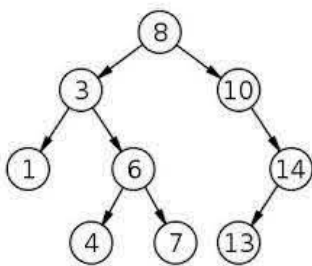


**Started on** Sunday, 30 October 2022, 2:56 PM**State** Finished**Completed on** Sunday, 30 October 2022, 3:02 PM**Time taken** 6 mins 38 secs**Marks** 5,00/5,00**Grade** 20,00 out of 20,00 (100%)**Question 1**

Correct

Mark 1,00 out of 1,00

Indique qual a sequência see percorrer a árvore em **pre-order**

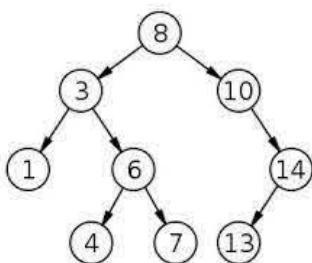
Select one:

- ☐ a. A, B, C, D, E, F, G, H
- ☐ b. A, B, E, C, F, G, H, D
- ☒ c. Nenhuma das anteriores ✓
- ☐ d. E, B, F, G, H, C, D, A

**Question 2**

Correct

Mark 1,00 out of 1,00

Indique a quantidade de **nós externos** da árvore da figura

Select one:

- ☒ a. 4 ✓
- ☐ b. 6
- ☐ c. 5
- ☐ d. Nenhum dos anteriores

**Question 3**

Correct

Mark 1,00 out of 1,00

O seguinte pseudocódigo especifica o algoritmo que determina o número de nós externos da árvore.

```

Algorithm: count_external
input: tree
output: natural
BEGIN
  IF is_empty(tree) THEN RETURN 0
  count ← 0
  IF isEmpty(children(tree)) THEN
    /**A**/
  FOR EACH child FROM children(tree)
    /**B**/
  RETURN count
END

```

Indique quais das instruções completam o código:

Select one:

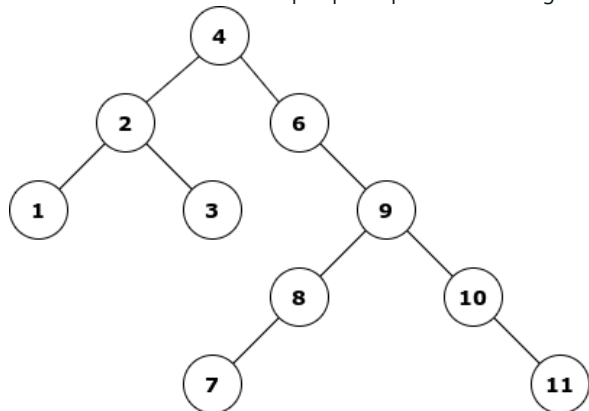
- ☐ a. [A]count ← 1 [B]count ← count + 1
- ☐ b. Nenhuma das anteriores
- ☒ c. [A]RETURN 1 [B]count ← count + count\_external(child) ✓
- ☐ d. [A]RETURN 1 [B]RETURN 1 + count\_external(child)

**Question 4**

Correct

Mark 1,00 out of 1,00

Considere a árvore binária de pesquisa apresentada na figura seguinte:



Numa *travessia pré-ordem*, qual a ordem pela qual os nós são visitados?

Select one:

- ☐ a. 4 2 1 3 6 9 10 11 8 7
- ☐ b. Outra resposta
- ☐ c. 4 2 6 1 3 9 8 10 7 11
- ☒ d. 4 2 1 3 6 9 8 7 10 11 ✓

## Question 5

Correct

Mark 1,00 out of 1,00

Considere a seguinte estrutura de dados para uma árvore binária de pesquisa:

```
public class SomeClass<T extends Comparable<T>> {  
    private BSTNode root;  
  
    //métodos...  
  
    private class BSTNode {  
        private T element;  
        private BSTNode left;  
        private BSTNode right;  
    }  
}
```

Para a implementação de um método recursivo que calcule quantos elementos da árvore são *maiores* que *threshold*, tem-se o seguinte código:

```
private int count(T threshold, BSTNode treeRoot) {  
    if( __A__ == null) return 0;  
    return (treeRoot.element.compareTo(threshold) __B__ 0 ? 1 : 0)  
        + count(threshold, __C__ )  
        + count(threshold, __D__);  
}
```

Qual a resposta que contém o código das letras em falta?

Select one:

- ☐ a. A: `this.root` | B: `<` | C: `treeRoot.left` | D: `treeRoot.right`
- ☐ b. A: `treeRoot` | B: `<` | C: `treeRoot.left` | D: `treeRoot.right`
- ☐ c. A: `this.root` | B: `>` | C: `treeRoot.right` | D: `treeRoot.left`
- ☒ d. A: `treeRoot` | B: `>` | C: `treeRoot.right` | D: `treeRoot.left` ✓