Gracefulness of TP-Tree with Five Levels Obtained by Java Programming

A.Solairaju[1], N. Abdul Ali[2] and S. Abdul Saleem[3]

[1,2] P.G & Research Department of Mathematics, Jamal Mohamed College, Trichirappalli - 620020
[3] P.G & Research Department of Computer Science, Jamal Mohamed College, Trichirappalli - 620020

Abstract- The Gracefulness of Tp-tree with Five levels is obtained by Java Programming.

I. INTRODUCTION

Most graph labeling methods trace their origin to one. There is a function f, a β-valuation of a graph with q edges if f is an injective map from the vertices of G to the set {0, 1, 2 ,…,q} such that when each edge xy is assigned the label $|f(x)-f(y)|$, the resulting edge labels are distinct.

A. Solairaju and K. Chitra [2009] first introduced the concept of edge-odd graceful labeling of graphs, and edge-odd graceful graphs.

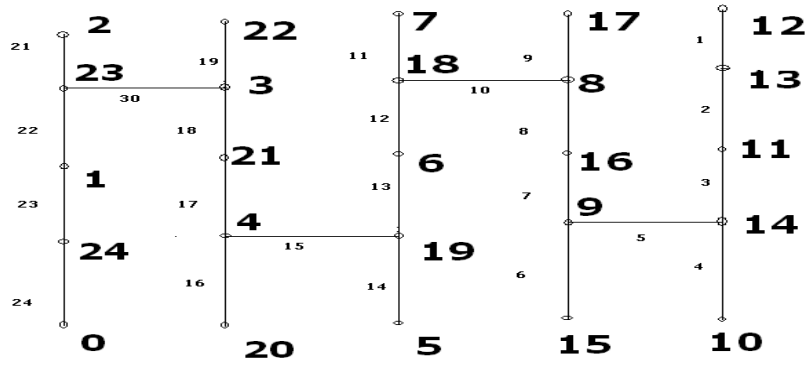
A. Solairaju and others [2008] proved the results that(1) the Gracefulness of a spanning tree of the graph of Cartesian product of P_m and C_n,was obtained (2) the Gracefulness of a spanning tree of the graph of cartesian product of S_m and S_n, was obtained (3) edge-odd Gracefulness of a spanning tree of Cartesian product of P_2 and C_n was obtained (4) Even -edge Gracefulness of the Graphs was obtained (5) ladder P_2 x P_n is even-edge graceful, and (6) the even-edge gracefulness of P_n o nC_5 is obtained.

Section – I: Preliminaries

Definition 1.1 Let G=(V,E) be a simple graph with p vertices and q edges. A map f:V(G) -> {0,1,2,……,q} is called a graceful labeling if

i) f is one-to-one
ii) The edges receive all the labels (numbers) from 1 to q where the label of an edge is the absolute value of the difference between the vertex labels at its end, a graph having a graceful labeling is called a graceful graph.

Example 1.1 : The graph 5 Δ P_5 is a graceful graph.



Section – II: Tp–(5n, 5n-1) tree

Definition 2.1: n Δ P_5 is a tree, becoming a path by moving edges between vertices of degree 3 defined in the following manner only. It is a T_p-tree obtained from n copies of P_5, and connected acyclic in the following manner (figures 1 and 2).

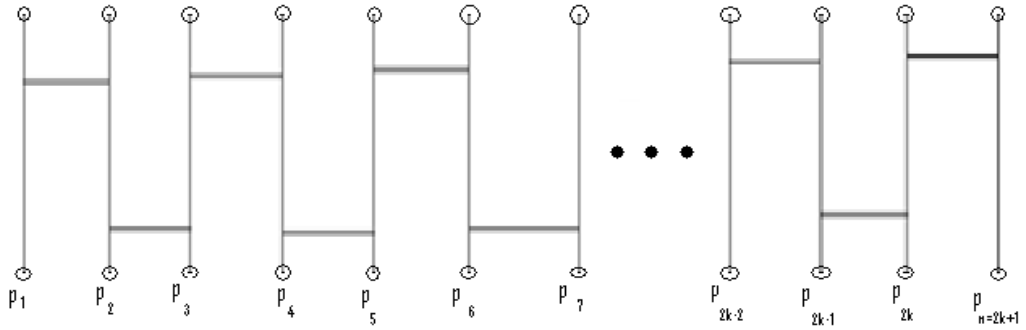


Figure 1 (n is odd)

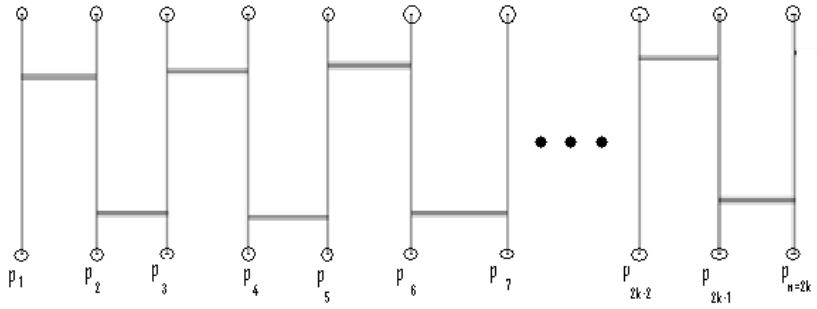


Figure 2 (n is even)

Main theorem 2.2: The connected graph Tp–tree with p = 5n and q = 5n-1 is graceful where n is any positive integer.
Proof: Due to definition (2.1), Tp–(5n,5n-1) is a connected graph (see figures 1 and 2) according as n is odd or even.

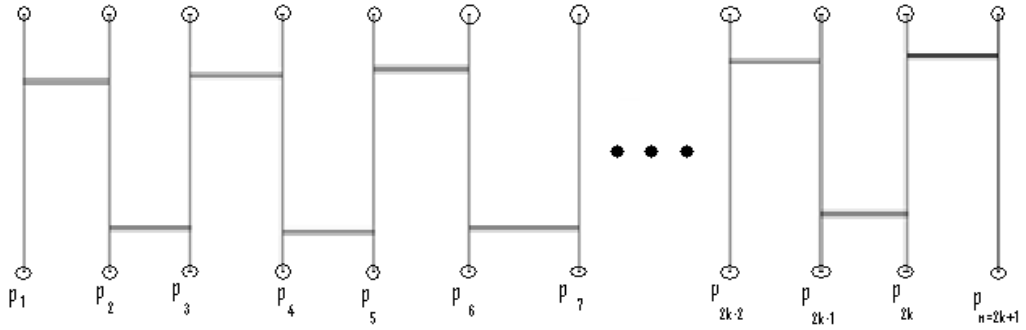


Figure 1 (n is odd)

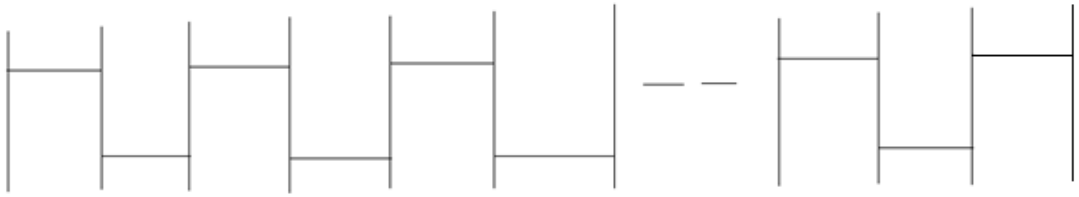
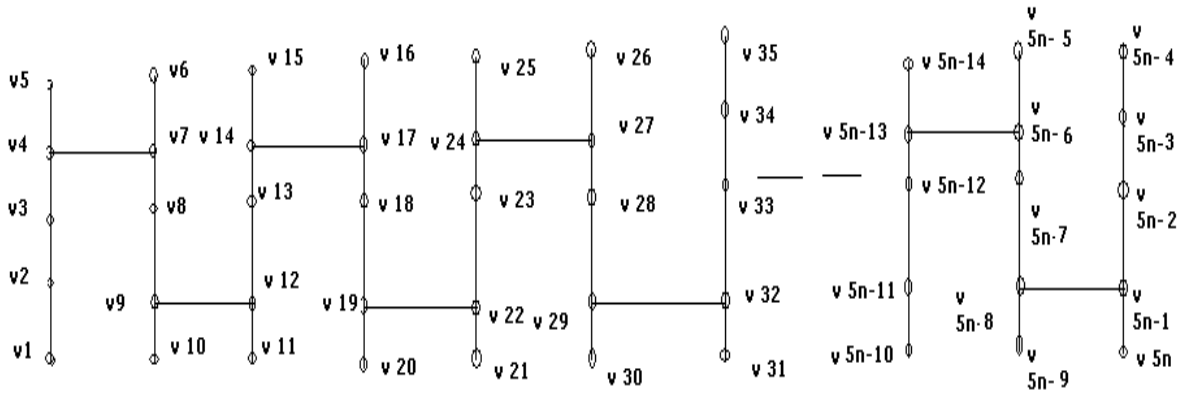


Figure 2 (n is even)

Case (1): n is odd.

The labelings of vertices and edges for Tp–(5n,5n-1) (Figure 1) are as follows:



Define f: v(G) → {0,1,2,…q) by

$$f(v_i) = \frac{i-1}{2} \quad ; \quad i = 2,4,6 \ldots 5n$$

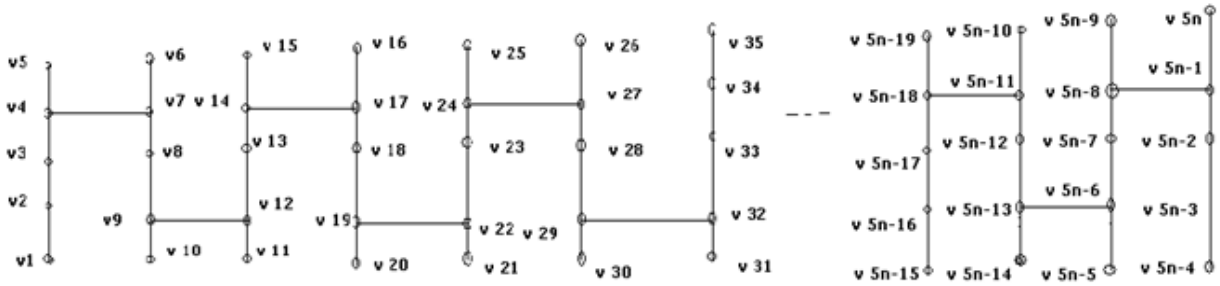
$$f(v_i) = q - \left(\frac{i-1}{2}\right) \quad ; \quad i = 1,3,5 \ldots 5n-1$$

Define $f_+ : E(G) \rightarrow \{1,2,3..q\}$ by $f_+(u,v) = |f(u)-f(v)|$, $\forall$ u,v Є E(G).

Hence, the bisection maps f_+ for vertices and f for edges in Tp–(5n,5n-1) satisfies all condition of graceful labeling. Thus Tp–(5n,5n-1) is a graceful if n is odd.

Case (2): n is even.

The labeling of vertices and edges for Tp–(5n,5n-1) (Figure 2) are as follows:



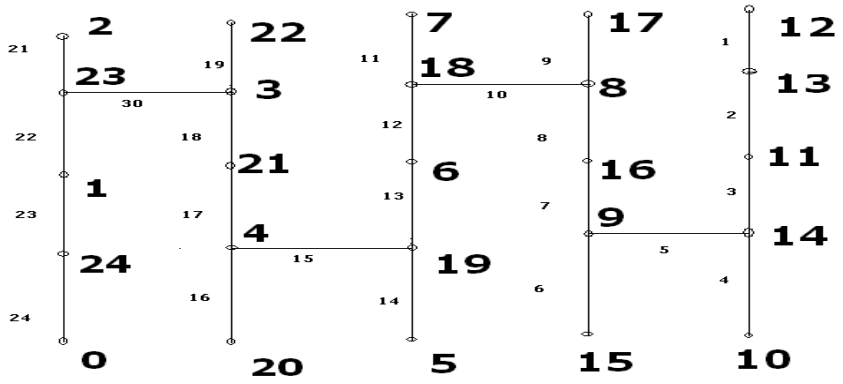
Define f: v(G) → {0,1,2,…q) by

$$f(v_i) = \frac{i-1}{2} \; ; \qquad i = 1,3,5 \ldots 5n-1$$

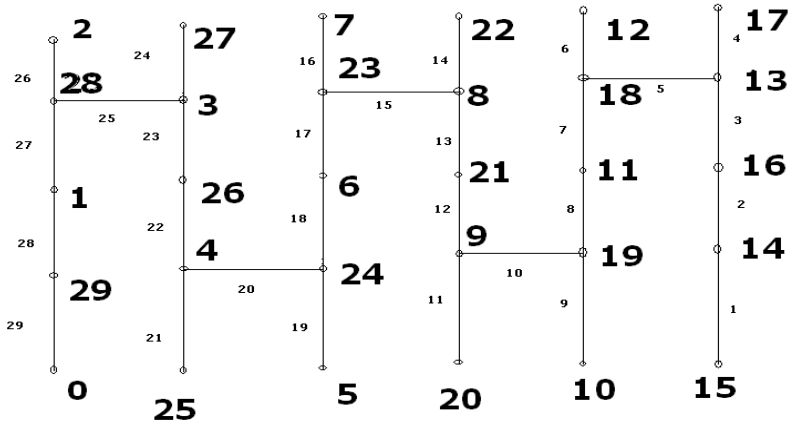
$$f(v_i) = q - \left(\frac{i-1}{2}\right) \; ; \qquad i = 2,4,6 \ldots 5n$$

Define f_+: E(G)→ {1,2,3..q} by $f_+(u,v) = |f(u)-f(v)|$, $\forall$ u,v Є E(G).
Hence, the bisection maps f for vertices and f_+ for edges in Tp–(5n,5n-1) satisfies all the conditions of graceful labeling. Thus, Tp–(5n,5n-1) is a graceful if n is even.
Corollary 2.3: Tp–(5n,5n-1) is a graceful tree

Example 2.1 : The graph 5 Δ P_5 is a graceful graph.



Example 2.2: The graph 6 Δ P_5 is a graceful graph.



Section – III: Java programming for the connected graceful tree Tp–(5n,5n-1)

The Gracefulness of Tp-tree with Five level is obtained by Java Programming

```
import java.awt.*;
import java.awt.event.*;
import java.awt.geom.*;
import javax.swing.*;

public class GFTree extends JApplet implements ActionListener
{


   final static Color bg = Color.white;
   final static Color fg = Color.black;
   JButton b1,b2;
   JLabel l0,l1;
   JTextField tf;
   static JPanel jp1,jp2,jp3,jp4;
   public void init()
   {
     l0 = new JLabel("GRACEFUL GRAPH");
     l0.setFont(new Font("Serif", Font.BOLD, 40));
     l0.setForeground(Color.MAGENTA);
     l1 = new JLabel(" Enter the number of terms : ");
     l1.setFont(new Font("Serif", Font.BOLD, 25));
     l1.setForeground(Color.BLUE);
     tf = new JTextField(20);
     tf.setFont(new Font("Verdana", Font.PLAIN, 25));
     tf.setForeground(Color.BLACK);
     tf.setText("0");
     b1 = new JButton("Submit");
     b1.setForeground(Color.darkGray);
     b1.setFont(new Font("Verdana", Font.PLAIN, 20));
     b1.addActionListener(this);
     b2 = new JButton("Exit");
     b2.setForeground(Color.darkGray);
     b2.setFont(new Font("Verdana", Font.PLAIN, 20));
     b2.addActionListener(this);
     jp1 = new JPanel();
     jp2 = new JPanel();
     jp1.add(l0);
     jp2.setLayout(new GridLayout(2,2));
     jp2.add(l1);
     jp2.add(tf);
     jp2.add(b1);
     b1.setBounds(100,100,200,200);
     jp2.add(b2);
     jp3 = new JPanel();
     jp3.setLayout(new BorderLayout());
     jp3.add(jp1,BorderLayout.NORTH);
     jp3.add(jp2,BorderLayout.SOUTH);
     jp4 = new JPanel();
     setBackground(bg);
     setForeground(fg);
   }

   public void actionPerformed(ActionEvent e)
   {

    if(e.getSource()==b1)
       { start(); repaint();}
    else
```

```
    System.exit(0);
  }
  public void paint(Graphics g)
  {
    g.clearRect(0,135,1024,550);
    Graphics2D g2 = (Graphics2D) g;
         int n = Integer.parseInt((String)tf.getText());
    int v= 5*n;
         int e = (5*n)-1;
    int v1[] = new int[v];
    int v2[] = new int[v];
    for(int i=0;i<v;i++)
      v1[i]=i;
    for(int j=0;j<v;j++)
     { v2[j]=e;e--; }
    int j1=0,j11=0,i1=0;


    for(int i=0;i<n;i++)
    {
    g2.draw(new Line2D.Double((i*50)+100,200,(i*50)+100,400));

    for(int j=0;j<5;j++)
    {
    g2.fillOval((i*50)+98,400-(j*50),5,5);

    }
         if(n>1)
    {
     if(i%3==0)
      { for(int p=0;p<n-1;p+=2)
        g2.draw(new Line2D.Double(100+(p*50),250,150+(p*50),250));
      }
      else
      {
        for(int p=1;p<n-1;p+=2)
        g2.draw(new Line2D.Double(100+(p*50),350,150+(p*50),350));
      }
      }
    }
   int o1=0,o11=0,k1=-1,k2=-1;
   for(int i=0;i<n;i++)
   {
    if(i%2==0)
    {
     for(int j=0;j<3;j++)
     {    int x = j+j1;
          g2.drawString(v1[x]+"",(i*50)+109,412-(j*100));
          o1=x;
     }
          j1+=5;
    }
    else
    {
         for(int j=0;j<2;j++)
          g2.drawString(v1[++o1]+"",(i*50)+102,(j*100)+262);
          o1+=5;

     }
```

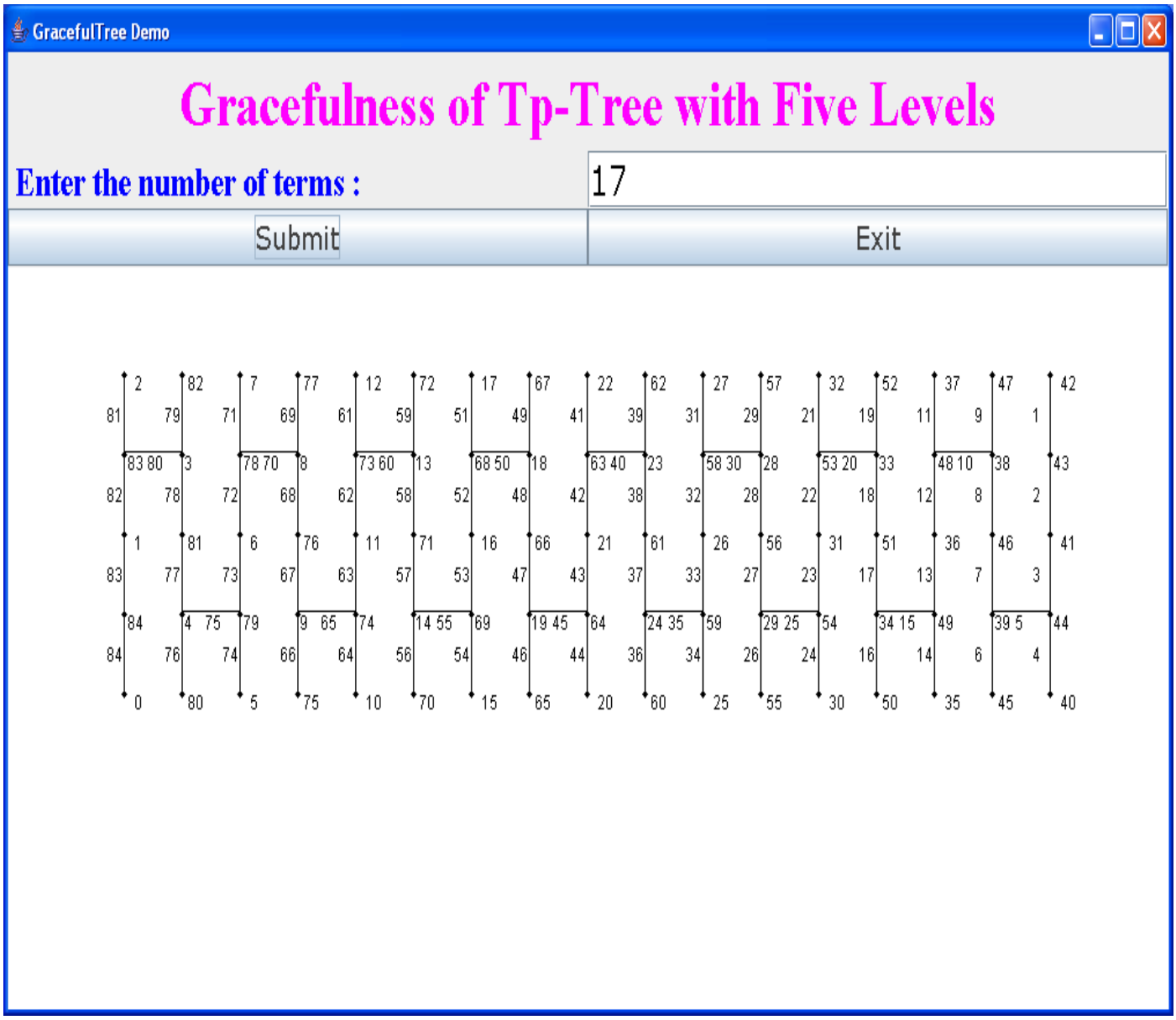
```
    }
     j1=0 ;

    for(int i=0;i<n;i++)
    {
     if(i%2!=0)
     {
       for(int j=0;j<3;j++)
       {
           g2.drawString(v2[++o11]+"",(i*50)+105,(j*100)+212);
       }
           o11+=5;
     }
     else
     {

          for(int j=0;j<2;j++)
          { int x = j+j1;
           g2.drawString(v2[x]+"",(i*50)+103,362- (j*100));
           o11=x;
           }
          j1+=5;
      }
    }
  int d[]= new int[5*n];
for(int i=0;i<d.length;i++)
d[i]=i+1;

int k=-1,y;
boolean flag=true;
if(n%2==0)
{
for(int i=n-1;i>=0;i--)
{

for(int j=5-1;j>=0;j--)
{
 y =d[++k];
    if(y%5==0)
    {
     if(flag)
     {
      g2.drawString(y+"",(i*50)+70,(j*50)+262);
      flag=false;
      }
     else
     {
      if(y<v)
      g2.drawString(y+"",(i*50)+70,(j*50)+362);
      flag=true;
     }
    }
  else
   {if(flag)
    {g2.drawString(y+"",(i*50)+85,(j*50)+182);
    }
    else
    {
```

```
    g2.drawString(y+"",(i*50)+85,432-(j*50));

    }
   }
  } //Inner for loop ends

 } //Outer for loop ends
 }
 else
 {

for(int i=n-1;i>=0;i--)
{

for(int j=5-1;j>=0;j--)
{
 y =d[++k];
    if(y%5==0)
    {
     if(flag)
     {
       if(y<v)
       g2.drawString(y+"",(i*50)+70,(j*50)+362);
       flag=false;
      }
     else
     {
      g2.drawString(y+"",(i*50)+70,(j*50)+262);
      flag=true;
     }
    }
  else
   {if(flag)
    {
    g2.drawString(y+"",(i*50)+85,432-(j*50));
    flag=true;

    }
    else
    {
g2.drawString(y+"",(i*50)+85,(j*50)+182);

    }
   }
  }
 }
}
  }
  public static void main(String s[])
  {
       JFrame f = new JFrame("GracefulTree Demo");
       JApplet applet = new GFTree();
       applet.setLayout(new BorderLayout());
       f.getContentPane().add("Center", applet);
       applet.init();
       applet.add(jp3,BorderLayout.NORTH);
       applet.add(jp4,BorderLayout.SOUTH);
       f.pack();
       f.setSize(600,400);//1024,768);
```

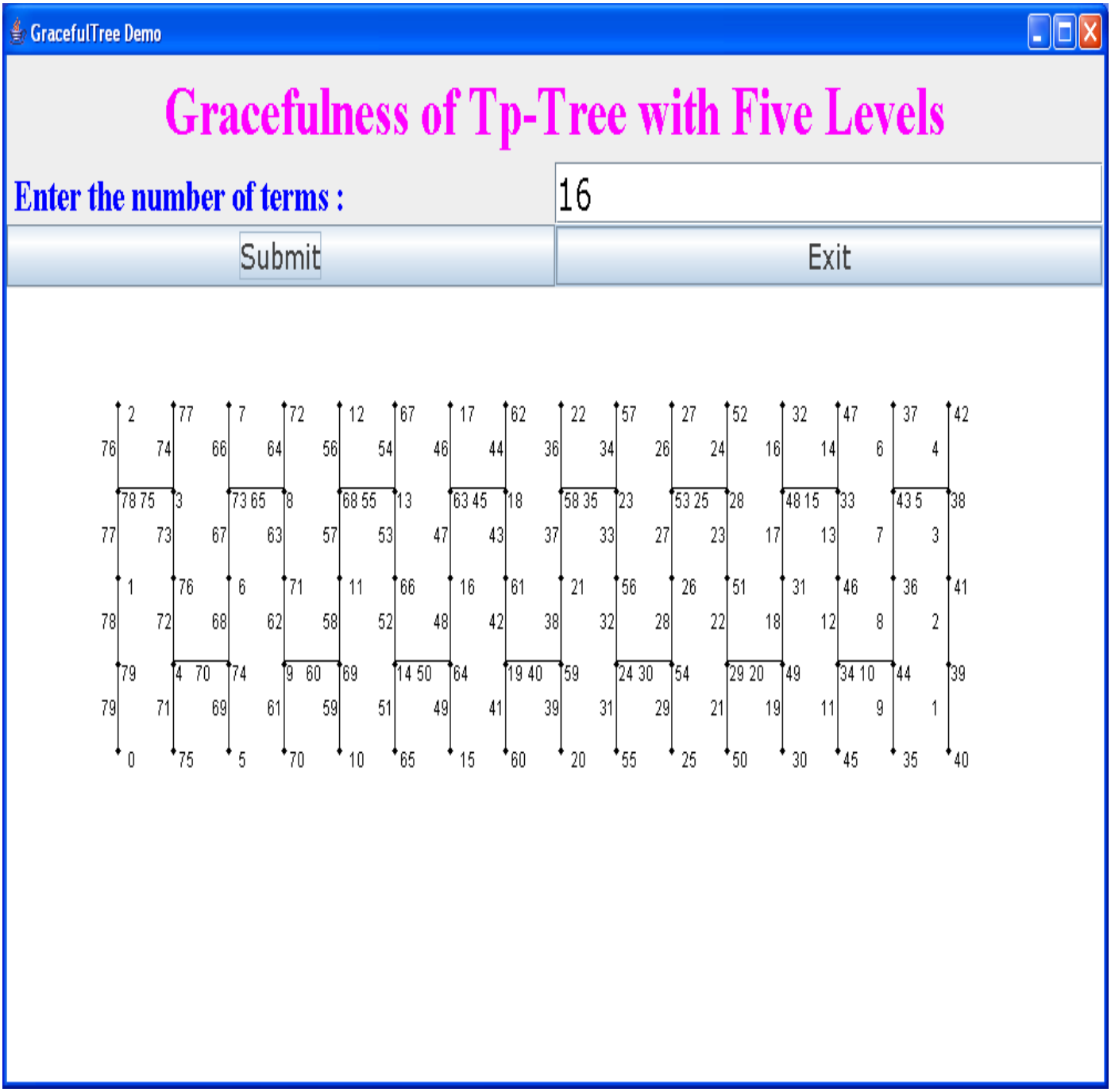
```
        f.setVisible(true);
    }
}
```

Section – IV:

Example 4.1 : The graph $17\Delta P_5$ is a graceful graph.



Example 4.2 : The graph $16\Delta P_5$ is a graceful graph.



REFERENCES

[1] A.Solairaju and K.Chitra Edge-odd graceful labeling of some graphs, Electronics Notes in Discrete Mathematics Volume 33,April 2009,Pages 1.

[2] A. Solairaju and P.Muruganantham, even-edge gracefulness of ladder, The Global Journal of Applied Mathematics & Mathematical Sciences(GJ-AMMS). Vol.1.No.2, (July-December-2008):pp.149-153.

[3] A.Solairaju, A.Sasikala, C.Vimala Gracefulness of a spanning tree of the graph of product of P_m and C_n, The **Global Journal** of Pure and Applied Mathematics of Mathematical Sciences, Vol. 1, No-2 (July-Dec 2008): pp 133-136.

[4] A. Solairaju, C.Vimala,A.Sasikala Gracefulness of a spanning tree of the graph of Cartesian product of S_m and S_n, The **Global Journal** of Pure and Applied Mathematics of Mathematical Sciences, Vol. 1, No-2 (July-Dec 2008): pp117-120.

AUTHORS

First Author – A.Solairaju, P.G & Research Department of Mathematics , Jamal Mohamed College, Trichirappalli – 620020, email: solairama@yahoo.co.in

Second Author – N. Abdul Ali, P.G & Research Department of Mathematics , Jamal Mohamed College, Trichirappalli – 620020, email : abdul_ali_2003@yahoo.com

Third Author – S. Abdul Saleem, P.G & Research Department of Computer Science, Jamal Mohamed College, Trichirappalli – 620020, email : saleem_nts@yahoo.com