Practical 3

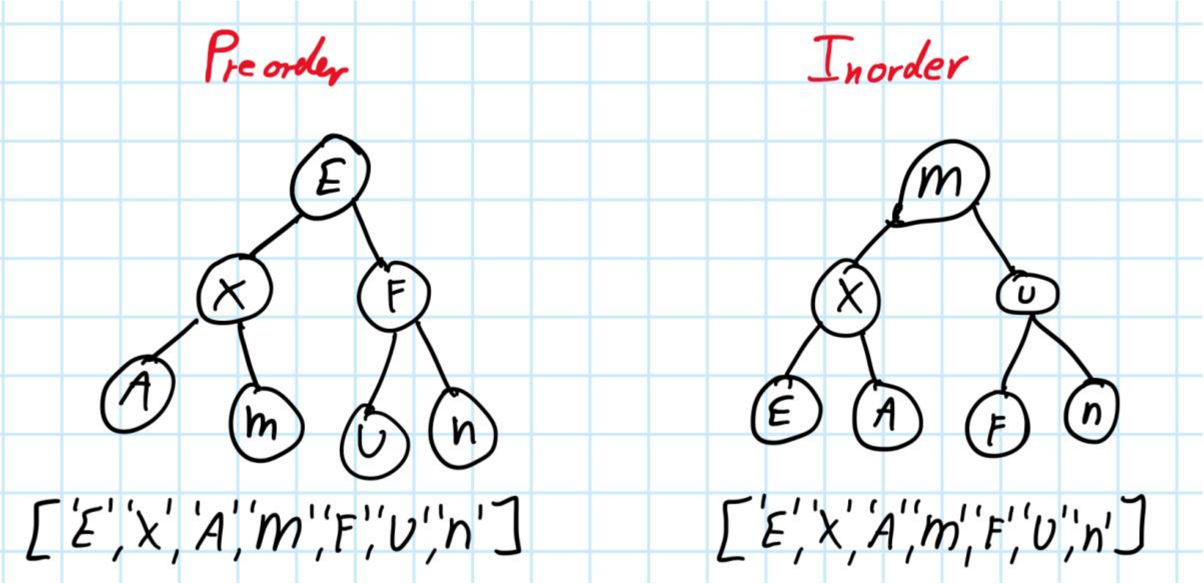
Q2.

Code for a function that only counts left external nodes. The code is below and the only change I made from the code of counting all external nodes is when the count external is called recursively only go down the parent.getLeft() path. This will only count left external nodes.

```
private int countExternal(Position<E> p) {
    if (p == null) {//base case
        return 0;
    }
    Node<E> parent = validate(p);

    if (isExternal(parent)) {
        return 1;
    } else {//if not external count the external nodes of both left and
right children
        return countExternal(parent.getLeft());
    }
}
```

Q3.



Q4.

Pseudocode for an algorithm to count the total number of descendants a node has.

```
If node is null:
        Return 0
Else:
Return 1 + numChildern(node.left) + numChildren(node.right)
```

This recursive function will count the number of descendant nodes including the one passed to the function as it calls the function numChildren on the left and right child of the node.