**ST.XAVIER’S COLLEGE**

MAITIGHAR, KATHMANDU



**Computer GraphicsAssignment #5**

**Draw a circle using midpointalgorithm**

**Submitted By:**

Divya Jyoti Pokharel

013BSCCSIT019

**Submitted to:**

|  |  |
| --- | --- |
| Er. Anil K. Sah  Lecturer, Department of Computer Science |  |

**Date of Submission: August 21, 2015**

**STATEMENT**

Write a program to draw a circle using the midpoint algorithm.

­

**ALGORITHM**

1. Take center (x0,y0) and radius r as input.
2. Set x=0 and y=radius
3. Calculate the initial decision parameter, P = 1-y
4. Do while (x<=y)
   1. Increment the value of x by 1
   2. If P<0
      * Calculate the next decision parameter as, P = P+2\*x+1
   3. else
      * Calculate the next decision parameter as, P= P+2\*(x-y)+1;
      * Decrement the value of y by 1
   4. Plot pixels at
      * (x0+x,y0+y)
      * (x0+x,y0-y)
      * (x0-x,y0+y)
      * (x0-x,y0-y)
      * (x0+y,y0+x)
      * (x0-y,y0+x)
      * (x0+y,y0-x)
      * (x0-y,y0-x)

**Source code**

//---------------------------------------------------------------------------

#include <vcl\vcl.h>

#pragma hdrstop

#include "assigmnent.h"

//---------------------------------------------------------------------------

#pragma resource "\*.dfm"

TForm1 \*Form1;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

void drawCircle(int x,int y, int xC,int yC,TCanvas \* Canvas )

{

Canvas->Pixels[xC+x][yC+y] = RGB(255,255,0); //1st

Canvas->Pixels[xC+x][yC-y] = RGB(0,0,255); //2nd

Canvas->Pixels[xC-x][yC+y] = RGB(255,0,0); //3rd

Canvas->Pixels[xC-x][yC-y] = RGB(0,255,0); //4th

Canvas->Pixels[xC+y][yC+x] = RGB(0,0,255); //5th

Canvas->Pixels[xC-y][yC+x] = RGB(255,0,255); //6th

Canvas->Pixels[xC+y][yC-x] = RGB(0,255,0); //7th

Canvas->Pixels[xC-y][yC-x] = RGB(0,0,255); //8th

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::createClick(TObject \*Sender)

{

int x,y;

int xC,yC;

int P;

int r=StrToInt(inradius->Text);

xC=StrToInt(inX->Text);

yC=StrToInt(inY->Text);

x=0;

y=r;

P=1-y;

while(x<=y) {

++x; //X is incrememted in every step

if(P<0) {

P+=2\*x+1; //The next decision parameter

} else {

P+=2\*(x-y)+1;

--y; //Plot y one coordinate time the next time

}

drawCircle(x,y,xC,yC,output->Canvas); //Plot the Pixel

}

}

//-----------------------

**Output**

