The International Journal of Engineering And Science (IJES)
||Volume|| 1 ||Issue|| 2 ||Pages|| 287-296 ||2012||
ISSN: 2319 – 1813 ISBN: 2319 – 1805

An Algorithm for Finding Graceful Labeling For $P_k \circ 2_{C_k}$

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

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

Abstract

In this paper, we obtained that the connected graph $P_k \Delta 2C_4$ is graceful. And also an expression for the java programming of gracefull ness of $p_k \circ 2_{C_k}$

Date of Submission: 27, November, 2012 ⟺ Date of Publication: 30, December 2012

I. Introduction:

Most graph labeling methods trace their origin to one introduced by Rosa [2] or one given Graham and Sloane [1]. Rosa defined 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 others [4,5] 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 $P_2 \times P_n$ is even-edge graceful, and (6) the even-edge gracefulness of $P_n \circ nC_5$ is obtained.

Section 1 : Preliminaries

Definition 1.1: Let G = (V,E) be a simple graph with p vertices and q edges.

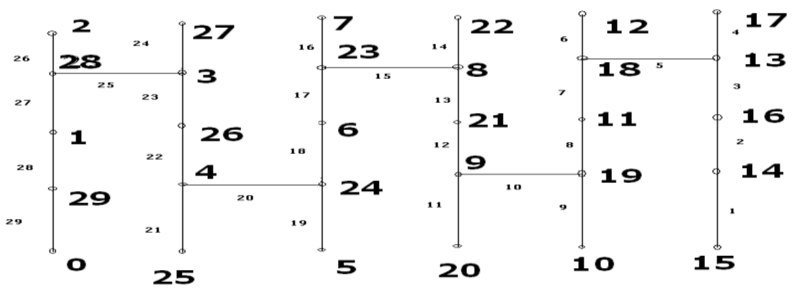
A map $f : V(G) \to \{0,1,2,\ldots,q\}$ is called a graceful labeling if

(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 ends.

A graph having a graceful labeling is called a graceful graph.

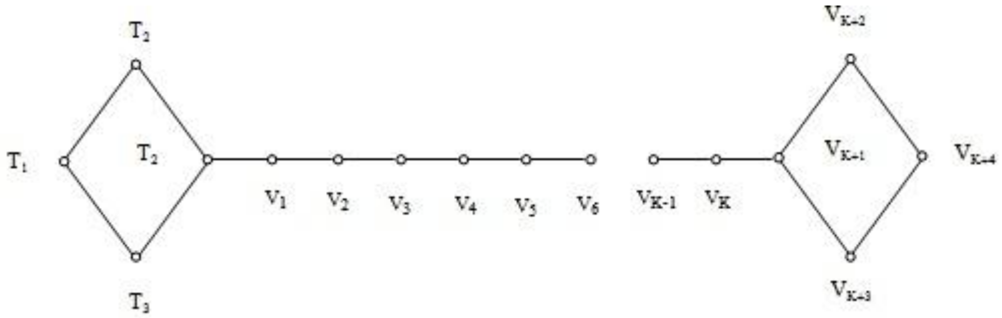
Example 1.1: The graph $6 \Delta P_5$ is a graceful graph.



Section II – Path merging with circuits of length four

Definition 2.1: $P_k \Delta 2C_4$ is a connected graph obtained by merging a circuit of length 4 with isolated vertex of a path of length k.

Theorem 2.1: The connected graph $P_k \Delta 2C_4$ is graceful.



Case (i): k is even.

Define f: V {1,…, q} by

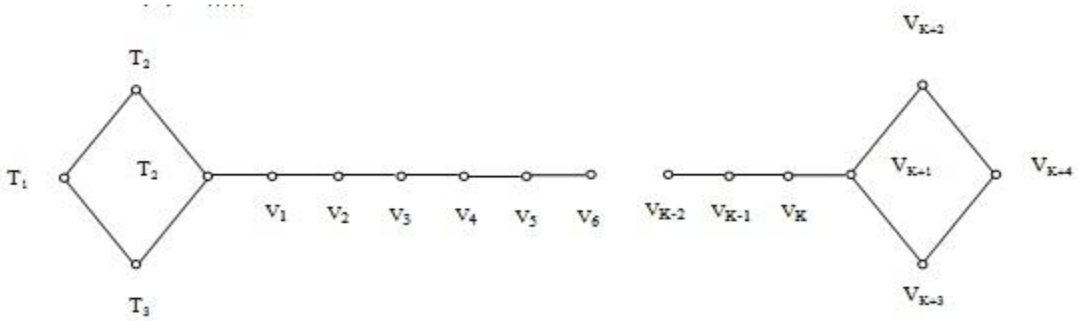
$f(T_1) = 0$; $f(T_2) = q$, $f(T_3) = q-1$, $f(T_4) = 2$

$$f(V_i) = \begin{cases} (q-2) - \left(\frac{i-1}{2}\right), & \text{i is odd, } i = 1,3, \dots, k+1 \\ \left(2+\frac{i}{2}\right), & \text{i is even, } i = 2,4,\dots, k+2 \end{cases}$$

$f(V_{k+3}) = f(V_{k+2}) + 1$

$f(V_{k+4}) = f(V_{k+3}) + 1$

Case (ii): k is odd.



Define f: V {1,…, q}by

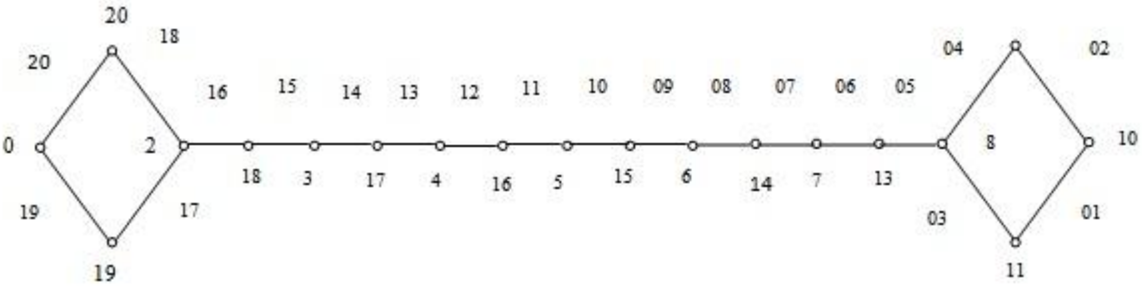
$f(T_1) = 0$; $f(T_2) = q$, $f(T_3) = q-1$, $f(T_4) = 2$

$$f(V_i) = \begin{cases} (q-2) - \left(\frac{i-1}{2}\right), & \text{i is odd, } i = 1,3, \dots, k, k+2 \\ \left(2+\frac{i}{2}\right), & \text{i is even, } i = 2,4,\dots, k+1 \end{cases}$$

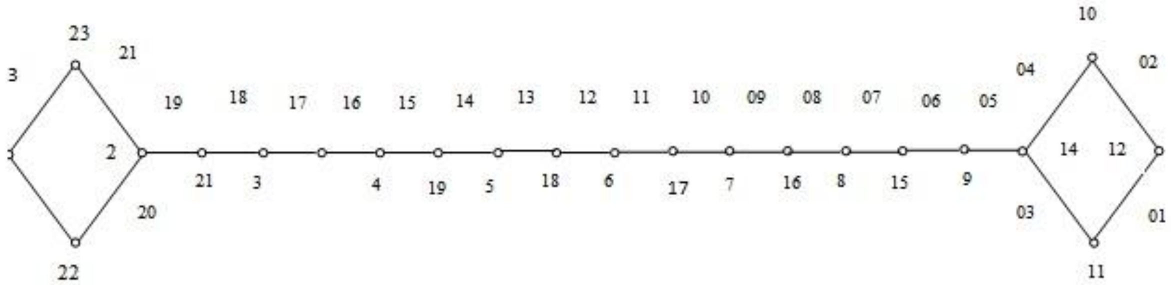
$f(V_{k+3}) = f(V_{k+2}) - 1$

$f(V_{k+4}) = f(V_{k+3}) - 1$

xample 2.1: k = 11 (odd) ; P: V |→ 19; Q: e |→ 20



Example 2.2: k =14 (even) ; P: V |→ 22; Q: e |→ 23



Section 3: AN ALGORITHM IN JAVA PROGRAMMING FOR GRACEFULNESS OF $P_K \circ 2_{C_k}$

```
import java.awt.*;

import java.awt.event.*;

import java.awt.geom.*;

import javax.swing.*;

import java.util.*;

public class GFTree1 extends JApplet implements ActionListener

{

   final static Color bg = Color.white;

   final static Color fg = Color.black;

   static int flag=0;

   JButton b1,b2;

   JLabel l0,l1;

   JTextField tf;

   static JPanel jp1,jp2,jp3,jp4;

   public void init()

   {
```

```
l0 = new JLabel("Gracefulness of Pk o 2Ck");

l0.setFont(new Font("Serif", Font.BOLD, 40));
l0.setForeground(Color.MAGENTA);
l1 = new JLabel(" Enter the value of K : ");
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)
  {
    flag=0;
     g.clearRect(0,135,1024,550);
     Graphics2D g2 = (Graphics2D) g;
          int k = Integer.parseInt((String)tf.getText());
     int v= k+8;
     int e = v+1;

   if(k>0)
  {
     int v1[] = new int[k+4];
     for(int i=0;i<=k+3;i++)
     {int j = i+1;
       v1[i]=j;}
     int j1=0,j11=0,i1=0;
    // Loop for triangle

     int m=0;
```

```
    int x[] = new int[10];
          int y[] = new int[10];
  for(int i=0;i<200;i+=100)
  {
    g2.drawOval(50+i,300,5,5);
     x[m] = 50+i;
     x[m+1] = 300;
     m+=2;
  }
  g2.drawString("0",50,320);
  g2.drawString("2",150,320);
  g2.drawString(e+"",100,240);
  g2.drawString(e+"",70,270);
  g2.drawString(v+"",100,370);
  g2.drawString(v+"",70,340);
  g2.drawString((e-2)+"",130,280);
  g2.drawString((v-2)+"",130,340);

  m=0;
  for(int j=100;j<=200;j+=100)
  {
    g2.drawOval(100,150+j,5,5);
     y[m] = 100;
     y[m+1] = 150+j;
     m+=2;
  }
// Diamond symbol
    for(int i=0;i<=2;i+=2)
    g2.drawLine(x[i],x[i+1],y[i],y[i+1]);
    g2.drawLine(50,300,100,350);
    g2.drawLine(100,250,150,300);
```

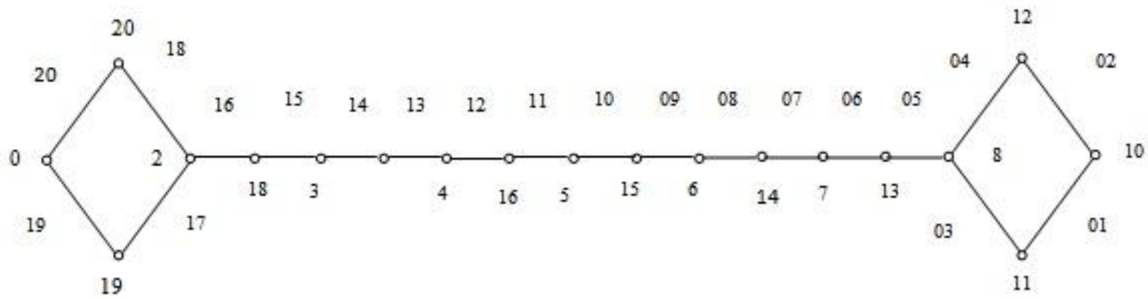
```
int x1=0,y1=0,x2,y2;
  // Line dots
  for(i1=0;i1<k;i1++)
  {
  g2.drawOval(200+i1*50,300,5,5);
   g2.drawLine(150+i1*50,300,250+i1*50,300);
  x1 = 250+i1*50;
  y1 = 300;
  }
int odd=0,even=2,f1=0;
 for(i1=1;i1<=k+1;i1++){
 g2.drawString(f(i1,k)+" ",148+i1*50,320);
 if(i1%2!=0) odd=f(i1,k);
 else even=f(i1,k);
 if(i1<=2) odd=v-1;
 g2.drawString(Math.abs(odd-even)+" ",125+i1*50,290);
 f1 = f(i1,k);
}
 if(k%2==0)
 {
 g2.drawString(Math.abs(f1-even-1)+" ",110+i1*50,275);
 g2.drawString(Math.abs(f1-even-2)+" ",110+i1*50,340);
 g2.drawString(Math.abs(f1-even-3)+" ",175+i1*50,275);
 g2.drawString(Math.abs(f1-even-4)+" ",175+i1*50,340);
 g2.drawString((even+1)+"",148+i1*50,240);
 g2.drawString((even+2)+"",148+i1*50,370);
 g2.drawString((odd-2)+" ",198+i1*50,320);
 }
else
{
g2.drawString(Math.abs(f1-even+4)+" ",110+i1*50,275);
```

```
  g2.drawString(Math.abs(f1-even+3)+" ",110+i1*50,340);
  g2.drawString(Math.abs(f1-even-2)+" ",175+i1*50,275);
  g2.drawString(Math.abs(f1-even-1)+" ",175+i1*50,340);
  g2.drawString((even+4)+"",148+i1*50,240);
  g2.drawString((even+3)+"",148+i1*50,370);
  g2.drawString((even+2)+" ",198+i1*50,320);
  }
   if(x1!=0)
   {
    g2.drawLine(x1,y1,x1+50,350);
    g2.drawLine(x1,y1,x1+50,250);
    g2.drawLine(x1+50,250,x1+100,300);
    g2.drawLine(x1+50,350,x1+100,300);
   }
         // Diamond
    for(int i=k*50;i<k*50+200;i+=100)
    g2.drawOval(200+i,300,5,5);

    for(int j=k*50;j<=k*50+50;j+=100)
    g2.drawOval(250+j,250,5,5);

    for(int j=k*50;j<=k*50+50;j+=100)
    g2.drawOval(250+j,350,5,5);
   }
 }
public static int f(int x,int k1)
 {
  if(flag!=x)
  {
   int v= k1+8;
   int e = v+1;
```

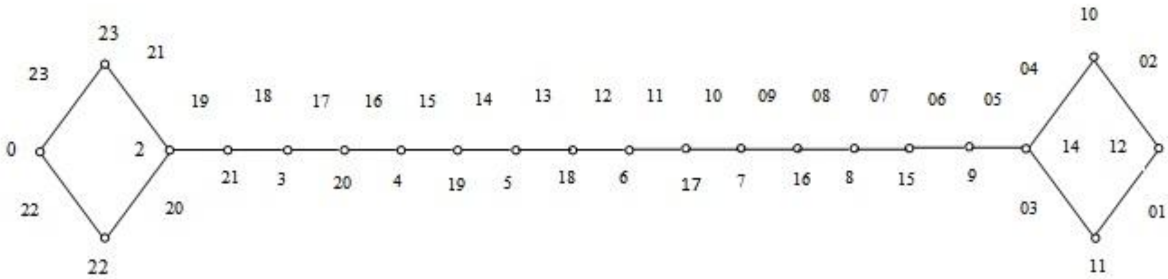
```
flag=x;


if((flag%=2)==0)
{
 if(x<=k1+2)
  { int ev=(2 + ( x / 2));
      return ev;}
  else
   return 0;
 }
 else
    if(x<=k1+1)
    {
    int odd = (e-2)-((x -1) / 2);

    return odd;
    }
   else
     return 0;
  }
 return 0;
 }

public static void main(String s[])
{
     JFrame f = new JFrame("GracefulTree Demo");
     JApplet applet = new GFTree1();
     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(1024,786);

      f.setVisible(true);

  }

 }
}
```

Example 3.1: k = 11 (odd) ; P: V |→ 19; Q: e |→ 20



Example 3.2 : k =14 (even) ; P: V |→ 22; Q: e |→ 23



References:

[1]. R. L. Graham and N. J. A. Sloane, On additive bases and harmonious graph, SIAM J. Alg. Discrete Math., 1 (1980) 382 – 404.

[2]. A. Rosa, On certain valuation of the vertices of a graph, Theory of graphs (International

[3]. Synposium,Rome,July 1966),Gordon and Breach, N.Y.and Dunod Paris (1967), 349-355.

[4]. 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.

[5]. 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.

[6]. Herbert Scheidt , "The Complete ReferenceJava2",2002,Fifth Edition,TATA Mc Graw-Hill

[7]. E.Balagurusamy Programming with java (A Primer). 3rd edition year: 2008, 8th re print .TATA Mc Graw-Hill