**R-3.11 Answer:**

|  |  |
| --- | --- |
| a | C:\Users\Jainal Uddin\Downloads\Shareit\Photo\20150308_191111 (2).jpg |
| b | C:\Users\Jainal Uddin\Downloads\Shareit\Photo\20150308_191111 (3).jpg |

**R-3.14 Answer:**

|  |  |
| --- | --- |
| a | False.  Reason: To be a red-black root has to be black but there is no guarantee root of subtree will be black. It might be red or black. |
| b | True.  Reason: if by another external node it is a black external nodes, because there's a rule that all external nodes are black. And also it can be red, because when we insert, or doing recolor, or restructure, the node will still be red. |
| c | False.  Reason: Every red-black tree can become (2,4) tree, and the other way around. But we can make the tree with the same red black tree and produce a different (2,4) tree, even though it's result will be the same. So basically, it's not unique. |
| d | False.  Reason:  Every (2,4) tree can become red-black tree, and the other way around. But we can make the tree with the same (2,4) tree and produce a different red black tree, even though it's result will be the same. So basically, it's not unique. |

**C-3.10 Answer:**

|  |  |
| --- | --- |
| Algorithm findAllInRange**(**k1**,**k2**)**  Input**:** key k1**,** k2  Ouput**:** return iterator **for** all the elements in D within the range of k1 and k2  T**<-** tree of D  S**<-**findElements**(**T**,**T**.**root**(),**k1**,**k2**)**  **return** S**.**iterator**()** | Algorithm findElements**(**T**,**p**,**k1**,**k2**)**  Input**:** Tree T**,** position of a node p**,** key k1**,** k2  Output**:** Sequence S with all the elements between the range of k1 and k2 inclusive**.**  S**<-new** Sequence  k **<-** T**.**key**(**p**)**  **if** k1 **<=** k **^** k **<=** k2 then  S**.**insertLast**(**D**.**findElement**(**k**))**  findElements**(**T**,**T**.**leftChild**(**p**),**k1**,**k2**)**  findElements**(**T**,**T**.**rightChild**(**p**),**k1**,**k2**)**  **else** **if** k **<** k1 then  **return** findElements**(**T**,**T**.**leftChild**(**p**),**k1**,**k2**)**  **return** S |