SATYAM TRIPATHI
202151141

LAB 10

CODE

KRUSKAL'S ALGO

```
import java.util.*;
import java.lang.*;
import java.io.*;


public class Main{
    public static void main (String[] args) {
        // ARRAY IS IN THE FORM <VERTEX 1> <EDGE LENGTH>
<VERTEX 2>
        int arr[]={2,4,3,0,5,3,0,10,1};
        int edges[]=new int[arr.length/3];
        int vertices[][]=new int[arr.length/3][2];
        for(int i=0;i<9;i+=3){
            edges[i/3]=arr[i+1];
            vertices[i/3][0]=arr[i];
            vertices[i/3][1]=arr[i+2];
        }
        for(int i=0;i<edges.length;i+=1){
            int min=edges[i];
            int pos=i;
            for(int j=i+1;j<edges.length;j+=1)
            {
                if(min>edges[j]){
                    min=edges[j];
                    pos=j;
                }
            }
            int temp=edges[i];
```

SATYAM TRIPATHI
202151141

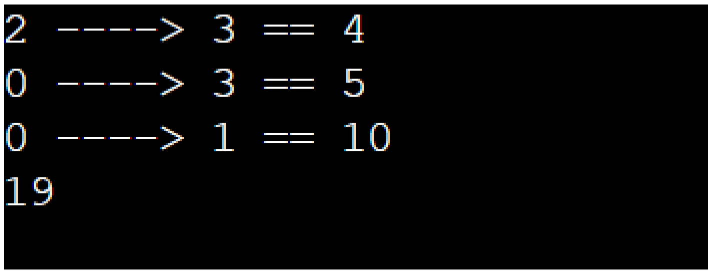
```
            edges[i]=edges[pos];
            edges[pos]=temp;
            int temp1=vertices[pos][0];
            int temp2=vertices[pos][1];
            vertices[pos][0]=vertices[i][0];
            vertices[pos][1]=vertices[i][1];
            vertices[i][0]=temp1;
            vertices[i][1]=temp2;


        }

        HashSet<Integer> map=new HashSet<>();
        int sum=0;
        for(int i=0;i<edges.length;i+=1){
            if(map.contains(vertices[i][0]) &&
map.contains(vertices[i][1])){
                continue;
            }
            else{
                System.out.println(vertices[i][0]+" ---->
"+vertices[i][1]+" == "+edges[i]);
                sum+=edges[i];
                map.add(vertices[i][0]);
                map.add(vertices[i][1]);
            }
        }
        System.out.println(sum);
    }
}
```

OUTPUT

SATYAM TRIPATHI
202151141



```
2 ----> 3 == 4
0 ----> 3 == 5
0 ----> 1 == 10
19
```

SATYAM TRIPATHI
202151141

RED BLACK TREE

CODE

```java
class Node
    {
        int data;
        Node left;
        Node right;
        char colour;
        Node parent;

        Node(int data)
        {

            this.data = data;
            left = null;
            right = null;
            colour = 'R';
            parent = null;
        }
    }

class redblack
{
    public Node root;


    Node rotateLeft(Node node)
    {
        Node x = node.right;
        Node y = x.left;
```

SATYAM TRIPATHI
202151141

```
        x.left = node;
        node.right = y;
        node.parent = x;
        if(y!=null)
            y.parent = node;
        return(x);
    }

    Node rotateRight(Node node)
    {
        Node x = node.left;
        Node y = x.right;
        x.right = node;
        node.left = y;
        node.parent = x;
        if(y!=null)
            y.parent = node;
        return(x);
    }

    boolean ll = false;
    boolean rr = false;
    boolean lr = false;
    boolean rl = false;
    Node insertNode(Node root, int data)
    {

        boolean f=false;
        if(root==null)
            return(new Node(data));
        else if(data<root.data)
        {
            root.left = insertNode(root.left, data);
```

SATYAM TRIPATHI
202151141

```
            root.left.parent = root;
            if(root!=this.root)
            {
                if(root.colour=='R' &&
root.left.colour=='R')
                    f = true;
            }
        }
        else
        {
            root.right = insertNode(root.right,data);
            root.right.parent = root;
            if(root!=this.root)
            {
                if(root.colour=='R' &&
root.right.colour=='R')
                    f = true;
            }
        }
        if(this.ll)
        {
            root = rotateLeft(root);
            root.colour = 'B';
            root.left.colour = 'R';
            this.ll = false;
        }
        else if(this.rr)
        {
            root = rotateRight(root);
            root.colour = 'B';
            root.right.colour = 'R';
            this.rr = false;
        }
```

SATYAM TRIPATHI
202151141

```
        else if(this.rl)
        {
            root.right = rotateRight(root.right);
            root.right.parent = root;
            root = rotateLeft(root);
            root.colour = 'B';
            root.left.colour = 'R';

            this.rl = false;
        }
        else if(this.lr)
        {
            root.left = rotateLeft(root.left);
            root.left.parent = root;
            root = rotateRight(root);
            root.colour = 'B';
            root.right.colour = 'R';
            this.lr = false;
        }

        if(f)
        {
            if(root.parent.right == root)
            {
                if(root.parent.left==null ||
root.parent.left.colour=='B')
                {
                    if(root.left!=null &&
root.left.colour=='R')
                        this.rl = true;
                    else if(root.right!=null &&
root.right.colour=='R')
                        this.ll = true;
```

SATYAM TRIPATHI
202151141

```
                }
                else
                {
                    root.parent.left.colour = 'B';
                    root.colour = 'B';
                    if(root.parent!=this.root)
                        root.parent.colour = 'R';
                }
            }
            else
            {
                if(root.parent.right==null ||
root.parent.right.colour=='B')
                {
                    if(root.left!=null &&
root.left.colour=='R')
                        this.rr = true;
                    else if(root.right!=null &&
root.right.colour=='R')
                        this.lr = true;
                }
                else
                {
                    root.parent.right.colour = 'B';
                    root.colour = 'B';
                    if(root.parent!=this.root)
                        root.parent.colour = 'R';
                }
            }
            f = false;
        }
        return(root);
    }
```

SATYAM TRIPATHI
202151141

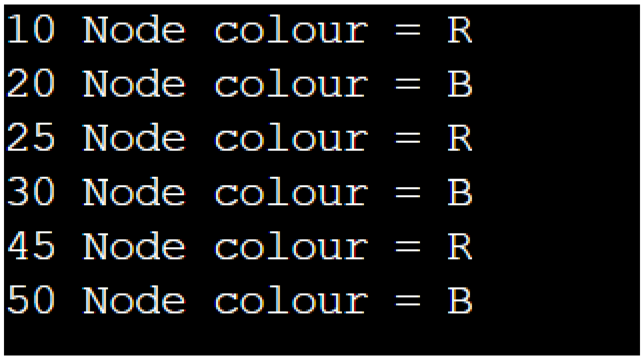
```java
    public void insert(int data)
    {
        if(this.root==null)
        {
            this.root = new Node(data);
            this.root.colour = 'B';
        }
        else
            this.root = insertNode(this.root,data);
    }
    void inorderTraversalHelper(Node node)
    {
        if(node!=null)
        {
            inorderTraversalHelper(node.left);
            System.out.println( node.data+" "+"Node colour
= "+node.colour);
            inorderTraversalHelper(node.right);
        }
    }
    public void inorderTraversal()
    {
        inorderTraversalHelper(this.root);
    }


}
public class Main{
    public static void main(String[] args)
    {
        redblack t = new redblack();
        t.insert(30);
```

SATYAM TRIPATHI
202151141

```
        t.insert(50);
        t.insert(20);
        t.insert(25);
        t.insert(45);
        t.insert(10);
        t.inorderTraversal();
    }
}
```

OUTPUT



```
10 Node colour = R
20 Node colour = B
25 Node colour = R
30 Node colour = B
45 Node colour = R
50 Node colour = B
```