Extra Credit Assignment – Red Black Tree Implementation to store processes – Sonia Kargutkar

In this program, a red-black tree data structure is used for storing processes.
The tree is created using the newTree() method.

The following operations have been performed:

1. Inserting a key value using the tr_insert() method:

   Here, we are initializing node color as RED for the node to be inserted. The left and right child nodes are assigned null values, and the node key value is assigned -1. We create two new nodes, n1 and n2, which act as pointers for traversing the tree. The node n1 is assigned to root node and node n2 is assigned null. BST insert algorithm is used for insertion whereby, the value of the node to be inserted is compared to the root node. So, if n1 is not -1, it means the tree is not empty, then n2 is assigned the root node and BST insert algorithm is used for inserting the key. If n1 is -1 and n2 is -1, it means the tree is empty and key value is assigned to the root node. If the value of n2 is not -1, it means the tree is not empty and new key is inserted using the BST insert algorithm. Inserting new nodes could imbalance the tree and this is handled by calling the rbTreeRotation() method.

2. Searching for a node is done by using the tr_search() method

   Here, a node n is assigned the root node. The key value passed to this method is compared with the value of the root node and based on whether it is lesser or greater than the root node, either the left or right side of the tree is searched until the node matching the key value is found or until the end of the tree is reached.

3. Deleting a node value is done using the tr_delete() method

   Here, the BST deletion algorithm is applied whereby if the node to be deleted has no children then the node is deleted. Else, the node is replaced by it's child node. Nodes n and temp are used to track the node to be deleted and the node to be replaced respectively, basically acting as leading and trailing pointers. The transplant() method is used for node replacement. As deletion of nodes in a red black tree could violate the properties of the red-black tree, a variable color is used to track the node color. The method fixDelete() is used to fix any violations that may arise.

4. Processes are stored using the storeProcess() method

   Here, the prev variable is used to store the value of the process ID. This value is then passed to tr_insert() method to be store in the Red Black Tree structure.

5. Values of the tree are printed using the printTree() method.