ADS BONUS HOMEWORK 9

Problem 9.1

C) (BONUS)

Proof by Induction:

For our Induction proof, we will take the simple base case that n = 2. In this base case, the first element inserted is the root which will be black and now, since every newly inserted element will have to be red, the second element will have to always be red.

Now, assume that for a red-black tree formed by inserting n nodes where n>1, there is at least one red node present in the tree and consider the two cases that occur while inserting the (n+1)th node:

1. If the red node is inserted as a child of a black node, this is the base case we stated. Hence the statement that there is at least one red node in a red-black tree holds true.
2. If the red node inserted is a child of a red node, then since there are two consecutive red nodes present now, we will need to make changes to the tree. And again, we consider two subcases of this scenario. Labelling the different parties, let N be the newly inserted node, RP its red parent, GP its grandparent and U its uncle.
    - In the case of a red U, N remains red after the recoloring of RP, GP, and U. (Since only RP, GP and U will be recolored, N remains in the original color which is red.)
    - If U is black we would have to rotate N, RP and GP. After rotation, we will have to recolor these nodes in such a way that the parent is black and the children are all red.

As can be seen in the above induction proof, in all cases, there will always be at least one red node in the Red-Black Tree. Therefore, the statement in the question "For n>1, a Red-Black Tree contains at least one Red node" holds.