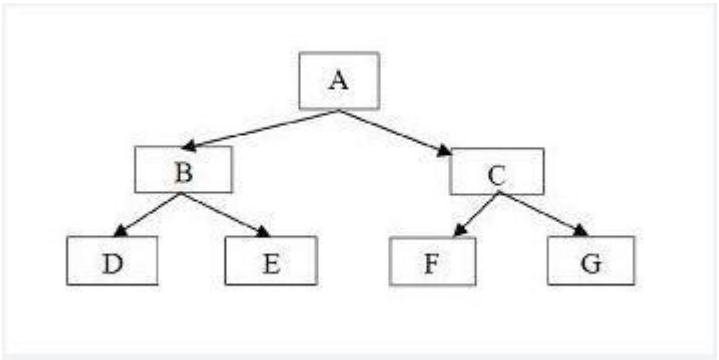
14- The data structure*tree*inherits the characteristics of the topologies in *tree*, whose data are arranged in a hierarchical manner, with the main element a*source*that connects to other elements through its
branches. After analysis by the development team, it was observed that this structure is widely used in various situations such as ordering operating system folders, graphical interfaces and databases; therefore, the team decided that the structure could be perfectly employed within the development project of a new application
financial control to be developed. Regarding the type of structure, analyze the statements below.

I. The*us*who do not have*children*are called*we leaf.*
II. The height of a*tree*represents the distance between the root and a*leaf node*bigger level of*tree*.
III. The degree is the property that qualifies the*us*of a*tree*, defining the quantity of*children*that each*at the*it has.

What is stated in

a - I, II and III.
b - I, just.
c-II, only.II, only.
d-III, only.
e - I and II, only.

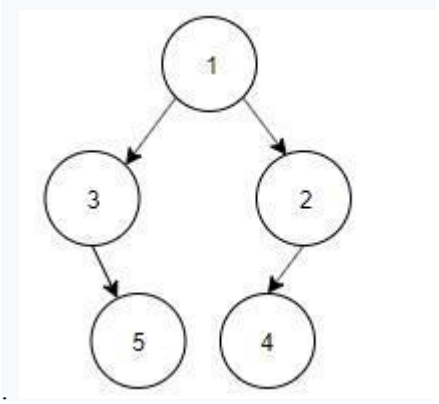
15 -Consider the following linked binary tree. When applying the type route "Pre-Order", the linear sequence is obtained:



a-ABDECFG.
b-DBEAFCG.
c-DEBFGCA.
d-DEFGBCA.
e-BCADEFG.

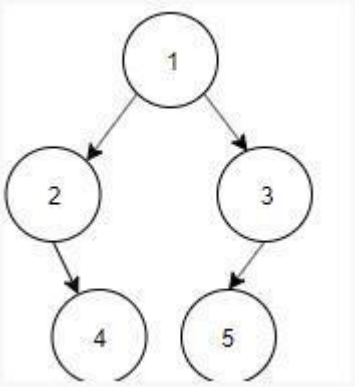
16- Build a binary tree using the following data.
The traversal order of a binary tree is 1, 2, 5, 3, 4. The offset
internal of the same binary tree is 2, 5, 1, 4, 3

The-

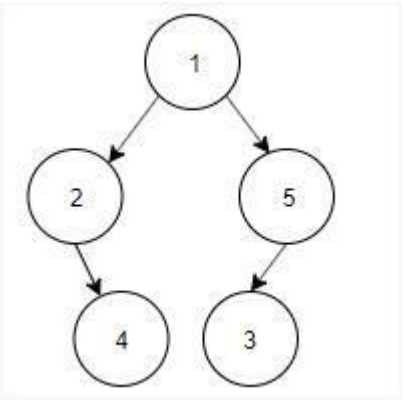


.

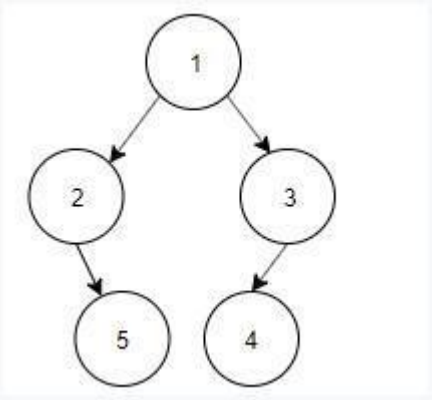
B-



w-



d-



17- Which of the following options is incorrect in relation to binary trees?

a) Let T be a binary tree. For every k ≥ 0, there are no more than 2k nodes at level k
b) Let T be a binary tree with levels λ. Then T has no more than 2λ - 1 knots
c) Let T be a binary tree with N nodes. Then the number of levels is at least ceil (log (N + 1))
d) Let T be a binary tree with N nodes. Then the number of levels is at least the floor (log (N + 1))

18- What logic should be missing in place of the missing lines to find the sum of binary tree nodes at alternative levels?

```
    // logica ausente
}
```

ai=i+pow(2,current level);current level=current level+2; j=1;

bi=i+pow(2,current level);current level=current level+2; j=0;

ci=i-pow(2,current level);current level=current level+2; j=1;

di=i+pow(2,current level);current level=current level+1; j =1;

19-How to search for keys in a binary tree

a-

```
public Tree search(Tree root, int key)
{
        if( root == null || root.key == key )
        {
                return root;
        }
        if( root.key < key )
        {
                return search(root.right,key);
        }
        else
        return search(root.left,key);
}
```

b-

```
public Tree search(Tree root, int key)
{
        if( root == null || root.key == key )
        {
                return root;
        }
        if( root.key < key )
        {
                return search(root.left,key);
        }
        else
        return search(root.right,key);
}
```

c-

```
public Tree search(Tree root, int key)
{
        if( root == null)
        {
                return root;
        }
        if( root.key < key )
        {
                return search(root.right,key);
        }
        else
                return search(root.left,key);
}
```

d-

```
public Tree search(Tree root, int key)
{
        if( root == null)
        {
                return root;
        }
        if( root.key < key )
        {
                return search(root.right.right,key);
        }
        else
                return search(root.left.left,key);
}
```

20- What are the conditions for an ideal binary search tree and what is the your advantage?

a) The tree should not be modified and you should know how often keys are accessed as it improves the search cost

b) You must know the access frequency of the keys, it improves the time of search

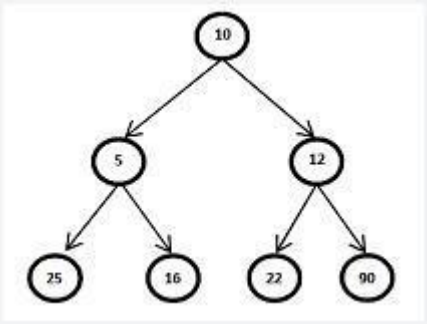
c) The tree can be modified and you must know the number of elements in the tree before hand, improves deletion time

d) The tree must be modified and improves search time

21 - How to remove an element that has two children in a tree binary?

a-Erasing only the two sheets.
bThe parent points to the child being removed.
c-Replacing the value of the node with that of its largest child.
d-It cannot be removed as it is impossible.

22-Given the following binary search tree, select the alternative that presents the insertion sequence that generates it.



a-10, 5, 22, 25, 16, 12, 90.
b-10, 5, 22, 25, 16, 20, 12 .
c-10, 5, 12, 25, 16, 22, 90.
d-10, 12, 16, 90, 5, 22, 25.
e-10, 12, 22, 5, 25, 16, 90.