7 Algorithms (FMS)

This question is about Binary Search Trees (BSTs) and Red-Black Trees (RBTs).

(*a*) Using a diagram, explain what a BST rotation is and its purpose. [3 marks]

(*b*) Consider the following buggy pseudocode.

```
0  def mystery(x):
1      y = x.r
2      x.r = y.l
3      if y.l != null:
4          y.l.p = x
5      x.p = y.p
6      if x == x.p.l:
7          x.p.l = y
8      else:
9          x.p.r = y
10     y.l = x
```

(*i*) Explain what it intends to do, give it a meaningful name, describe all the identifiers used (`x`, `y`, `r`, `l`, `p`) and the (intended) precondition and postcondition of the routine. [4 marks]

(*ii*) Identify, explain and fix the bugs, one by one, referring to a diagram if useful. Finally, give a fully corrected version of the code. [8 marks]

(*c*) State, with a proof or counterexample as appropriate, whether each of the following statements is true or false.

(*i*) In an RBT with more than one node, at least one node is red. [2 marks]

(*ii*) In a BST with n nodes, exactly $n - 1$ rotations are possible. [3 marks]