Algorithma

<multiple algorithms in one package>

PROJECT REPORT

Submitted in fulfilment for the J-Component of

Advance java Programming (ITE2005)

CAL COURSE

in

B.Tech – Information Technology

by

SOURABH AGRAWAL 15BIT0298

RUBAL NANDAL 15BIT0134

Under the guidance of

Prof. VARALAKSHMI M

SITE

School of Information Technology and Engineering

Winter semester 2016-2017

TABLE OF CONTENTS

ACKNOWLEDGMENT

I wish to extend my heartiest gratitude to our faculty Prof. VARALAKSHMI M, who was kind and flexible enough to help us develop the java based application - Algorithma without any pressure and hesitation. Her leniency with a touch of strictness is something without which the project would not have been as successfully implemented as it is now. I sincerely thank and congratulate the team members.

Algorithms- it has multiple algorithm, Dijkstra algorithm and prims and insertion sort, with it we can show the difference b/w both algorithms in a simplest way of animation. Insertion sort will show how sorting happen at each and every step.

INTRODUCTION

Our project- Algorithma is basically a platform where we are trying to implement 3 algorithms called “Dijkstra algorithm”, “Prims algorithm” and “insertion sort”.

Algorithma starts with a Log-in page where we are performing some validation and from this page new users can sign-up to this application.

Sign-up page also has some validation like- “checking redundancies, validating password, and confirm password”.

After successfully Log-in new jFrame will open which is our Homepage of the Algorithms. Here user can go with “Dijkstra algorithm” or “Prims algorithm” or with “insertion sort”. Selecting each algorithm will take you to a new jFrame where animation will start accordingly by which anyone can easily understand all the algorithms in a simplest way.

Dijkstra’s algorithm is implemented by simple awt event handling by using buttons and prims algorithm is implemented by using Threads.

Insertion sort is happening with the use of thread and values are presenting with the help of bar chart.

APPLICATION Contain:

(i) Login page.
(ii) Sign-up page.
(iii) Home-page.
(iv) Dijkstra algorithm jframe.
(v) Prims algorithm jframe.
(vi) Insertion sort jframe.

APPLICATION DESIGN

Used jFrames files:

1. **Log_in.java**
2. **Sign_up.java**
3. **Home_page.java**
4. **Dijkstar1.java**
5. **Prims.java**
6. **Insertionsort.java**

Codes:

1. **Log_in.java:**

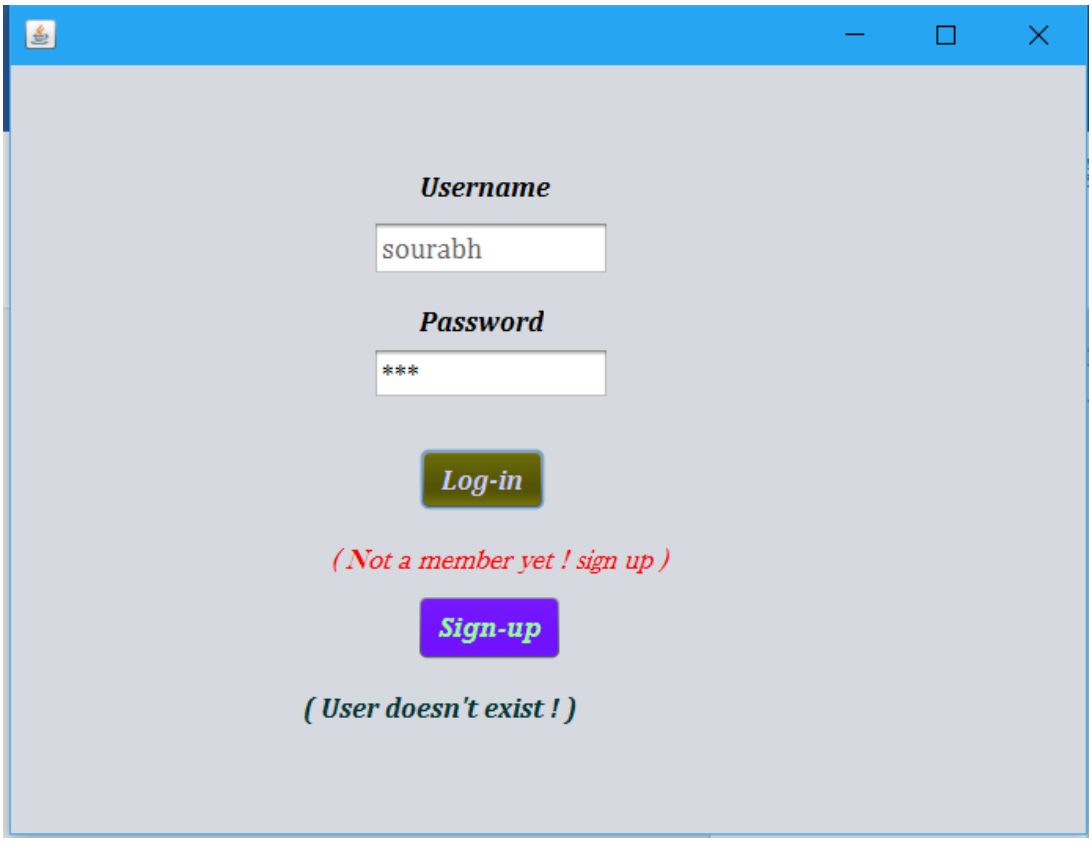
Explanation:

we used some effects like:

- When text field getting focused then the background colour of the text field will get change. For this purpose, we simply add FocusListner with the text field and calling focusGained and focusLost method. In the focusGained method I wrote the code to change the background colour to some other colour and in the focusLost method I reset the code to white colour.
- For password, I used passwordfield so that I don't have to use method to set * symbol for typing password.
- Log-in page has some validations like:
    - If username and password are empty then page will show error message like

For this purpose, I used if else loops to ensure that the log-in will happen only if username and password fields are not null

- Other validation is like if user enter wrong password and username then my code will check the database and move accordingly. If the user details are correct then it will move to next page otherwise it will again show error message like:



All these validations and moving to next page is happening only by clicking "Log-in" page because I add MouseListner with the log-in button and calling mouseClicked method so that whenever I clicked on log-in button then only it will move further.

All the validation or moving to the next page's code is written inside the mouseClicked method.

Code:

```
/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
import java.sql.*;

/**
 *
 * @author sourabh
 */
public class Log_in extends javax.swing.JFrame {
    int flag =0;
```

```
    /**
     * Creates new form Sign_up
     */
    public Log_in() {
        initComponents();
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method
is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        Textfield_username = new javax.swing.JTextField();
        Passwordfield_password = new javax.swing.JPasswordField();
        Log_in_Button = new javax.swing.JButton();
        Sign_up_Button = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        Username = new javax.swing.JLabel();
        Password = new javax.swing.JLabel();
        comment = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_C
LOSE);

        Textfield_username.setFont(new java.awt.Font("Cambria", 0, 18));
// NOI18N
        Textfield_username.setForeground(new java.awt.Color(102, 102,
102));
        Textfield_username.addFocusListener(new
java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                Textfield_usernameFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent evt) {
                Textfield_usernameFocusLost(evt);
            }
        });
        Textfield_username.addKeyListener(new
java.awt.event.KeyAdapter() {
            public void keyPressed(java.awt.event.KeyEvent evt) {
                Textfield_usernameKeyPressed(evt);
            }
```

```
        });

        Passwordfield_password.setFont(new java.awt.Font("Cambria", 0,
18)); // NOI18N
        Passwordfield_password.addFocusListener(new
java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent  evt) {
                Passwordfield_passwordFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent  evt) {
                Passwordfield_passwordFocusLost(evt);
            }
        });

        Log_in_Button.setBackground(new  java.awt.Color(51, 51, 0));
        Log_in_Button.setFont(new  java.awt.Font("Cambria", 3, 18)); //
NOI18N
        Log_in_Button.setForeground(new  java.awt.Color(204, 204,
255));
        Log_in_Button.setText("Log-in");
        Log_in_Button.addMouseListener(new
java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent  evt) {
                Log_in_ButtonMouseClicked(evt);
            }
        });

        Sign_up_Button.setBackground(new  java.awt.Color(102, 0, 255));
        Sign_up_Button.setFont(new  java.awt.Font("Rockwell", 3, 18)); //
NOI18N
        Sign_up_Button.setForeground(new  java.awt.Color(153, 255,
153));
        Sign_up_Button.setText("Sign-up");
        Sign_up_Button.addMouseListener(new
java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent  evt) {
                Sign_up_ButtonMouseClicked(evt);
            }
        });

        jLabel1.setFont(new java.awt.Font("Baskerville Old Face", 2, 18));
// NOI18N
        jLabel1.setForeground(java.awt.Color.red);
        jLabel1.setText("( Not a member yet ! sign up )");

        Username.setFont(new  java.awt.Font("Cambria", 3, 18)); //
NOI18N
        Username.setText("Username");
```

```
        Password.setFont(new java.awt.Font("Cambria", 3, 18)); //
NOI18N
        Password.setText("Password");

        comment.setFont(new java.awt.Font("Cambria", 3, 18)); //
NOI18N
        comment.setForeground(new java.awt.Color(0, 51, 51));

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(260, 260, 260)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
                            .addComponent(Username,
javax.swing.GroupLayout.PREFERRED_SIZE, 92,
javax.swing.GroupLayout.PREFERRED_SIZE)
                            .addComponent(Password)))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(260, 260, 260)
                        .addComponent(Log_in_Button))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(230, 230, 230)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
                            .addComponent(Textfield_username,
javax.swing.GroupLayout.DEFAULT_SIZE, 151, Short.MAX_VALUE)
                            .addComponent(Passwordfield_password)))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(258, 258, 258)
                        .addComponent(Sign_up_Button))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(203, 203, 203)
                        .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 226,
javax.swing.GroupLayout.PREFERRED_SIZE))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(182, 182, 182)
                        .addComponent(comment)))
```

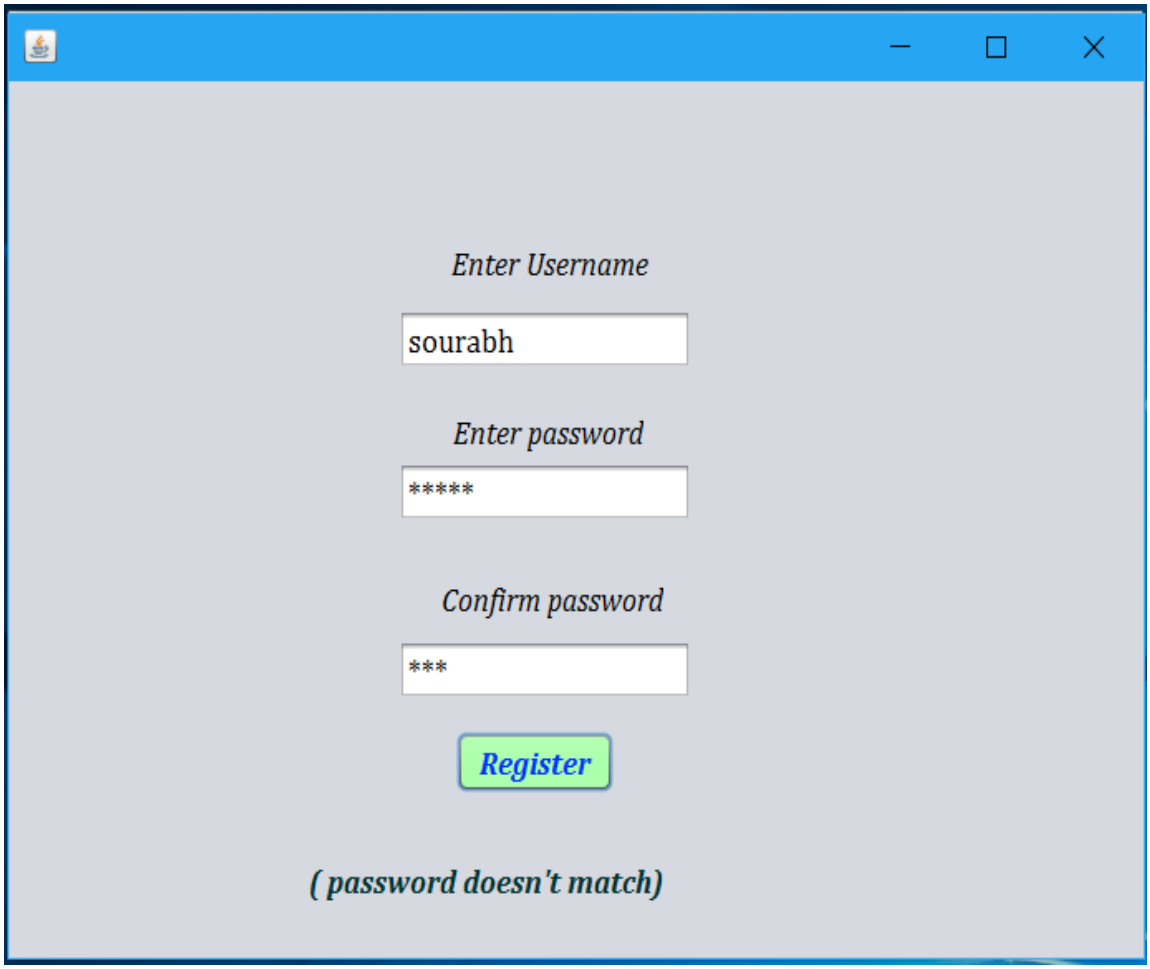
```
                .addContainerGap(255, Short.MAX_VALUE))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addGap(61, 61, 61)
                .addComponent(Username,
javax.swing.GroupLayout.PREFERRED_SIZE,  31,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.REL
ATED)
                .addComponent(Textfield_username,
javax.swing.GroupLayout.PREFERRED_SIZE,  35,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(Password,
javax.swing.GroupLayout.PREFERRED_SIZE,  21,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.REL
ATED)
                .addComponent(Passwordfield_password,
javax.swing.GroupLayout.PREFERRED_SIZE,  33,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(31, 31, 31)
                .addComponent(Log_in_Button,
javax.swing.GroupLayout.PREFERRED_SIZE,  39,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE,  29,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.REL
ATED)
                .addComponent(Sign_up_Button,
javax.swing.GroupLayout.PREFERRED_SIZE,  42,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(comment)
                .addContainerGap(91, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents
    Home_page home = new Home_page();
```

```
    private void
Log_in_ButtonMouseClicked(java.awt.event.MouseEvent evt) {//GEN-
FIRST:event_Log_in_ButtonMouseClicked
        try{
            Connection myconn =
DriverManager.getConnection("jdbc:mysql://Localhost:3306/algorithma
?autoReconnect=true&useSSL=false","root","Sourabh@67");
            Statement stmt = myconn.createStatement();
            ResultSet rs = stmt.executeQuery("select * from members");
            while(rs.next()){
              String user = rs.getString("username");
              String pass = rs.getString("password");

                if(user.equals(Textfield_username.getText()) &&
pass.equals(Passwordfield_password.getText())){
                    home.setusername(user);
                    home.setVisible(true);
                    flag = 1;
                }
            }

            if(flag == 0){
                comment.setText(" ( User doesn't exist ! ) ");

            }
            if(Textfield_username.getText().equals("") ||
Passwordfield_password.getText().equals("")){
                comment.setText(" ( username or password can't be empty )
");
            }
        }
        catch(Exception e){
            e.printStackTrace();
        }
    }//GEN-LAST:event_Log_in_ButtonMouseClicked
    Sign_up sign_up = new Sign_up();
    private void
Sign_up_ButtonMouseClicked(java.awt.event.MouseEvent evt)
{//GEN-FIRST:event_Sign_up_ButtonMouseClicked
        sign_up.setVisible(true);
    }//GEN-LAST:event_Sign_up_ButtonMouseClicked

    private void
Textfield_usernameKeyPressed(java.awt.event.KeyEvent evt) {//GEN-
FIRST:event_Textfield_usernameKeyPressed
            comment.setText("");
    }//GEN-LAST:event_Textfield_usernameKeyPressed
```

```
    private void
Textfield_usernameFocusGained(java.awt.event.FocusEvent evt)
{//GEN-FIRST:event_Textfield_usernameFocusGained
            Textfield_username.setBackground(new java.awt.Color(0, 255,
204));
    }//GEN-LAST:event_Textfield_usernameFocusGained

    private void
Textfield_usernameFocusLost(java.awt.event.FocusEvent evt) {//GEN-
FIRST:event_Textfield_usernameFocusLost
            Textfield_username.setBackground(new
java.awt.Color(255,255,255));
    }//GEN-LAST:event_Textfield_usernameFocusLost

    private void
Passwordfield_passwordFocusGained(java.awt.event.FocusEvent evt)
{//GEN-FIRST:event_Passwordfield_passwordFocusGained
            Passwordfield_password.setBackground(new
java.awt.Color(204,204,255));
    }//GEN-LAST:event_Passwordfield_passwordFocusGained

    private void
Passwordfield_passwordFocusLost(java.awt.event.FocusEvent evt)
{//GEN-FIRST:event_Passwordfield_passwordFocusLost
            Passwordfield_password.setBackground(new
java.awt.Color(255,255,255));
    }//GEN-LAST:event_Passwordfield_passwordFocusLost

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
```

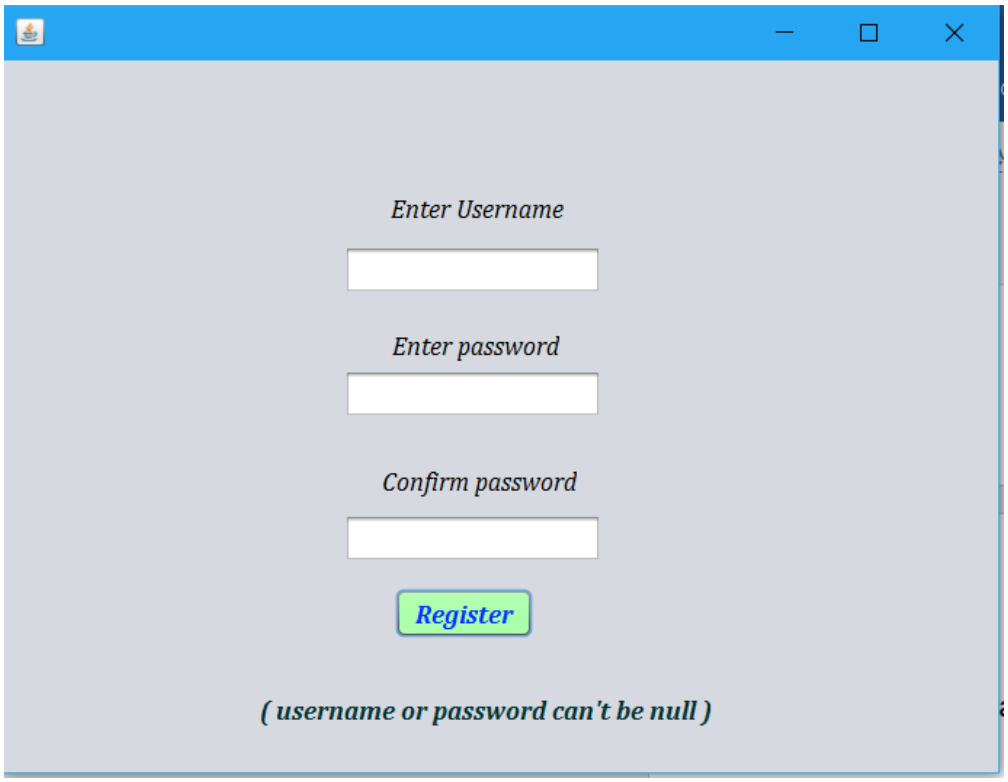
```
java.util.logging.Logger.getLogger(Log_in.class.getName()).log(java.uti
l.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Log_in.class.getName()).log(java.uti
l.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Log_in.class.getName()).log(java.uti
l.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Log_in.class.getName()).log(java.uti
l.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Log_in().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton Log_in_Button;
    private javax.swing.JLabel Password;
    private javax.swing.JPasswordField Passwordfield_password;
    private javax.swing.JButton Sign_up_Button;
    private javax.swing.JTextField Textfield_username;
    private javax.swing.JLabel Username;
    private javax.swing.JLabel comment;
    private javax.swing.JLabel jLabel1;
    // End of variables declaration//GEN-END:variables

}
```

2. **Sign_up.java:**

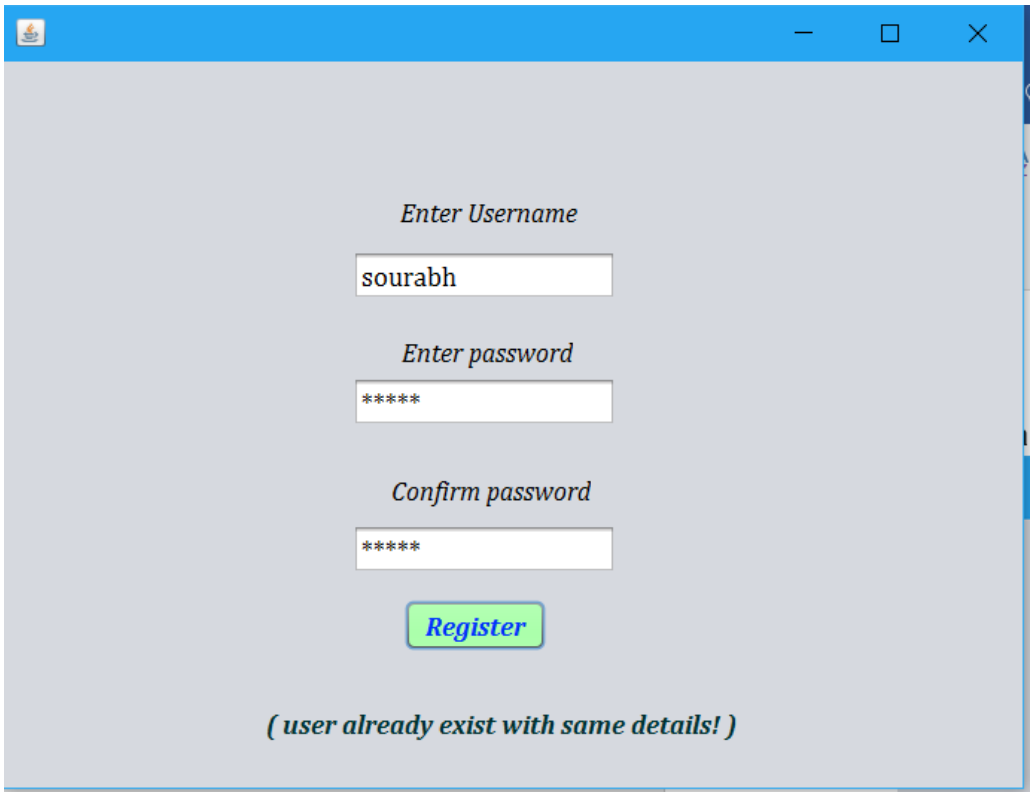


Explanation:

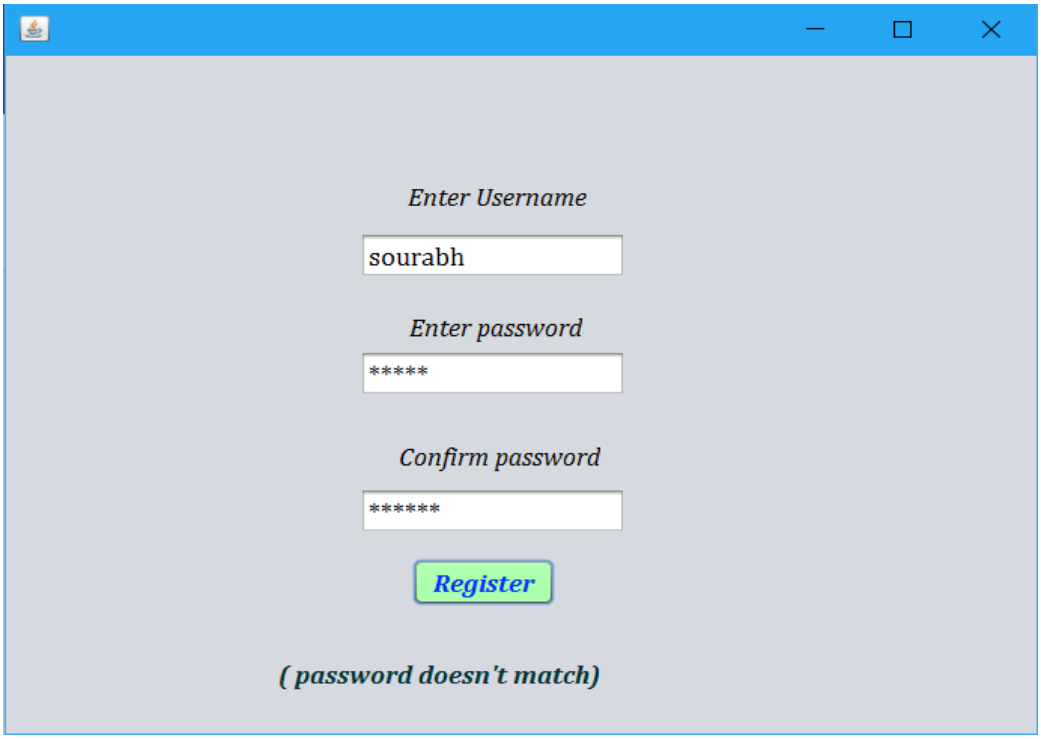
- For sign-up page, I used some colour changing effects while moving to the next text field as shown in log-in page explanation also. For this purpose, I add FocusListner with text fields and password fields. I used focusGained() and focusLost() methods and in these methods I wrote the code for changing the colour of the text field. On gaining focus textfield's background colour will automatically change to some other colour and after losing focus it will again reset to white colour. Code for white colour is written in the focusLost method and code for changing colour to some other colour is written in the focusGained method.
- Sign-up page contains some validations like:
    - If all the fields are empty then error message will be shown like:



- If user with same username is already in my database then again error message will be showed like:



- if password is not getting matched with the confirm password then again it will show error message like:



All these validations are happening because user will register in the system only by clicking register button for this I used MouseListner and mouseClicked method and in mouseClicked method I wrote all the code. I used if else conditions for checking conditions like if username and password are empty then accordingly I made cases to print the error message.

- After validating all the details user will registered in the system and user details will get add in the database. After successful registration, sign-up page will get **dispose**.

Code:

```
import java.sql.*;
public class Sign_up extends javax.swing.JFrame {

    /**
     * Creates new form Sign_up
     */
    public Sign_up() {
        initComponents();
```

```
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method
is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        jLabel1 = new javax.swing.JLabel();
        Textfield_username  = new javax.swing.JTextField();
        jLabel2 = new javax.swing.JLabel();
        passwordfield_password = new javax.swing.JPasswordField();
        jLabel3 = new javax.swing.JLabel();
        Passwordfield_confirm_password = new
javax.swing.JPasswordField();
        Register_Button = new javax.swing.JButton();
        comment = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_O
N_CLOSE);

        jLabel1.setFont(new java.awt.Font("Cambria", 2, 18)); // NOI18N
        jLabel1.setText("Enter Username");

        Textfield_username.setFont(new java.awt.Font("Cambria", 0, 18));
// NOI18N
        Textfield_username.addFocusListener(new
java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                Textfield_usernameFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent evt) {
                Textfield_usernameFocusLost(evt);
            }
        });
        Textfield_username.addKeyListener(new
java.awt.event.KeyAdapter() {
            public void keyPressed(java.awt.event.KeyEvent evt) {
                Textfield_usernameKeyPressed(evt);
            }
        });
```

```
        jLabel2.setFont(new java.awt.Font("Cambria", 2, 18)); // NOI18N
        jLabel2.setText("Enter password");

        passwordfield_password.setFont(new java.awt.Font("Cambria", 0,
18)); // NOI18N
        passwordfield_password.addFocusListener(new
java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                passwordfield_passwordFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent evt) {
                passwordfield_passwordFocusLost(evt);
            }
        });
        passwordfield_password.addKeyListener(new
java.awt.event.KeyAdapter() {
            public void keyPressed(java.awt.event.KeyEvent evt) {
                passwordfield_passwordKeyPressed(evt);
            }
        });

        jLabel3.setFont(new java.awt.Font("Cambria", 2, 18)); // NOI18N
        jLabel3.setText("Confirm password");

        Passwordfield_confirm_password.setFont(new
java.awt.Font("Cambria", 0, 18)); // NOI18N
        Passwordfield_confirm_password.addFocusListener(new
java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                Passwordfield_confirm_passwordFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent evt) {
                Passwordfield_confirm_passwordFocusLost(evt);
            }
        });
        Passwordfield_confirm_password.addKeyListener(new
java.awt.event.KeyAdapter() {
            public void keyPressed(java.awt.event.KeyEvent evt) {
                Passwordfield_confirm_passwordKeyPressed(evt);
            }
        });

        Register_Button.setBackground(new java.awt.Color(153, 255,
153));
        Register_Button.setFont(new java.awt.Font("Cambria", 3, 18)); //
NOI18N
        Register_Button.setForeground(new java.awt.Color(0, 51, 255));
```

```
        Register_Button.setText("Register");
        Register_Button.addMouseListener(new
java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent  evt) {
                Register_ButtonMouseClicked(evt);
            }
        });

        comment.setFont(new  java.awt.Font("Cambria", 3, 18)); //
NOI18N
        comment.setForeground(new  java.awt.Color(0, 51, 51));

        javax.swing.GroupLayout  layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(235, 235, 235)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING,  false)
                            .addComponent(Textfield_username,
javax.swing.GroupLayout.DEFAULT_SIZE,  177, Short.MAX_VALUE)
                            .addComponent(passwordfield_password)
                            .addComponent(Passwordfield_confirm_password)))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(267, 267, 267)
                        .addComponent(jLabel1))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(261, 261, 261)
                        .addComponent(jLabel3))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(268, 268, 268)
                        .addComponent(jLabel2))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(270, 270, 270)
                        .addComponent(Register_Button))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(177, 177, 177)
                        .addComponent(comment)))
                .addContainerGap(273, Short.MAX_VALUE))
```

```
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addGap(82, 82, 82)
                .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE,  39,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.REL
ATED)
                .addComponent(Textfield_username,
javax.swing.GroupLayout.PREFERRED_SIZE,  33,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(24, 24, 24)
                .addComponent(jLabel2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.REL
ATED)
                .addComponent(passwordfield_password,
javax.swing.GroupLayout.PREFERRED_SIZE,  33,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(32, 32, 32)
                .addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UN
RELATED)
                .addComponent(Passwordfield_confirm_password,
javax.swing.GroupLayout.PREFERRED_SIZE,  33,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(Register_Button)
                .addGap(33, 33, 33)
                .addComponent(comment,
javax.swing.GroupLayout.PREFERRED_SIZE,  33,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(26, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void
Register_ButtonMouseClicked(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_Register_ButtonMouseClicked
        try{
```

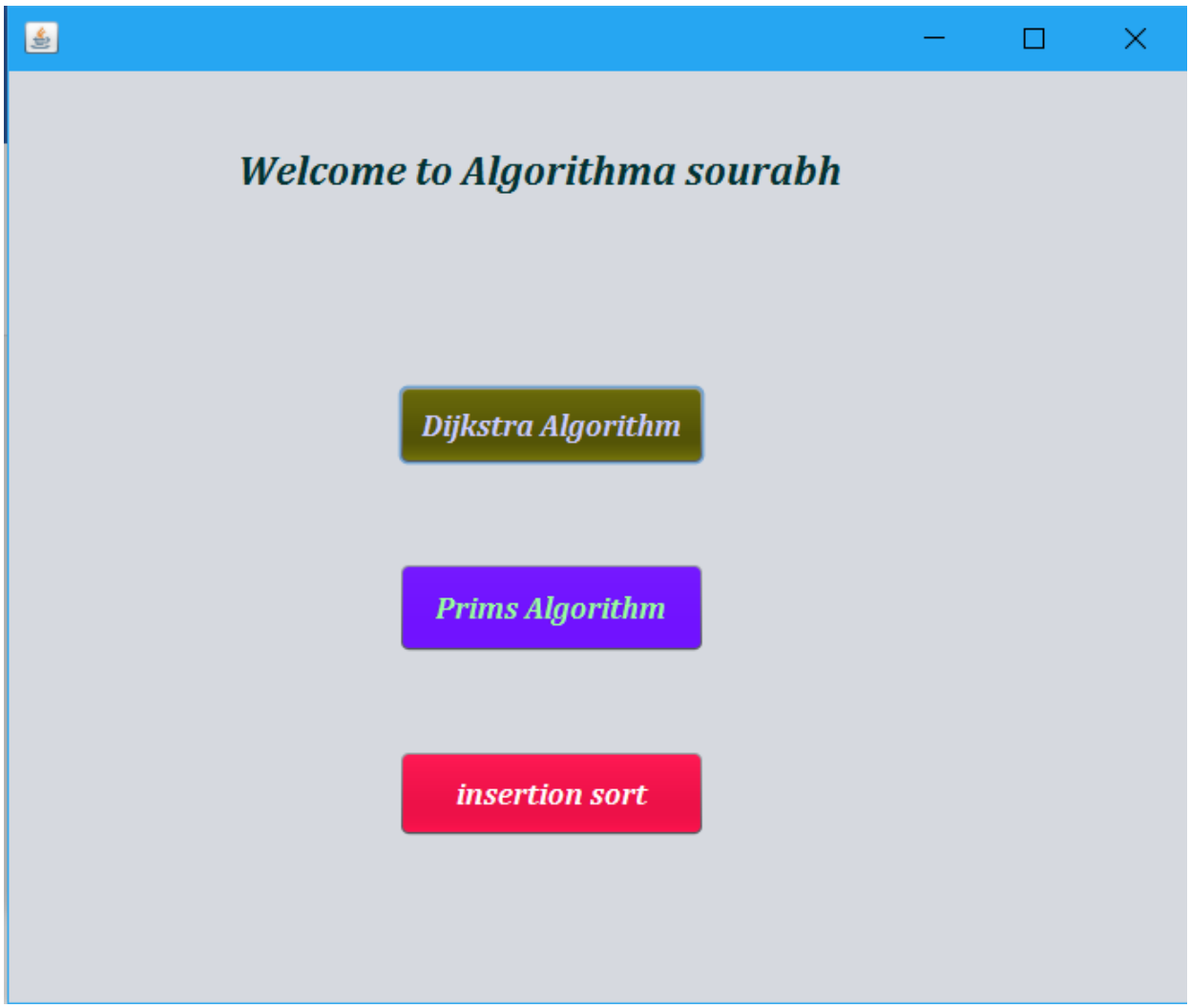
```
        int flag = 0;
        Connection myconn =
DriverManager.getConnection("jdbc:mysql://Localhost:3306/algorithma
?autoReconnect=true&useSSL=false","root","Sourabh@67");
        Statement stmt = myconn.createStatement();
        ResultSet rs = stmt.executeQuery("select * from members");
        String u= Textfield_username.getText();
        String pass = passwordfield_password.getText();
        String confirm_password =
Passwordfield_confirm_password.getText();
        if(u.equals("") || pass.equals("")){
           comment.setText("( username or password can't be null )");
        }
        else{
           String sql = "INSERT INTO members(username,password)
VALUES('"+u+"','"+pass+"')";
            if(pass.equals(confirm_password)){
               while(rs.next()){
                  String user = rs.getString("username");
                  String password = rs.getString("password");
                  if(u.equals(user)){
                     flag = 1;
                  }
               }
               if(flag == 0){
                  stmt.executeUpdate(sql);

                  this.dispose();
               }
               else if(flag == 1){
               comment.setText("( user already exist with same details!
)");
               }
            }
            else if(!pass.equals(confirm_password)){
               comment.setText(" ( password doesn't match) ");
            }
        }

      }
      catch(Exception e){
         e.printStackTrace();
      }
   }//GEN-LAST:event_Register_ButtonMouseClicked

   private void
Textfield_usernameKeyPressed(java.awt.event.KeyEvent evt) {//GEN-
FIRST:event_Textfield_usernameKeyPressed
```

```
        comment.setText("");
    }//GEN-LAST:event_Textfield_usernameKeyPressed

    private void
Passwordfield_confirm_passwordKeyPressed(java.awt.event.KeyEven
t evt) {//GEN-
FIRST:event_Passwordfield_confirm_passwordKeyPressed
        comment.setText("");
    }//GEN-LAST:event_Passwordfield_confirm_passwordKeyPressed

    private void
passwordfield_passwordKeyPressed(java.awt.event.KeyEvent evt)
{//GEN-FIRST:event_passwordfield_passwordKeyPressed
        comment.setText("");
    }//GEN-LAST:event_passwordfield_passwordKeyPressed

    private void
Textfield_usernameFocusGained(java.awt.event.FocusEvent evt)
{//GEN-FIRST:event_Textfield_usernameFocusGained
        Textfield_username.setBackground(new
java.awt.Color(204,204,255));
    }//GEN-LAST:event_Textfield_usernameFocusGained

    private void
Textfield_usernameFocusLost(java.awt.event.FocusEvent evt) {//GEN-
FIRST:event_Textfield_usernameFocusLost
        Textfield_username.setBackground(new
java.awt.Color(255,255,255));
    }//GEN-LAST:event_Textfield_usernameFocusLost

    private void
passwordfield_passwordFocusGained(java.awt.event.FocusEvent evt)
{//GEN-FIRST:event_passwordfield_passwordFocusGained
        passwordfield_password.setBackground(new java.awt.Color(0,
255, 204));
    }//GEN-LAST:event_passwordfield_passwordFocusGained

    private void
passwordfield_passwordFocusLost(java.awt.event.FocusEvent evt)
{//GEN-FIRST:event_passwordfield_passwordFocusLost
        passwordfield_password.setBackground(new
java.awt.Color(255,255,255));
    }//GEN-LAST:event_passwordfield_passwordFocusLost

    private void
Passwordfield_confirm_passwordFocusGained(java.awt.event.FocusE
vent evt) {//GEN-
FIRST:event_Passwordfield_confirm_passwordFocusGained
```

```
        Passwordfield_confirm_password.setBackground(new
java.awt.Color(0, 255, 204));
    }//GEN-LAST:event_Passwordfield_confirm_passwordFocusGained

    private void
Passwordfield_confirm_passwordFocusLost(java.awt.event.FocusEven
t evt) {//GEN-
FIRST:event_Passwordfield_confirm_passwordFocusLost
        Passwordfield_confirm_password.setBackground(new
java.awt.Color(255,255,255));
    }//GEN-LAST:event_Passwordfield_confirm_passwordFocusLost

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Sign_up.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Sign_up.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Sign_up.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(Sign_up.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Sign_up().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JPasswordField
Passwordfield_confirm_password;
    private javax.swing.JButton Register_Button;
    private javax.swing.JTextField Textfield_username;
    private javax.swing.JLabel comment;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JPasswordField passwordfield_password;
    // End of variables declaration//GEN-END:variables
}
```

3. **Home_page.java:**



Explanation:

Home page has buttons like: “Dijkstra Algorithm”, “Prims Algorithm”, “insertion sort”. On buttons, I add mouseListner and used mousedClicked method so that after clicking on the button I can go to the next frame. Code for moving to next frame is written in the mouseClicked method. In mouseClicked method I simply used “setVisible(true)” to show the next frame.

I used “Exit-on-Close” as defaultCloseOperation on insertion sort frame so that on the exit of insertion sort frame complete program will get closed.

In the beginning of the home page welcome message is getting shown which show the user name. I got the username by the previous log-in page by making getter and setter methods in home-page and called these methods by log-in after successful log-in.

Code:

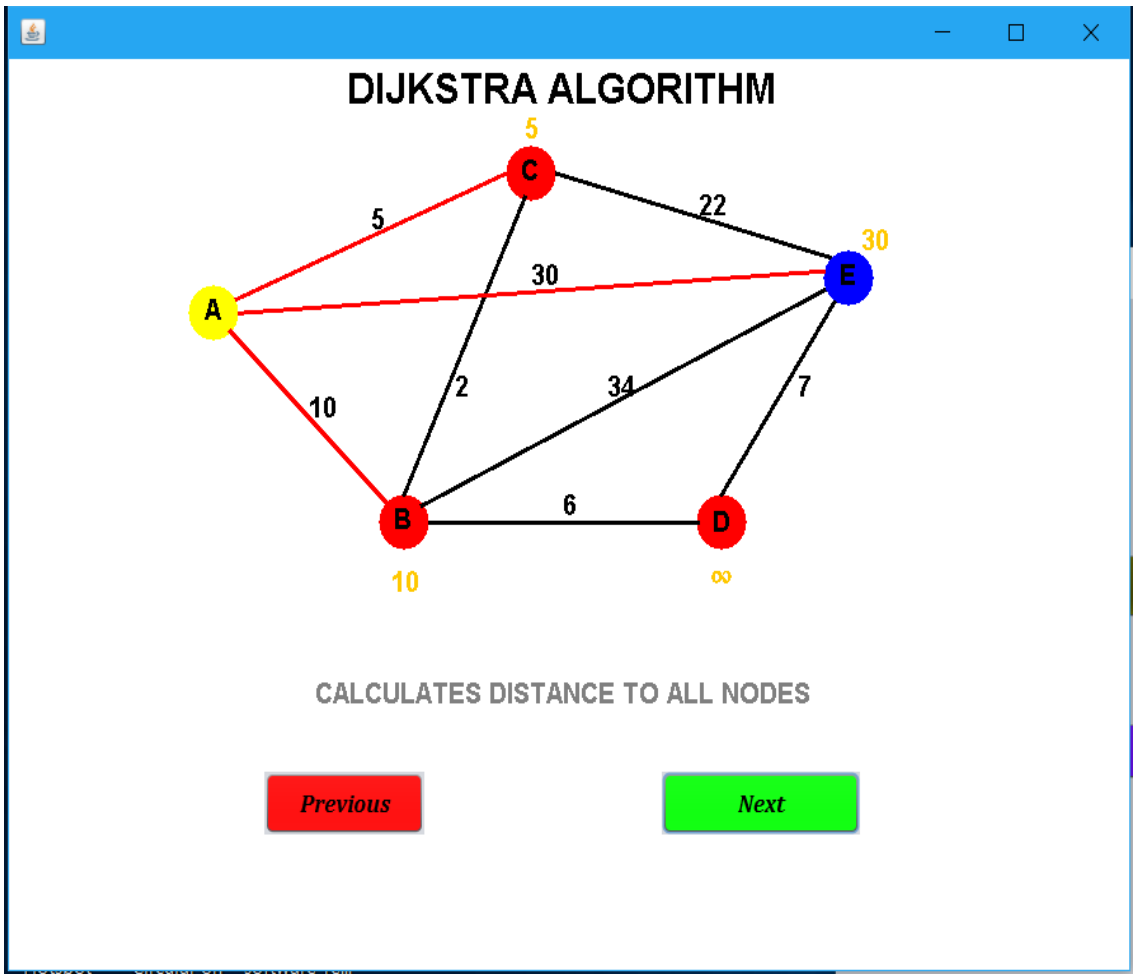
```
/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author sourabh
 */
public class Home_page extends javax.swing.JFrame {
 String username = "";
    public void setusername(String user){
    username = user;
    comment.setText("Welcome to Algorithma " + username);
    }
    /**
     * Creates new form Home_page
     */
    public Home_page() {
        initComponents();

    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method
is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        comment = new javax.swing.JLabel();
        da = new javax.swing.JButton();
        pri = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_O
N_CLOSE);

        comment.setFont(new java.awt.Font("Cambria", 3, 24)); //
NOI18N
        comment.setForeground(new java.awt.Color(0, 51, 51));
```

```
        da.setBackground(new java.awt.Color(51, 51, 0));
        da.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
        da.setForeground(new java.awt.Color(204, 204, 255));
        da.setText("Dijkstra Algorithm");
        da.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                daMouseClicked(evt);
            }
        });

        pri.setBackground(new java.awt.Color(102, 0, 255));
        pri.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
        pri.setForeground(new java.awt.Color(153, 255, 153));
        pri.setText("Prims Algorithm");
        pri.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                priMouseClicked(evt);
            }
        });

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(133, 133, 133)
                        .addComponent(comment))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(226, 226, 226)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
                            .addComponent(da,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                            .addComponent(pri,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))
                .addContainerGap(280, Short.MAX_VALUE))
        );
        layout.setVerticalGroup(
```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addGap(43, 43, 43)
                .addComponent(comment)
                .addGap(110, 110, 110)
                .addComponent(da,
javax.swing.GroupLayout.PREFERRED_SIZE,  47,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(74, 74, 74)
                .addComponent(pri,
javax.swing.GroupLayout.PREFERRED_SIZE,  42,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(225, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents
    Dijkstar1 dij = new Dijkstar1();
    private void daMouseClicked(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_daMouseClicked
        dij.setVisible(true);
    }//GEN-LAST:event_daMouseClicked
    prims pr = new prims();
    private void priMouseClicked(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_priMouseClicked
        pr.setVisible(true);
    }//GEN-LAST:event_priMouseClicked

    /**
     * @param args the command  line arguments
     */
    public static void main(String  args[]) {
        /* Set the Nimbus  look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus  (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo  info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
```

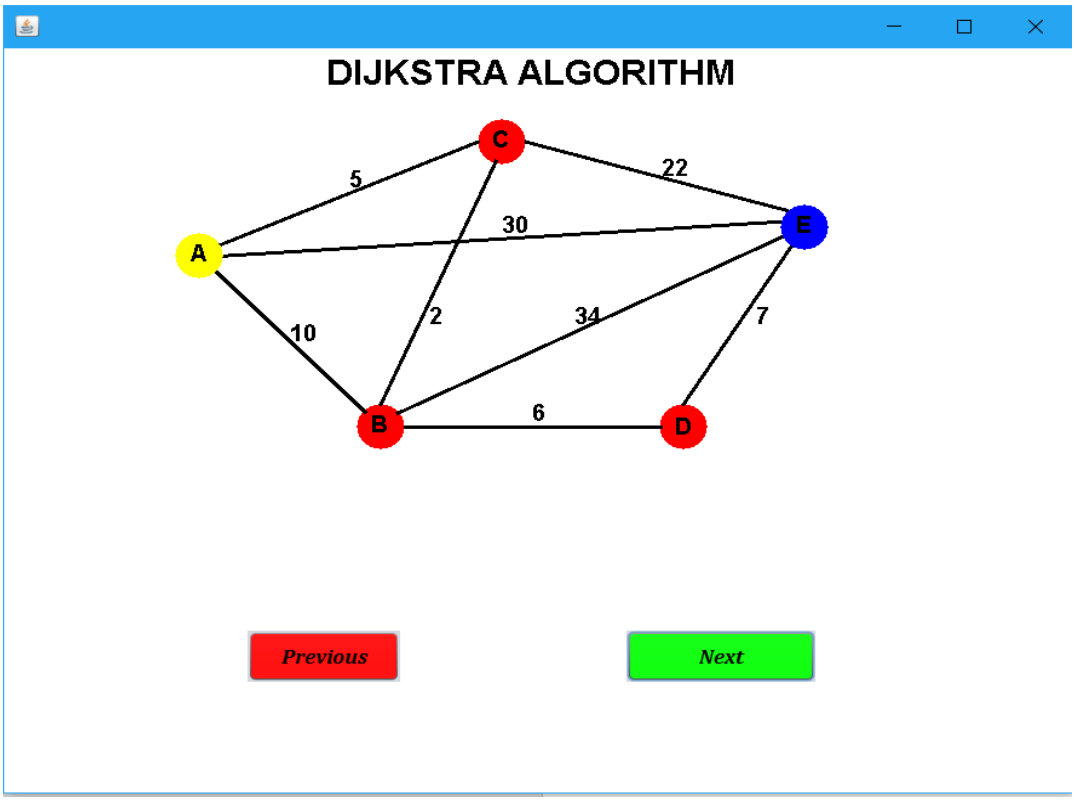
```
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Home_page.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Home_page.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Home_page.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Home_page.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Home_page().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JLabel comment;
    private javax.swing.JButton da;
    private javax.swing.JButton pri;
    // End of variables declaration//GEN-END:variables

}
```

4. **Dijkstar1.java:**

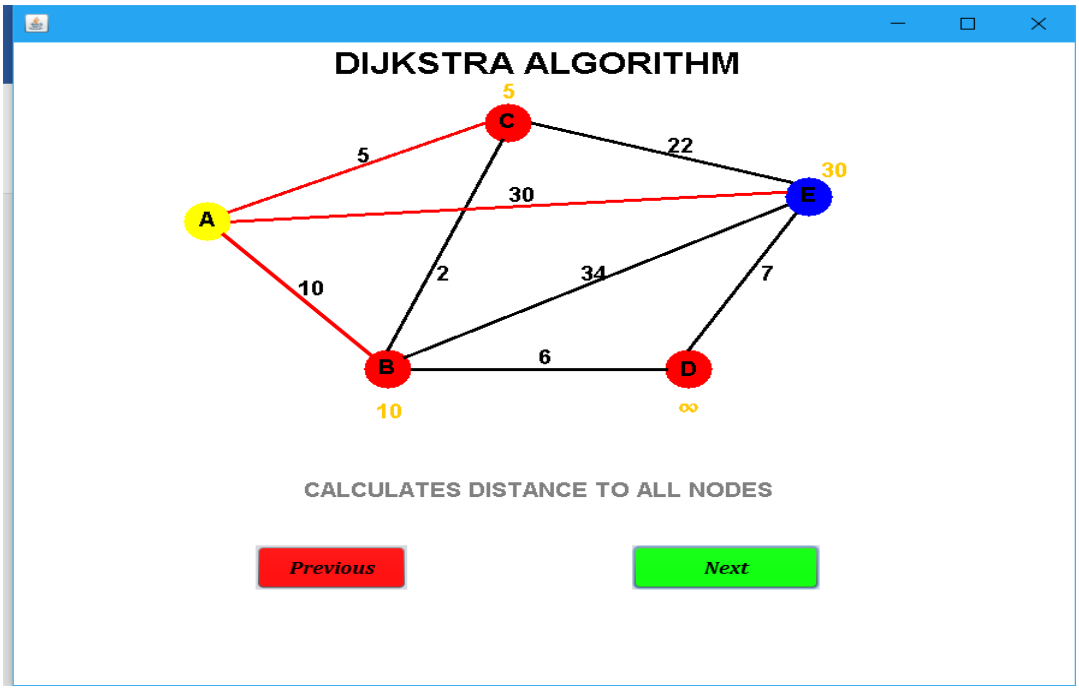


Explanation:

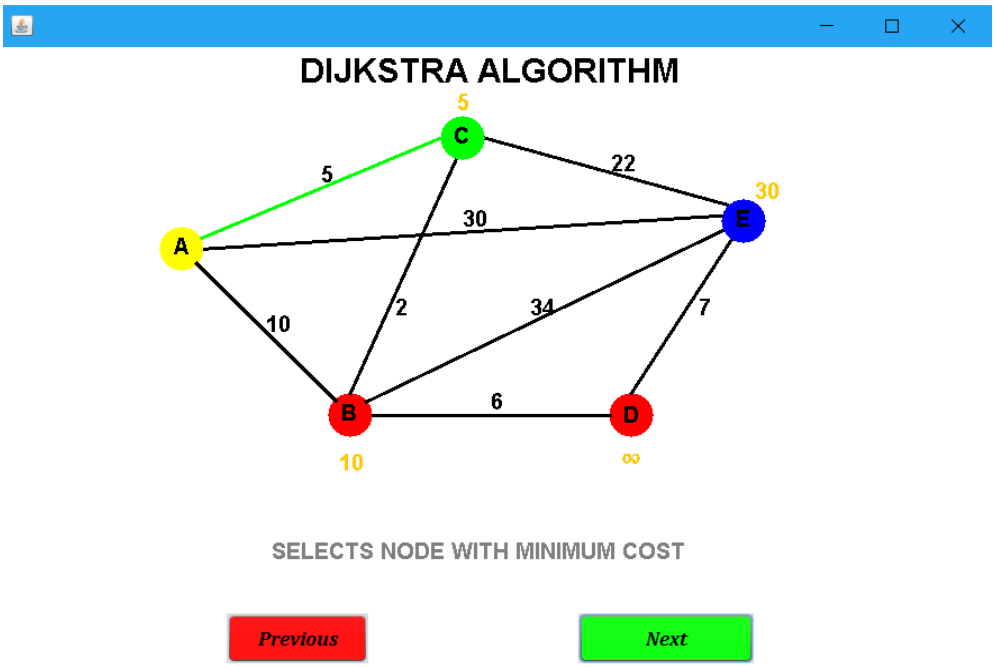
We implemented Dijkstra's Algorithm by using 2 buttons called "Next" and "Previous". Initially we made one variable 'n' and define its value as '-1', On both of the buttons I add MouseListner and used MouseClicked method so after each click on next button some code will get implemented and after clicking on previous button other code is getting implemented. On nextbuttonMouseClicked method I am increasing the value of n by 1 and after it calling repaint() method. Same in previousButtonMouseClicked method I am decreasing the value of n by 1. So algorithm is moving further by checking the value of n and simple using paint() method accordingly. We also showing comments related to events occurring at that time of the algorithm. At last we are showing graph of shorted path and distance.



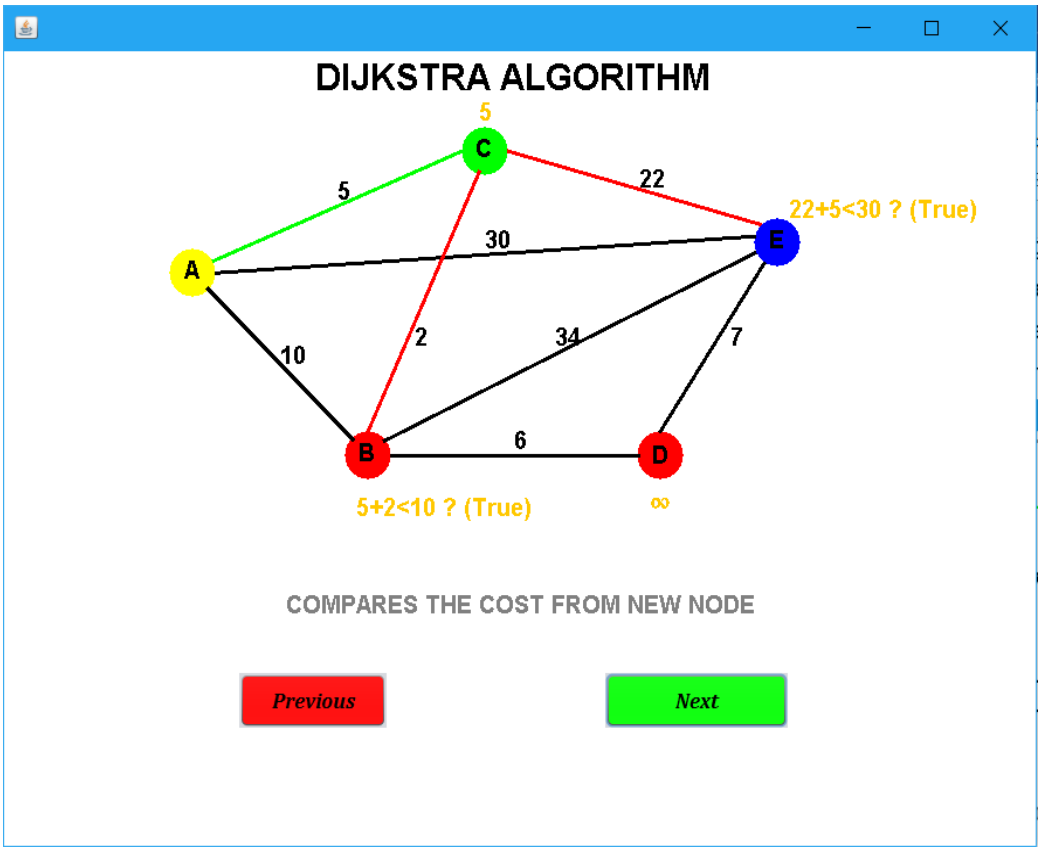
Like this the second step of the algorithm in which I am selecting root node as A and making it yellow colour.



This is the third step of the algorithm where I am selecting the nodes which are connected with root node A and showing the weights on the top of the nodes. All selecting nodes are connecting with red lines. The node which is going to be selected will connected with green line. Like

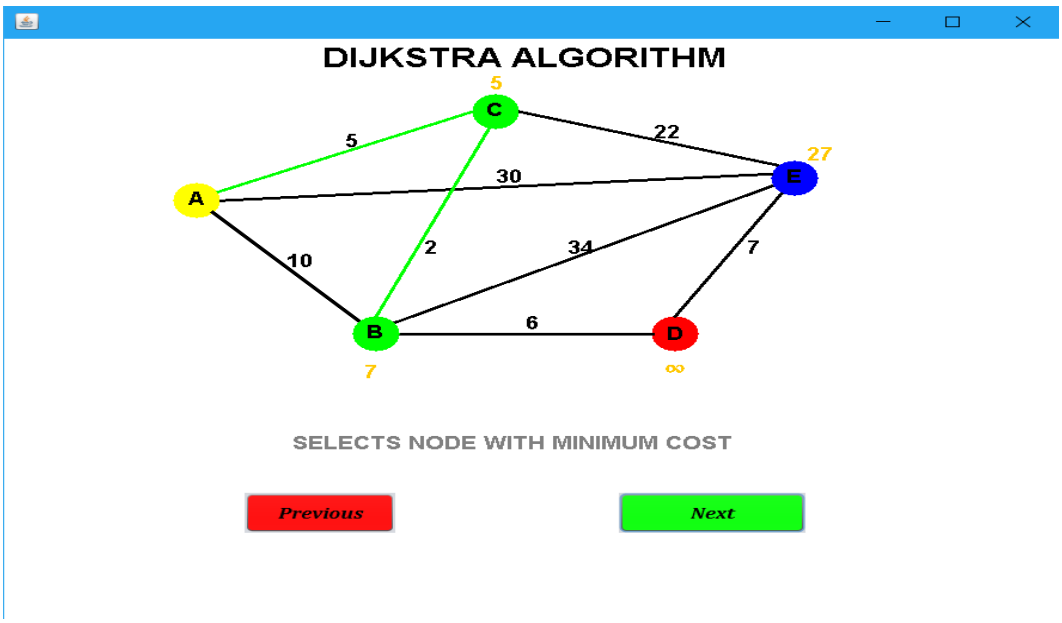


After this step 2 nodes are already selected ‘A’ and ‘C’. now algorithm is going to check for shortest node which is connected with C.

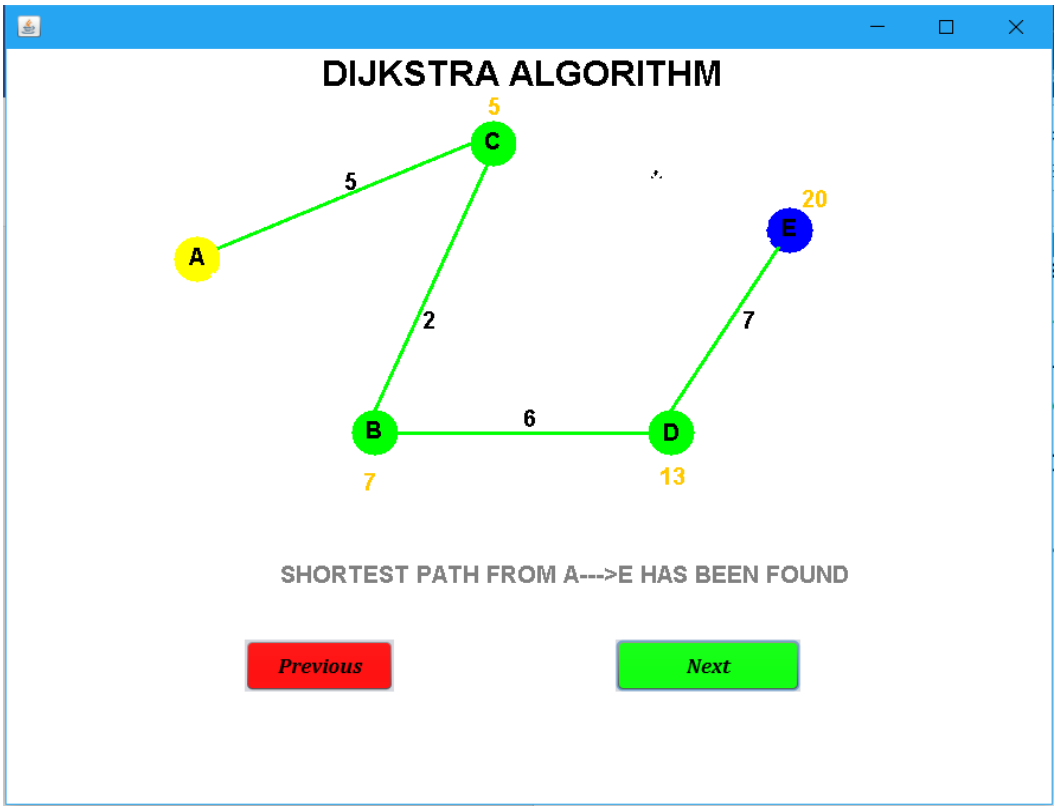


In this step 2 red lines are visible which means algorithm selected node B and node E because they are connected with node C. now algorithm is comparing

the weight of the nodes with the previous weight. If the new weight is lesser than the previous then the new weight will get updated otherwise it will remain same it was before. After updating weights algorithm will calculate the next shortest path which is available and connect it with green line like:



After this step nodes ‘A’, ‘C’, ’B’ are selected so algorithm will move further in this way and finally it will show the shortest path between root node ‘A’ and the destination node ‘E’ like:



Code:

```
import java.awt.BasicStroke;
import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;


public class Dijkstar1 extends javax.swing.JFrame {

   int n=-1;
   int m=0;
   public Dijkstar1() {
      initComponents();
      setSize(900,700);
   }

   /**
    * This method is called from within the constructor to initialize the
form.
    * WARNING: Do NOT modify this code. The content of this method
is always
    * regenerated by the Form Editor.
    */
   @SuppressWarnings("unchecked")
   // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN:initComponents
   private void initComponents() {

      previous = new javax.swing.JButton();
      next = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.DISPOSE_O
N_CLOSE);

      previous.setBackground(java.awt.Color.red);
      previous.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
      previous.setText("Previous");
      previous.addMouseListener(new java.awt.event.MouseAdapter() {
         public void mousePressed(java.awt.event.MouseEvent evt) {
            previousMousePressed(evt);
         }
      });

      next.setBackground(java.awt.Color.green);
      next.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
      next.setText("Next");
```

```
        next.addMouseListener(new  java.awt.event.MouseAdapter() {
            public void mousePressed(java.awt.event.MouseEvent  evt) {
                nextMousePressed(evt);
            }
        });

        javax.swing.GroupLayout  layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addGap(201, 201, 201)
                .addComponent(previous,
javax.swing.GroupLayout.PREFERRED_SIZE,  126,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(187, 187, 187)
                .addComponent(next,
javax.swing.GroupLayout.PREFERRED_SIZE,  156,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(285, Short.MAX_VALUE))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap(332, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
                    .addComponent(previous,
javax.swing.GroupLayout.PREFERRED_SIZE,  45,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(next,
javax.swing.GroupLayout.PREFERRED_SIZE,  45,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(97, 97, 97))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void nextMousePressed(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_nextMousePressed
n++;
repaint();
    }//GEN-LAST:event_nextMousePressed
```

```
    private void previousMousePressed(java.awt.event.MouseEvent evt)
{//GEN-FIRST:event_previousMousePressed
n--;
m=1;
repaint();
    }//GEN-LAST:event_previousMousePressed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Dijkstar1.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Dijkstar1.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Dijkstar1.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Dijkstar1.class.getName()).log(java.
util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
```

```
            public void run() {
                new Dijkstar1().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton next;
    private javax.swing.JButton previous;
    // End of variables declaration//GEN-END:variables
public void paint(Graphics g)
    {

        if(m==1)
        {
            g.setColor(Color.white);
            g.fillRect(0, 70, 1000, 1000);
            m=0;
        }

        if(n>=0){
            g.setColor(Color.white);
            g.fillRect(0, 75, 1000, 450);
            Font f=new Font("TimesNewRoman",Font.BOLD,30);
        g.setFont(f);
        g.setColor(Color.black);
          g.drawString("DIJKSTRA ALGORITHM", 275,70);
        f=new Font("TimesNewRoman",Font.BOLD,20);
        g.setFont(f);
         Graphics2D g2 = (Graphics2D) g;
         g2.setStroke(new BasicStroke(3));
        g.setColor(Color.red);
       g.fillOval(150,200,40,40);//A
       g.fillOval(300,350,40,40);//B
       g.fillOval(400,100,40,40);//c
       g.fillOval(550,350,40,40);//D
       g.fillOval(650,175,40,40);//E
        g.setColor(Color.black);




        g.drawString("A", 163,225);
        g.drawString("B", 312,375);
        g.drawString("C", 412,125);
        g.drawString("D", 563,377);
        g.drawString("E", 663,200);
```

```
        g.drawLine(183,233,307,357);//AB
        g.drawLine(188,210,400,120);//AC
        g.drawLine(190,220,650,190);//AE
        g.drawLine(320,350,415,137);//BC
        g.drawLine(340,370,551,370);//BD
        g.drawLine(570,350,660,210);//DE
        g.drawLine(334,358,652,203);//BE
        g.drawLine(440,120,655,180);//CE

        g.drawString("10",245,295);//AB weight
        g.drawString("5", 294,160);//AC weight
        g.drawString("30", 420,200);//AE weight
        g.drawString("2", 360,280);//BC weight
        g.drawString("6", 445,365);//BD weight
        g.drawString("7", 630,280);//DE weight
        g.drawString("34", 480,280);//BE weight
        g.drawString("22", 552,150);//CE weight
        g.setColor(Color.yellow);
                g.fillOval(150,200,40,40);//A
                g.setColor(Color.black);
                g.drawString("A", 163,225);
                g.setColor(Color.green);

        g.setColor(Color.blue);
                g.fillOval(650,175,40,40);//E
                g.setColor(Color.black);
                g.drawString("E", 663,200);


         if (n>=1)
        {
            g.setColor(Color.orange);
            g.drawString("5",415, 95);//over C
            g.drawString("10",310,420);//below B
            g.drawString("∞",560, 415);//below D
            g.drawString("30",680, 175);//below E
            g.setColor(Color.gray);
            g.drawString("CALCULATES DISTANCE TO ALL NODES
",250,500);
            g.setColor(Color.red);
            g.drawLine(188, 210, 400, 120);
            g.drawLine(183, 233, 307, 357);
            g.drawLine(190, 220, 650, 190);

        if (n>=2)
        {
                g.setColor(Color.green);
               g.drawLine(188,210,400,120);//AC
                g.setColor(Color.green);
              g.fillOval(400,100,40,40);//C
```

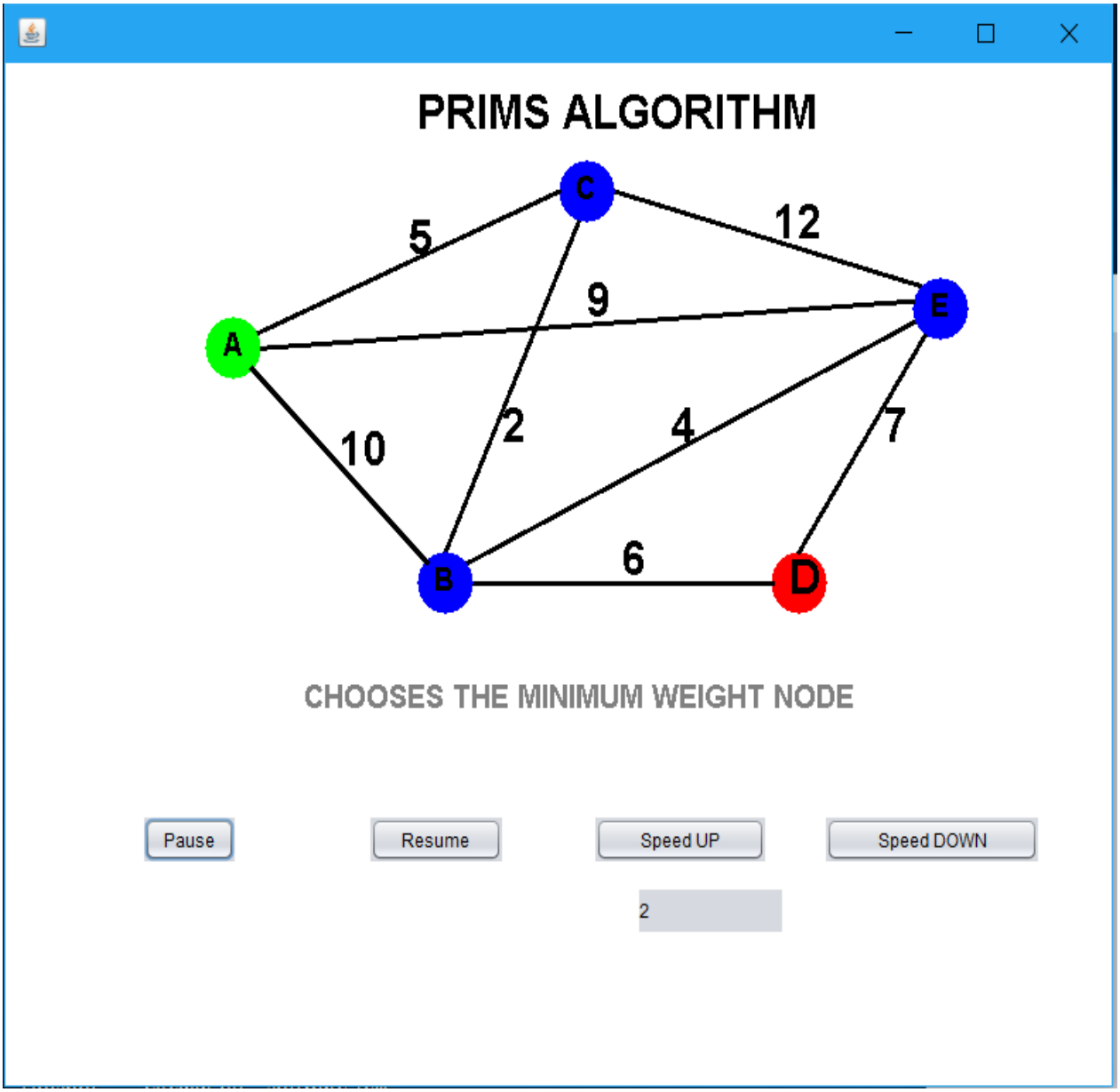
```
            g.setColor(Color.black);
           g.drawString("C", 412,125);
           g.setColor(Color.white);
            g.drawString("CALCULATES DISTANCE TO ALL NODES
",250,500);
            g.setColor(Color.gray);
         g.drawString("SELECTS NODE WITH MINIMUM COST
",250,500);
            g.setColor(Color.black);
            g.drawLine(183, 233, 307, 357);
         g.drawLine(190, 220, 650, 190);
      }

       if (n>=3)
      {
         g.setColor(Color.white);
         g.drawString("10",310,420);//below B
         g.drawString("30",680, 175);//below E
             g.setColor(Color.orange);
          g.drawString("5+2<10 ? (True)",310,420);//below B
         g.drawString("22+5<30 ? (True)",680, 175);//below E
         g.setColor(Color.white);
         g.drawString("SELECTS NODE WITH MINIMUM COST
",250,500);
         g.setColor(Color.gray);
         g.drawString("COMPARES THE COST FROM NEW
NODE",250,500);
         g.setColor(Color.red);
         g.drawLine(320,350,415,137);
         g.drawLine(440, 120, 655, 180);
       if (n>=4)
      {
             g.setColor(Color.white);
          g.drawString("5+2<10 ? (True)",310,420);//below B
         g.drawString("22+5<30 ? (True)",680, 175);//below E
         g.setColor(Color.orange);
          g.drawString("7",310,420);//below B
         g.drawString("27",680, 175);//below E
         g.setColor(Color.white);
         g.drawString("COMPARES THE COST FROM NEW
NODE",250,500);
          g.setColor(Color.gray);
         g.drawString("UPDATES THE NEW COSTS FOR
NODES",250,500);
         g.setColor(Color.red);
         g.drawLine(320,350,415,137);
         g.drawLine(440, 120, 655, 180);
       if (n>=5)
            {
            g.setColor(Color.white);
```

```
            g.drawString("UPDATES THE NEW COSTS FOR
NODES",250,500);
            g.setColor(Color.gray);
            g.drawString("SELECTS NODE WITH MINIMUM COST
",250,500);
           g.setColor(Color.green);
           g.drawLine(320,350,415,137);//BC
            g.fillOval(300,350,40,40);//B
             g.setColor(Color.black);
            g.drawString("B", 312,375);
            g.setColor(Color.black);
            g.drawLine(440, 120, 655, 180);

        if (n>=6)
            {g.setColor(Color.white);
               g.drawString("SELECTS NODE WITH MINIMUM COST
",250,500);
          g.setColor(Color.gray);
          g.drawString("COMPARES THE COST FROM NEW
NODE",250,500);
              g.setColor(Color.white);
          g.drawString("∞",560, 415);//below D
          g.drawString("27",680, 175);//below E
           g.setColor(Color.orange);
           g.drawString("5+2+6<∞ ? (True)",560, 415);//below D
          g.drawString("5+2+34<27 ? (False)",680, 175);//below E
          g.setColor(Color.red);
          g.drawLine(340, 370, 551, 370);
          g.drawLine(334, 358, 652, 203);

        if (n>=7)
            {
            g.setColor(Color.white);
             g.drawString("5+2+6<∞ ? (True)",560, 415);//below D
             g.drawString("5+2+34<27 ? (False)",680, 175);//below E
            g.setColor(Color.orange);
             g.drawString("13",560, 415);//below D
             g.drawString("27",680, 175);//below E
             g.setColor(Color.white);
             g.drawString("COMPARES THE COST FROM NEW
NODE",250,500);
           g.setColor(Color.gray);
          g.drawString("UPDATES THE NEW COSTS FOR
NODES",250,500);

        if (n>=8)
            {
             g.setColor(Color.green);
             g.fillOval(550,350,40,40);//D
             g.drawLine(340,370,551,370);//BD
```

```
                g.setColor(Color.black);
                g.drawString("D", 563,377);
                g.setColor(Color.white);
               g.drawString("UPDATES THE NEW COSTS FOR
NODES",250,500);
               g.setColor(Color.gray);
               g.drawString("SELECTS NODE WITH MINIMUM COST
",250,500);
                g.setColor(Color.black);
                g.drawLine(334,358,652,203);

       if (n>=9)
              {g.setColor(Color.white);
                g.drawString("27",680, 175);//below E
                g.setColor(Color.orange);
                g.drawString("5+2+6+7<27 ? (True)",680, 175);//below E
                g.setColor(Color.white);
                g.drawString("SELECTS NODE WITH MINIMUM COST
",250,500);
                  g.setColor(Color.gray);
                  g.drawString("COMPARES THE COST FROM NEW
NODE",250,500);
                  g.setColor(Color.red);
                  g.drawLine(570,350,660,210);


        if (n>=10)
       {
               g.setColor(Color.white);
                g.drawString("5+2+6+7<27 ? (True)",680, 175);//below E
              g.setColor(Color.orange);
                g.drawString("20",680, 175);//below E
                g.setColor(Color.green);
              g.drawLine(570,350,660,210);//DE
              g.setColor(Color.white);
                g.drawString("COMPARES THE COST FROM NEW NODE
",250,500);
                  g.setColor(Color.gray);
                  g.drawString("SELECTS NODE WITH MINIMUM
COST",250,500);


        if (n>=11)
       {g.setColor(Color.white);
                  g.drawString("SELECTS NODE WITH MINIMUM
COST",250,500);
                  g.setColor(Color.gray);
                  g.drawString("SHORTEST PATH FROM A--->E HAS
BEEN FOUND",240,500);
           g.setColor(Color.white);
```

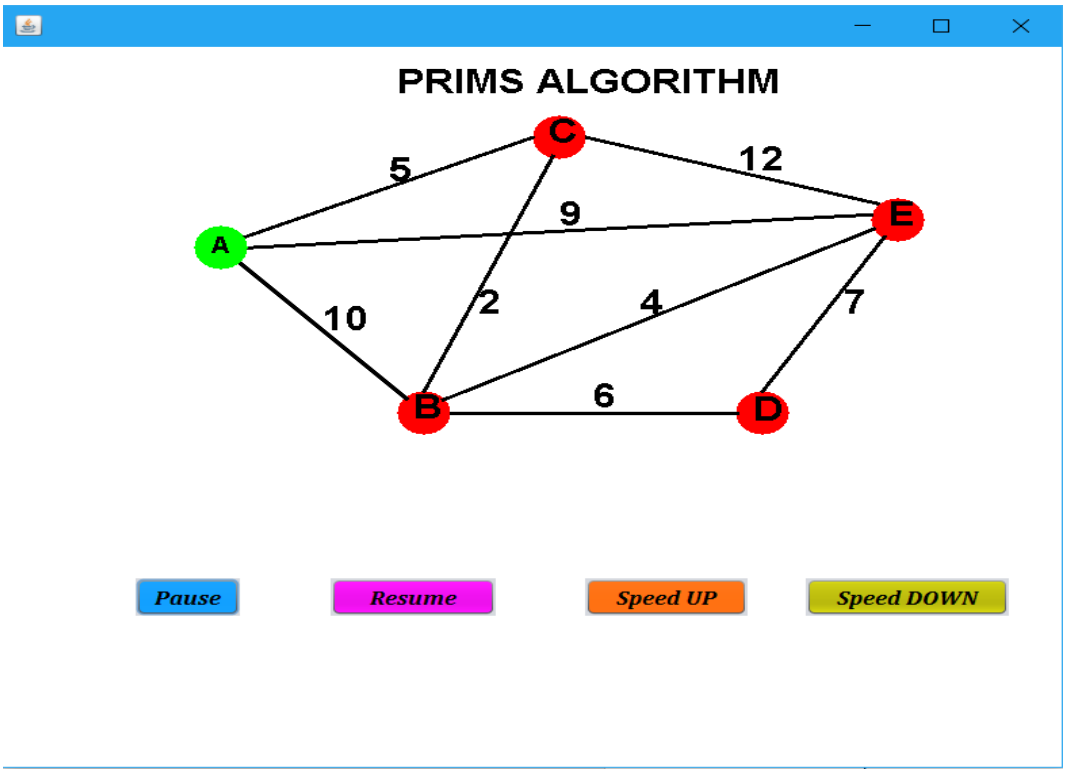
```
            g.drawLine(190,220,650,190);//AE
            g.drawLine(183,233,307,357);//AB
            g.drawLine(334,358,652,203);//BE
            g.drawLine(440,120,655,180);//CE
            g.drawString("30", 420,200);//AE weight
             g.drawString("10",245,295);//AB weight
            g.drawString("34", 480,280);//BE weight
            g.drawString("32", 552,150);//CE weight
            g.setColor(Color.green);
               g.drawLine(320,350,415,137);//BC
                g.fillOval(300,350,40,40);//B
                 g.setColor(Color.black);
                g.drawString("B", 312,375);
          if (n>=12)
          {

             if (n>=13)
          {

          if (n>=14)
          {


       if (n>=15)
          {




          }}}}}}}}}}}}}}}


          }




}
```

5. **Prims.java:**

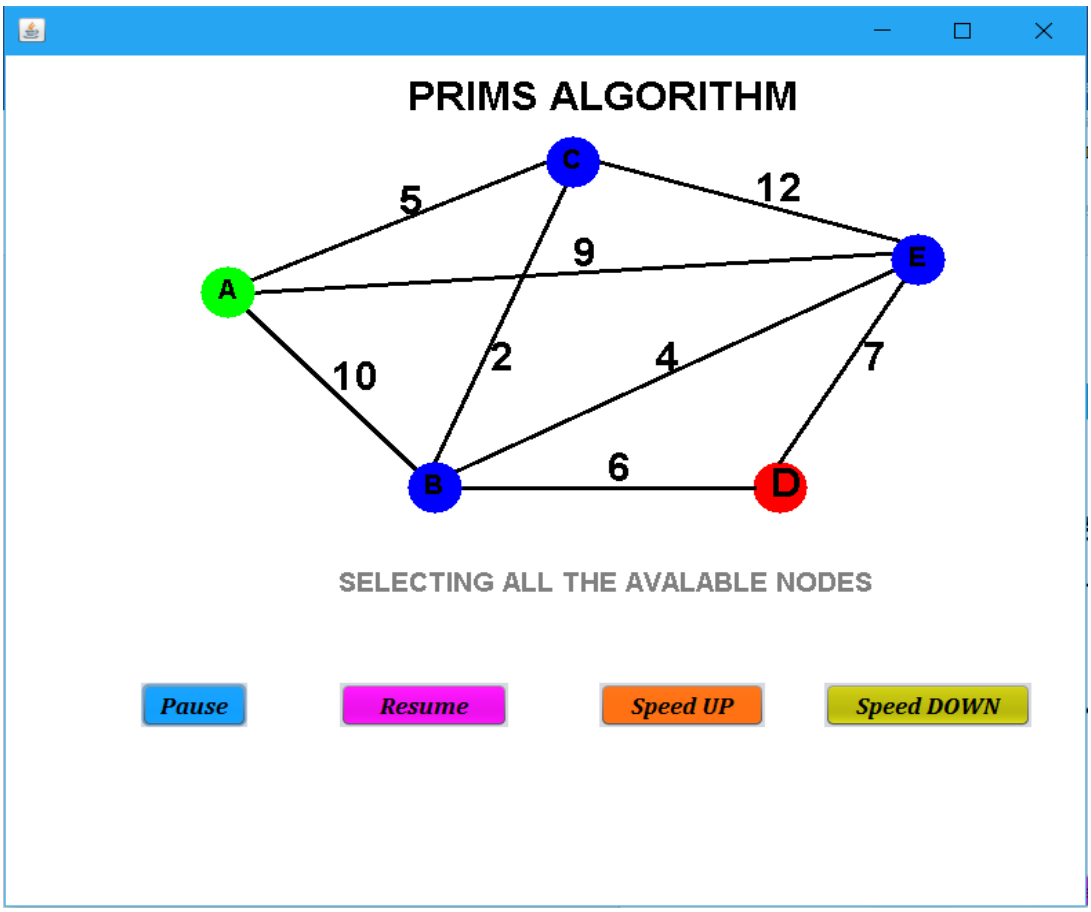


Explanation:

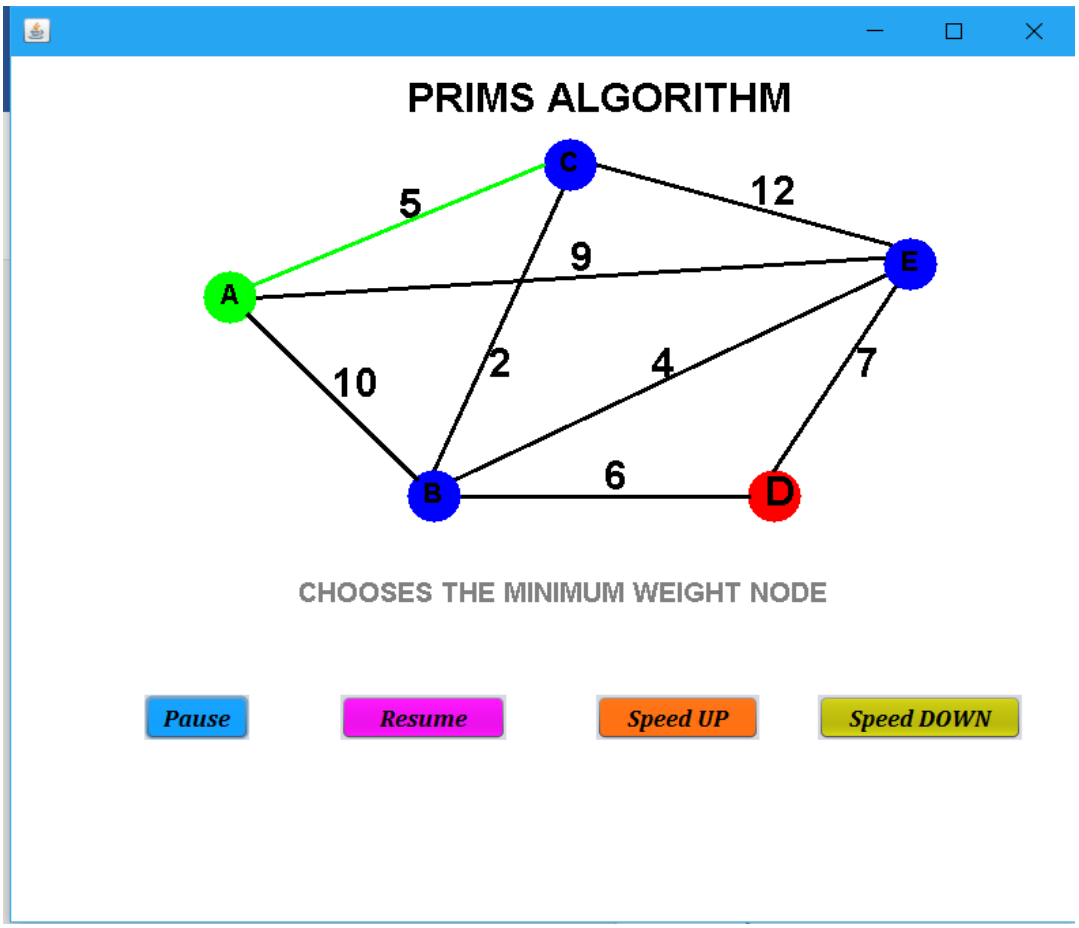
In prims algorithm, we try to find minimum spanning tree for the given nodes here in th example we will start from node A. We implemented this algorithm using threds, which allow us to control thread speed, and execution.When we are at node A it turns green , add all possible traversal nodes becomes blue colour .D is still red i.e. we cannot traverse to D node.



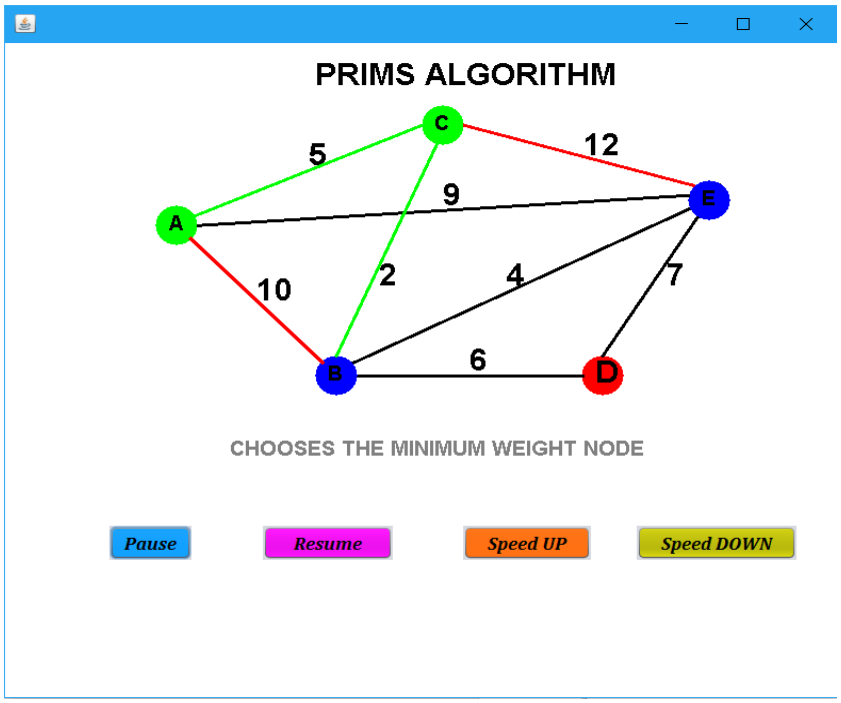
Here starting node is selected, and turned green from red, further the possible traversal nodes will be selected



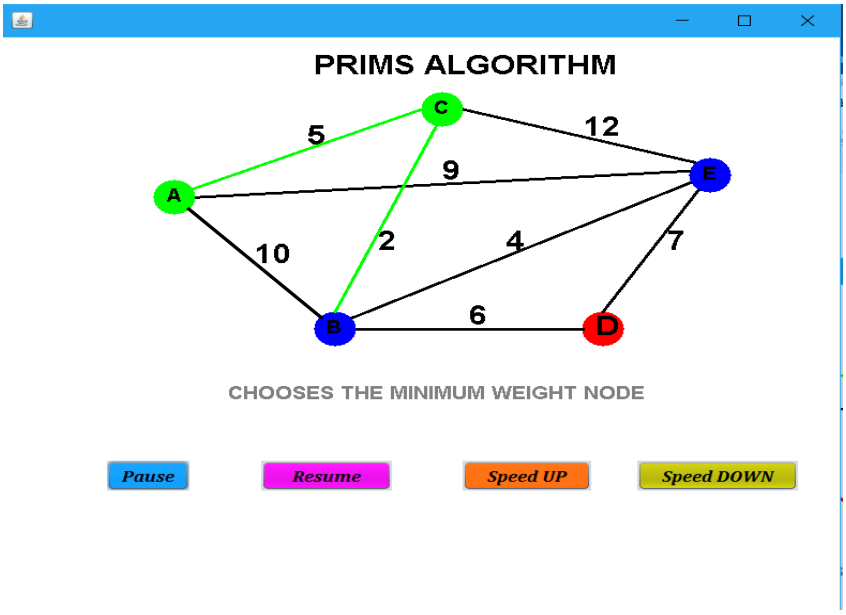
Here in the example we will start from node A. When we are at node A it turns green, add all possible traversal nodes becomes blue colour D is still red i.e. we cannot traverse to D node



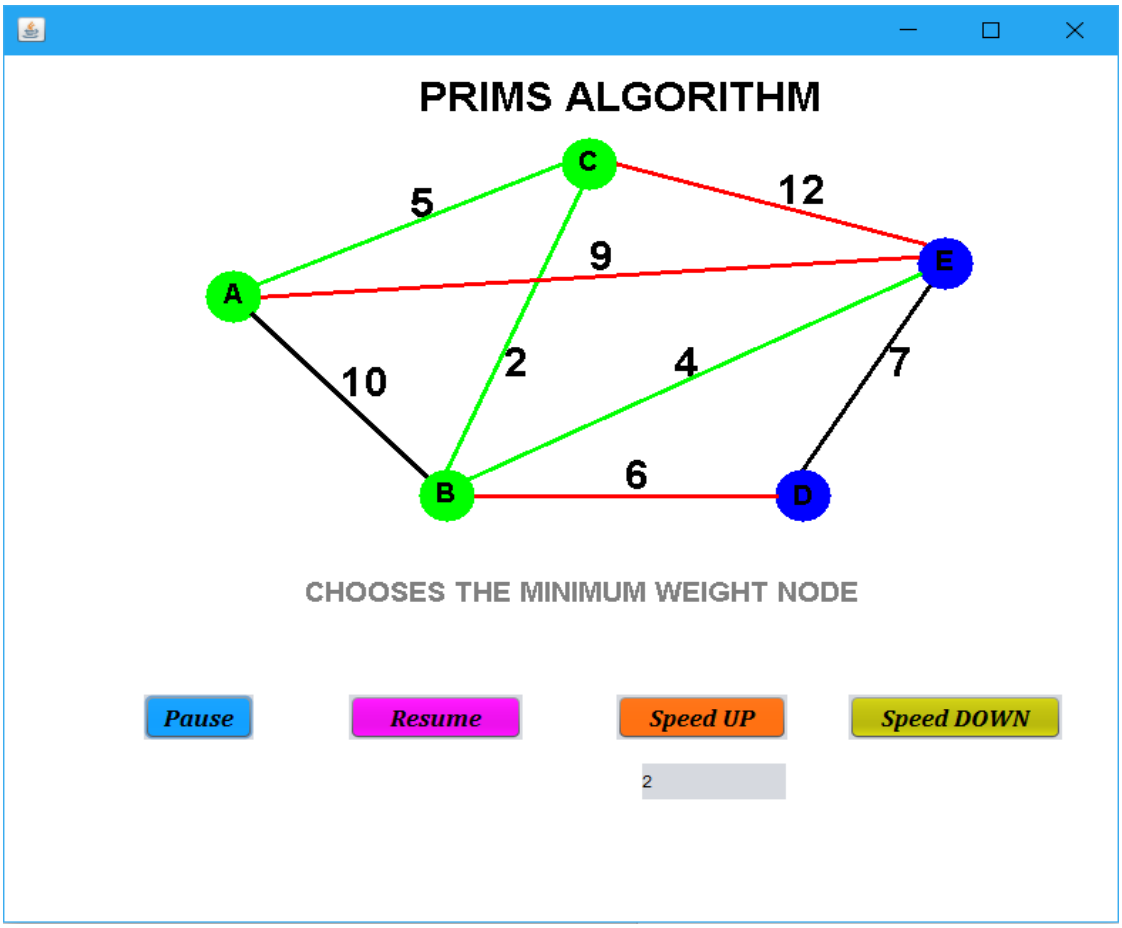
Now the algorithm Chooses the minimum path and rejects other possible traversals and go for node C with minimum weight 5



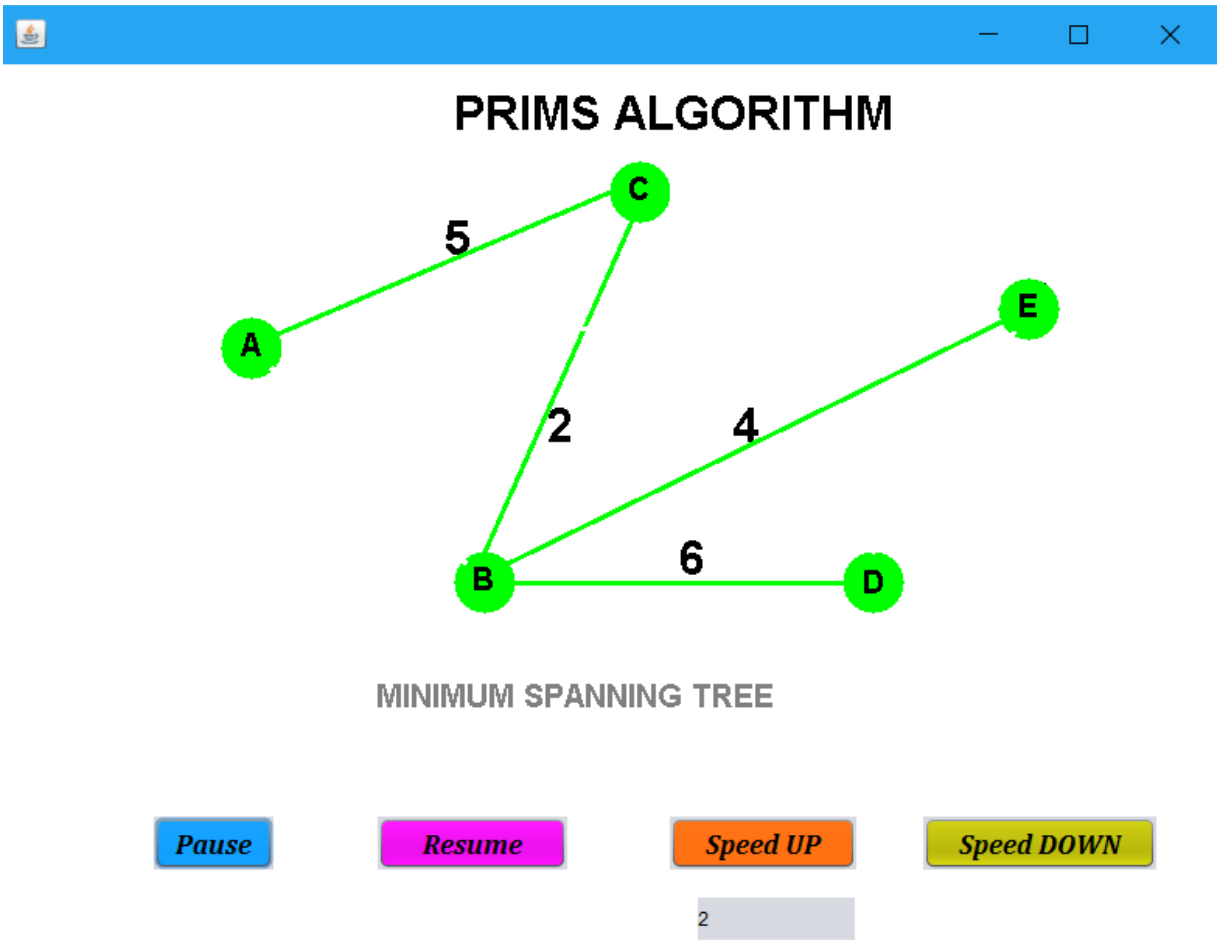
now C node been selected Algorithm will try to find out all possible travers from all nodes without forming a closed loop so possible traversals are to B from A with cost 10, to B from C with cost 2 and from C to E with cost 12. So 2 being the minimum, therefore B will be selected from C



So B is selected as its coast is less than all the available costs similarly preceding same steps we will obtain the minimum spanning tress, the above processes are are repeated for node E and D to obtain minimum spanning tree.



Now after selecting B available possible nodes are E and D so it will find out from which previous nodes will the cost be minimum to travers to E or D. Here the cost to E from B being 4 is minimum so it will chooses to go to node E next and similarly for D from E in net step.



The minimum spanning tree is obtained after repeating above steps multiple times.
Here we also have pause, resume, speed UP and speed Down button

Pause Button: pause button pauses the current thread by calling the function thred.suspend()

Resume Button: the resume button resumes the execution of thread by calling the function thread.resume()

Speed UP Button: it speeds the execution speed of the thread by decreasing the sleep time for thread.

Speed DOWN button: it speeds down the execution speed of the thread by increasing the sleep time for thread.

Code:

```
import java.awt.BasicStroke;
import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.util.logging.Level;
import java.util.logging.Logger;

public class prims extends javax.swing.JFrame implements Runnable {
    int n=-1;
    int m=2000;
    int k=0;

   Thread t;
   /**
    * Creates new form prims
    */
  public void run()
   {
       try {
           t.suspend();
          while(n!=31)
          { repaint();
            t.sleep(m);
            n++;
          }


       } catch (InterruptedException ex) {
          Logger.getLogger(prims.class.getName()).log(Level.SEVERE,
null, ex);
      }


   }
   public prims() {

       initComponents();
       t=new Thread(this);
      t.start();
     setSize(800,700);

   }

   /**
```

```
    * This method is called from within the constructor to initialize the
form.
    * WARNING: Do NOT modify this code. The content of this method
is always
    * regenerated by the Form Editor.
    */
   @SuppressWarnings("unchecked")
   // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN:initComponents
   private void initComponents() {

       jButton1 = new javax.swing.JButton();
       jButton2 = new javax.swing.JButton();
       jButton3 = new javax.swing.JButton();
       jButton4 = new javax.swing.JButton();
       up = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_C
LOSE);
       setPreferredSize(new java.awt.Dimension(900,  400));

       jButton1.setText("Pause");
       jButton1.addMouseListener(new  java.awt.event.MouseAdapter() {
           public void mousePressed(java.awt.event.MouseEvent  evt) {
               jButton1MousePressed(evt);
           }
       });

       jButton2.setText("Resume");
       jButton2.addMouseListener(new  java.awt.event.MouseAdapter() {
           public void mousePressed(java.awt.event.MouseEvent  evt) {
               jButton2MousePressed(evt);
           }
       });

       jButton3.setText("Speed  UP");
       jButton3.addMouseListener(new  java.awt.event.MouseAdapter() {
           public void mousePressed(java.awt.event.MouseEvent  evt) {
               jButton3MousePressed(evt);
           }
       });
       jButton3.addActionListener(new  java.awt.event.ActionListener() {
           public void actionPerformed(java.awt.event.ActionEvent  evt) {
               jButton3ActionPerformed(evt);
           }
       });

       jButton4.setText("Speed  DOWN");
       jButton4.addMouseListener(new  java.awt.event.MouseAdapter() {
```

```
            public void mousePressed(java.awt.event.MouseEvent evt) {
                jButton4MousePressed(evt);
            }
        });

        up.setText("Current speed");

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addGap(98, 98, 98)
                .addComponent(jButton1)
                .addGap(96, 96, 96)
                .addComponent(jButton2,
javax.swing.GroupLayout.PREFERRED_SIZE, 93,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(66, 66, 66)
                .addComponent(jButton3,
javax.swing.GroupLayout.PREFERRED_SIZE, 120,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(43, 43, 43)
                .addComponent(jButton4,
javax.swing.GroupLayout.PREFERRED_SIZE, 150,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(107, Short.MAX_VALUE))
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(up,
javax.swing.GroupLayout.PREFERRED_SIZE, 101,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(233, 233, 233))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
            .addGroup(layout.createSequentialGroup()
                .addGap(482, 482, 482)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
                    .addComponent(jButton1)
```

```
                    .addComponent(jButton2)
                    .addComponent(jButton3)
                    .addComponent(jButton4))
                .addGap(18, 18, 18)
                .addComponent(up,
javax.swing.GroupLayout.PREFERRED_SIZE,  27,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(52, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void jButton1MousePressed(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_jButton1MousePressed
    t.suspend();        // TODO add your handling  code here:
    }//GEN-LAST:event_jButton1MousePressed

    private void jButton2MousePressed(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_jButton2MousePressed
t.resume();         // TODO add your handling  code here:
    }//GEN-LAST:event_jButton2MousePressed

    private void jButton3MousePressed(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_jButton3MousePressed
if (k<=3)
{
m=m-500;
int l=m/500;
 k=4-l;
String tx=k+"";
if(m==0)
{   m=100;
    tx="max speed";
}

up.setText(tx);
}// TODO add your handling  code here:
    }//GEN-LAST:event_jButton3MousePressed

    private void jButton3ActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_jButton3ActionPerformed
        // TODO add your handling  code here:
    }//GEN-LAST:event_jButton3ActionPerformed

    private void jButton4MousePressed(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_jButton4MousePressed
if(m==100)
    m=0;
m=m+500;
```

```
if(k>=1)
 k--;
String tx=k+"";
if(k==0)
{  m=2000;
   tx="Slowest speed";
}
up.setText(tx);
   }//GEN-LAST:event_jButton4MousePressed

   /**
    * @param args the command line arguments
    */
   public static void main(String args[]) {
      /* Set the Nimbus look and feel */
      //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
      /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
       * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
       */
      try {
         for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
               break;
            }
         }
      } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(prims.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
      } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(prims.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
      } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(prims.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
      } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(prims.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
      }
      //</editor-fold>
```

```
    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new  Runnable() {
        public void run() {
            new prims().setVisible(true);
        }
    });
}

public void paint(Graphics g)
{


    Graphics2D g2 = (Graphics2D) g;
   Font  f=new Font("TimesNewRoma",Font.BOLD,20);
    g.setFont(f);

    g2.setStroke(new BasicStroke(3));


    if(n==0){


        f=new Font("TimesNewRoma",Font.BOLD,30);
    g.setFont(f);
      g.drawString("PRIMS ALGORITHM", 300,80);

    g.setColor(Color.red);
   g.fillOval(150,200,40,40);//A
   g.fillOval(300,350,40,40);//B
   g.fillOval(400,100,40,40);//c
   g.fillOval(550,350,40,40);//D
   g.fillOval(650,175,40,40);//E
    g.setColor(Color.black);




    g.drawString("A", 163,225);
    g.drawString("B", 312,375);
    g.drawString("C", 412,125);
    g.drawString("D", 563,377);
    g.drawString("E", 663,200);



   g.drawLine(183,233,307,357);//AB
   g.drawLine(188,210,400,120);//AC
   g.drawLine(190,220,650,190);//AE
   g.drawLine(320,350,415,137);//BC
   g.drawLine(340,370,551,370);//BD
```

```
        g.drawLine(570,350,660,210);//DE
        g.drawLine(334,358,652,203);//BE
        g.drawLine(440,120,655,180);//CE

        g.drawString("10",245,295);//AB weight
        g.drawString("5", 294,160);//AC weight
        g.drawString("9", 420,200);//AE weight
        g.drawString("2", 360,280);//BC weight
        g.drawString("6", 445,365);//BD weight
        g.drawString("7", 630,280);//DE weight
        g.drawString("4", 480,280);//BE weight
        g.drawString("12", 552,150);//CE weight


        }

        else if (n==1)
        {
               g.setColor(Color.green);
               g.fillOval(150,200,40,40);//A
               g.setColor(Color.black);
               g.drawString("A", 163,225);

        }

        else if (n==2)
        {
           g.setColor(Color.gray);
           g.drawString("SELECTING ALL THE AVALABLE NODES
",250,450);
               g.setColor(Color.blue);
               g.fillOval(300,350,40,40);//B
               g.setColor(Color.black);
               g.drawString("B", 312,375);

        }

        else if (n==3)
        {
               g.setColor(Color.blue);
              g.fillOval(400,100,40,40);//C
               g.setColor(Color.black);
              g.drawString("C", 412,125);

        }
        else if (n==4)
        {
               g.setColor(Color.blue);
              g.fillOval(650,175,40,40);//E
               g.setColor(Color.black);
```

```
            g.drawString("E", 663,200);

        }
        else if (n==5)
              {
             g.setColor(Color.white);
            g.drawString("SELECTING ALL THE AVALABLE NODES
",250,450);
            g.setColor(Color.gray);
            g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
                 g.setColor(Color.green);
                g.drawLine(188,210,400,120);//AC
             //   g.drawString("5",415, 95);//over C
                g.setColor(Color.red);
                g.drawLine(183,233,307,357);//AB
             //   g.drawString("10",310,420);//below B
                g.drawLine(190,220,650,190);//AE
              //  g.drawString("9",680, 175);//below E
                }

        else if (n==6)
              {

              g.setColor(Color.black);
              g.drawLine(183,233,307,357);//AB
                g.drawLine(188,210,400,120);//AC
               g.drawLine(190,220,650,190);//AE
               g.setColor(Color.white);
           //     g.drawString("10",310,420);//below B
           //     g.drawString("9",680, 175);//below E
              }
        else if (n==7)
              {

              g.setColor(Color.green);
              g.drawLine(188,210,400,120);//AC
          //    g.drawString("5",415, 95);//over C
              g.setColor(Color.red);
              g.drawLine(183,233,307,357);//AB
             //   g.drawString("10",310,420);//below B
              g.drawLine(190,220,650,190);//AE
            //    g.drawString("9",680, 175);//below E
              }

        else if (n==8)
              {
             //     g.setColor(Color.white);
             //   g.drawOval(285, 140, 30, 30);
              g.setColor(Color.black);
```

```
            g.drawLine(183,233,307,357);//AB
        g.drawLine(188,210,400,120);//AC
        g.drawLine(190,220,650,190);//AE
            }
         else if (n==9)
            {
            g.setColor(Color.green);
            g.drawLine(188,210,400,120);//AC
          //      g.setColor(Color.white);
          //       g.drawString("10",310,420);//below B
           //      g.drawString("9",680, 175);//below E
            }
        else if (n==10)
        {
            g.setColor(Color.white);
            g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
             g.setColor(Color.gray);
            g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);

              g.setColor(Color.green);
             g.fillOval(400,100,40,40);//C
              g.setColor(Color.black);
             g.drawString("C", 412,125);

        }

        else if (n==11)
        {
             g.setColor(Color.white);
            g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);

            g.setColor(Color.gray);
            g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
            g.setColor(Color.green);
            g.drawLine(320,350,415,137);//BC
            g.setColor(Color.red);
            g.drawLine(440,120,655,180);//CE
            g.drawLine(183,233,307,357);//AB
        }
        else if (n==12)
        {
            g.setColor(Color.black);
            g.drawLine(320,350,415,137);//BC

            g.drawLine(440,120,655,180);//CE
            g.drawLine(183,233,307,357);//AB
```

```
        }
          else if (n==13)
        {
            g.setColor(Color.green);
            g.drawLine(320,350,415,137);//BC
            g.setColor(Color.red);
            g.drawLine(440,120,655,180);//CE
            g.drawLine(183,233,307,357);//AB
        }
       else if (n==14)
        {
            g.setColor(Color.black);
            g.drawLine(320,350,415,137);//BC

            g.drawLine(440,120,655,180);//CE
            g.drawLine(183,233,307,357);//AB
        }
    else if (n==15)
        {
            g.setColor(Color.green);
            g.drawLine(320,350,415,137);//BC
        }
    else if (n==16)
        {        g.setColor(Color.white);
            g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
            g.setColor(Color.gray);
            g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);
                g.setColor(Color.green);
               g.fillOval(300,350,40,40);//B
                g.setColor(Color.black);
               g.drawString("B", 312,375);

        }
    else if (n==17)
        {
            g.setColor(Color.white);
            g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);
            g.setColor(Color.gray);
            g.drawString("SELECTING ALL THE AVALABLE NODES
",250,450);

                g.setColor(Color.blue);
               g.fillOval(550,350,40,40);//D
                g.setColor(Color.black);
              g.drawString("D", 563,377);

        }
```

```
   else if (n==18)
      {
         g.setColor(Color.white);
         g.drawString("SELECTING ALL THE AVALABLE NODES
",250,450);
         g.setColor(Color.gray);
         g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
         g.setColor(Color.green);
          g.drawLine(334,358,652,203);//BE
          g.setColor(Color.red);
          g.drawLine(440,120,655,180);//CE
          g.drawLine(190,220,650,190);//AE
          g.drawLine(340,370,551,370);//BD
      }


   else if (n==19)
      {
         g.setColor(Color.black);
          g.drawLine(334,358,652,203);//BE
          g.drawLine(440,120,655,180);//CE
          g.drawLine(190,220,650,190);//AE
          g.drawLine(340,370,551,370);//BD
      }
   else if (n==20)
      {
         g.setColor(Color.green);
          g.drawLine(334,358,652,203);//BE
          g.setColor(Color.red);
          g.drawLine(440,120,655,180);//CE
          g.drawLine(190,220,650,190);//AE
          g.drawLine(340,370,551,370);//BD
      }


   else if (n==21)
      {
         g.setColor(Color.black);
          g.drawLine(334,358,652,203);//BE
          g.drawLine(440,120,655,180);//CE
          g.drawLine(190,220,650,190);//AE
          g.drawLine(340,370,551,370);//BD
      }

   else if (n==22)
      {
         g.setColor(Color.green);
          g.drawLine(334,358,652,203);//BE
```

```
      }
   else if (n==23)
      {
         g.setColor(Color.white);
         g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
         g.setColor(Color.gray);
         g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);
             g.setColor(Color.green);
            g.fillOval(650,175,40,40);//E
             g.setColor(Color.black);
           g.drawString("E", 663,200);

      }
   else if (n==24)
      {
         g.setColor(Color.white);
         g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);
         g.setColor(Color.gray);
         g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
         g.setColor(Color.green);
          g.drawLine(340,370,551,370);//BD
          g.setColor(Color.red);
          g.drawLine(570,350,660,210);//DE
      }
   else if (n==25)
      {
         g.setColor(Color.black);
          g.drawLine(340,370,551,370);//BD
          g.drawLine(570,350,660,210);//DE
      }
    else if (n==26)
      {
         g.setColor(Color.green);
          g.drawLine(340,370,551,370);//BD
          g.setColor(Color.red);
          g.drawLine(570,350,660,210);//DE
      }
   else if (n==27)
      {
         g.setColor(Color.black);
          g.drawLine(340,370,551,370);//BD
          g.drawLine(570,350,660,210);//DE
      }
   else if (n==28)
      {
         g.setColor(Color.green);
```

```
            g.drawLine(340,370,551,370);//BD

        }
    else if (n==29)
        {       g.setColor(Color.white);
                g.drawString("CHOOSES THE MINIMUM WEIGHT NODE
",220,450);
                g.setColor(Color.gray);
            g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);
                 g.setColor(Color.green);
                g.fillOval(550,350,40,40);//D
                 g.setColor(Color.black);
               g.drawString("D", 563,377);

        }
    else if (n==30)
    {
        g.setColor(Color.white);
            g.drawString("SELECTS NODE THE MINIMUM WEIGHT
",220,450);
            g.setColor(Color.gray);
            g.drawString("MINIMUM SPANNING  TREE ",250,450);
        g.setColor(Color.white);
        g.drawLine(183,233,307,357);//AB
        g.drawLine(190,220,650,190);//AE
        g.drawLine(440,120,655,180);//CE
        g.drawLine(570,350,660,210);//DE
        g.drawString("10",245,295);//AB weight
        g.drawString("9", 420,200);//AE weight
        g.drawString("12", 552,150);//CE weight
        g.drawString("7", 630,280);//DE weight
    }

    }
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JLabel up;
    // End of variables declaration//GEN-END:variables
}
```

6. Insertionsort.java:

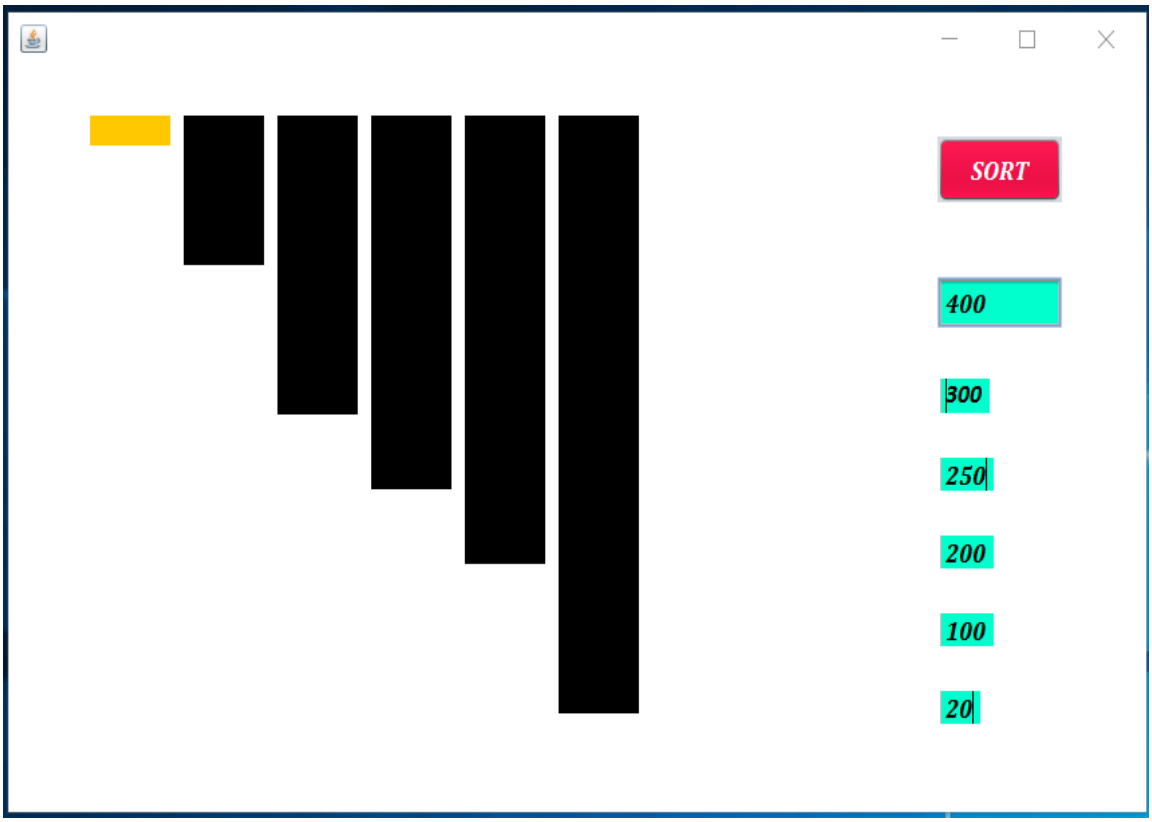
Explanation:

Before sorting:

In insertion sorting the algorithm will sort the given data in increasing order. In our code, we provide the user with 6 input fields and a sort button. When user enters an integer value in any of the given field and moves to the next field corresponding graph bar will be created in left side of screen. Each time when paint method is called some parameters are passed along with them too so as to tell paint method what it has to paint. Along with those repaint methods a del parameter is also passes if we have to delete some element.

After entering the numbers user can click SORT button the sort button uses thread.start so it will call run method which will further call the sorting method and the sorting will start

After sorting:



After clicking SORT button the sorting will start, starting from left index and assigning the 2st element as key, and painting it orange so as to represent which element is currently under sort, it will check if the elements to the left to it are larger is they are larger then it swaps the current key with this element and then comparing it with left next element until it finds suitable place for our key. Once it finds the correct place for key, it moves to next iteration and selects next key and compares and swaps it with its left elements it they are larger than key. this keeps on going until all the elements are sorted completely and we get the fully sorted graph one like as above.

Code:

```
import java.awt.Color;
import java.awt.Graphics;
import java.util.logging.Level;
import java.util.logging.Logger;

/*
 * To change this license header, choose License Headers in Project
Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

/**
 *
 * @author Rubal_cooL
 */
public class insertionsort extends javax.swing.JFrame implements
Runnable{

    /**
     * Creates new form insertionsort
     */
    Thread t=new Thread(this);

   int[] arr1 = {0,0,0,0,0,0}  ;
   int[] arr2={0,0,0,0,0,0};
   int[] arr3={0,0,0,0,0,0};
int m=-1;
int x=0;
int del=0;

int select=0;

    public insertionsort() {
        initComponents();


    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method
is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
```

```
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        A = new javax.swing.JTextField();
        B = new javax.swing.JTextField();
        C = new javax.swing.JTextField();
        D = new javax.swing.JTextField();
        E = new javax.swing.JTextField();
        F = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_C
LOSE);

        A.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
        A.setText("400");
        A.setToolTipText("");
        A.addFocusListener(new java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                AFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent evt) {
                AFocusLost(evt);
            }
        });

        B.setFont(new java.awt.Font("Calibri", 3, 18)); // NOI18N
        B.setText("300");
        B.addFocusListener(new java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                BFocusGained(evt);
            }
            public void focusLost(java.awt.event.FocusEvent evt) {
                BFocusLost(evt);
            }
        });
        B.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                BActionPerformed(evt);
            }
        });

        C.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
        C.setText("250");
        C.addFocusListener(new java.awt.event.FocusAdapter() {
            public void focusGained(java.awt.event.FocusEvent evt) {
                CFocusGained(evt);
            }
```

```
    public void focusLost(java.awt.event.FocusEvent evt) {
        CFocusLost(evt);
    }
});
C.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        CActionPerformed(evt);
    }
});

D.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
D.setText("200");
D.addFocusListener(new java.awt.event.FocusAdapter() {
    public void focusGained(java.awt.event.FocusEvent evt) {
        DFocusGained(evt);
    }
    public void focusLost(java.awt.event.FocusEvent evt) {
        DFocusLost(evt);
    }
});

E.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
E.setText("100");
E.addFocusListener(new java.awt.event.FocusAdapter() {
    public void focusGained(java.awt.event.FocusEvent evt) {
        EFocusGained(evt);
    }
    public void focusLost(java.awt.event.FocusEvent evt) {
        EFocusLost(evt);
    }
});

F.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
F.setText("20");
F.setToolTipText("");
F.addFocusListener(new java.awt.event.FocusAdapter() {
    public void focusGained(java.awt.event.FocusEvent evt) {
        FFocusGained(evt);
    }
    public void focusLost(java.awt.event.FocusEvent evt) {
        FFocusLost(evt);
    }
});

jButton1.setBackground(new java.awt.Color(204, 0, 51));
jButton1.setFont(new java.awt.Font("Cambria", 3, 18)); // NOI18N
jButton1.setForeground(new java.awt.Color(255, 255, 255));
jButton1.setText("SORT");
jButton1.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mousePressed(java.awt.event.MouseEvent evt) {
```

```
            jButton1MousePressed(evt);
        }
    });

    javax.swing.GroupLayout  layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
            .addContainerGap(694, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
                .addComponent(F)
                .addComponent(E)
                .addComponent(D)
                .addComponent(C)
                .addComponent(A)
                .addComponent(B)
                .addComponent(jButton1,
javax.swing.GroupLayout.DEFAULT_SIZE,  93, Short.MAX_VALUE))
            .addGap(63, 63, 63))
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
        .addGroup(layout.createSequentialGroup()
            .addGap(46, 46, 46)
            .addComponent(jButton1,
javax.swing.GroupLayout.PREFERRED_SIZE,  44,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(50, 50, 50)
            .addComponent(A,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(28, 28, 28)
            .addComponent(B,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(C,
javax.swing.GroupLayout.PREFERRED_SIZE,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(D,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(E,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)
                .addComponent(F,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(53, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void AFocusLost(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_AFocusLost
        arr1[0]=Integer.parseInt(A.getText());
        m=1;
        repaint();
    }//GEN-LAST:event_AFocusLost

    private void BActionPerformed(java.awt.event.ActionEvent  evt)
{//GEN-FIRST:event_BActionPerformed
        // TODO add your handling  code here:
    }//GEN-LAST:event_BActionPerformed

    private void BFocusLost(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_BFocusLost
arr1[1]=Integer.parseInt(B.getText());        // TODO add your handling
code here:
  m=1;
        repaint();
    }//GEN-LAST:event_BFocusLost

    private void CFocusLost(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_CFocusLost
arr1[2]=Integer.parseInt(C.getText());        // TODO add your handling
code here:
  m=1;
        repaint();
    }//GEN-LAST:event_CFocusLost
```

```
    private void DFocusLost(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_DFocusLost
arr1[3]=Integer.parseInt(D.getText());        // TODO add your handling
code here:
  m=1;
        repaint();
    }//GEN-LAST:event_DFocusLost

    private void EFocusLost(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_EFocusLost
arr1[4]=Integer.parseInt(E.getText());        // TODO add your handling
code here:
  m=1;
        repaint();
    }//GEN-LAST:event_EFocusLost

    private void FFocusLost(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_FFocusLost
arr1[5]=Integer.parseInt(F.getText());        // TODO add your handling
code here:
  m=1;
        repaint();
    }//GEN-LAST:event_FFocusLost

    private void jButton1MousePressed(java.awt.event.MouseEvent  evt)
{//GEN-FIRST:event_jButton1MousePressed

m=2;
repaint();
for (int i=0;i<=5;++i)
    arr2[i]=arr1[i];
t.start();


    }//GEN-LAST:event_jButton1MousePressed

    private void CActionPerformed(java.awt.event.ActionEvent  evt)
{//GEN-FIRST:event_CActionPerformed
        // TODO add your handling  code here:
    }//GEN-LAST:event_CActionPerformed

    private void AFocusGained(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_AFocusGained
            A.setBackground(new java.awt.Color(0, 255, 204));
    }//GEN-LAST:event_AFocusGained

    private void BFocusGained(java.awt.event.FocusEvent  evt) {//GEN-
FIRST:event_BFocusGained
            B.setBackground(new java.awt.Color(0, 255, 204));
```

```
    }//GEN-LAST:event_BFocusGained

    private void CFocusGained(java.awt.event.FocusEvent evt) {//GEN-
FIRST:event_CFocusGained
        C.setBackground(new java.awt.Color(0, 255, 204));
    }//GEN-LAST:event_CFocusGained

    private void DFocusGained(java.awt.event.FocusEvent evt) {//GEN-
FIRST:event_DFocusGained
        D.setBackground(new java.awt.Color(0, 255, 204));
    }//GEN-LAST:event_DFocusGained

    private void EFocusGained(java.awt.event.FocusEvent evt) {//GEN-
FIRST:event_EFocusGained
        E.setBackground(new java.awt.Color(0, 255, 204));
    }//GEN-LAST:event_EFocusGained

    private void FFocusGained(java.awt.event.FocusEvent evt) {//GEN-
FIRST:event_FFocusGained
        F.setBackground(new java.awt.Color(0, 255, 204));
    }//GEN-LAST:event_FFocusGained

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(insertionsort.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
```

```
java.util.logging.Logger.getLogger(insertionsort.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(insertionsort.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(insertionsort.class.getName()).log(ja
va.util.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new insertionsort().setVisible(true);
            }
        });
    }


    public  void run(){
         insertionSort(arr1);}

      void insertionSort(int array[]) {
        int n = array.length;
        for (int j = 1; j < n; j++) {
            int key = array[j];
            int i = j-1;
            while ( (i > -1) && ( array [i] > key ) ) {
                array [i+1] = array [i];
                for(int k=0;k<=5;++k)
                { if(k==i)
                { arr3[k]=key;
                select=k;
                }

                else
                    arr3[k]=array[k];}
                del=1;
                m=2;

                repaint();
            try {
                t.sleep(500);
            } catch (InterruptedException ex) {
```

```
Logger.getLogger(insertionsort.class.getName()).log(Level.SEVERE,
null, ex);
            }
                i--;
            }
            array[i+1] = key;


        }

    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JTextField A;
    private javax.swing.JTextField B;
    private javax.swing.JTextField C;
    private javax.swing.JTextField D;
    private javax.swing.JTextField E;
    private javax.swing.JTextField F;
    private javax.swing.JButton jButton1;
    // End of variables declaration//GEN-END:variables
public void clear()
{

}
public void paint(Graphics g)
{
if(m==1)
{
    x=70;

    g.setColor(Color.black);


    for(int i=0;i<=5;i++)
    {   System.out.print("|"+arr1[i]);

        g.fillRect(x, 70, 60, arr1[i]);
        x=x+70;

    }
    m=0;
}
    if(del==1)
    {
        x=70;
    g.setColor(Color.white);
    for(int i=0;i<=5;i++)
    {
```

```
        g.fillRect(x, 70, 60, arr2[i]+1000);
        x=x+70;

    }
    del=0;

    }
    if(m==2)
    {
    x=70;
    g.setColor(Color.black);
    for(int i=0;i<=5;i++)
    {
        if(i==select)
        {
            g.setColor(Color.orange);
            g.fillRect(x, 70, 60, arr3[i]);
        x=x+70;
        }
        else
        {
            g.setColor(Color.black);
        g.fillRect(x, 70, 60, arr3[i]);
        x=x+70;
        }
    }
    for(int i=0;i<=5;i++)
    {

        arr2[i]=arr3[i];
    }
    m=0;
}
}
}
```

CONCLUSION

The project submitted is a result of dedicated and enthusiastic efforts from all the team members. Although, we tried to include all the functionalities that we could. The project surely added to our knowledge and we definitely have learnt a lot of things that will help us in the technical fields ahead.

THANK YOU!!