Assignment No:2.1

Assignment Name:DDA Circle Algorithm.

Name:Komal Pravin killedar.

Roll No:12.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

#include<iostream.h>

#include<conio.h>

#include<stdlib.h>

#include<graphics.h>

#include<math.h>

class CIRCLE

{

int maxx, maxy, r;

public:

void INIT\_GRAPHICS();

void DRAW\_AXIS();

void READ\_RADIUS();

void DDA\_CIRCLE();

};

void CIRCLE::INIT\_GRAPHICS()

{

int gdriver = DETECT, gmode, erroecode;

initgraph(&gdriver, &gmode, "C:\\TURBOC3\\BGI");

}

void CIRCLE::DRAW\_AXIS()

{

maxx = getmaxx();

maxy = getmaxy();

line(0, maxy / 2, maxx, maxy / 2);

line(maxx / 2, 0, maxx / 2, maxy);

}

void CIRCLE::READ\_RADIUS()

{

cout << "\nEnter radius of circle:";

cin >> r;

DDA\_CIRCLE();

}

void CIRCLE::DDA\_CIRCLE()

{

int val,n=0;

float x1,y1,x2,y2;

do

{

val = pow(2, n);

n++;

} while (val<r);

float e=pow(2,-(n - 1));

float start\_x=r;

float start\_y=0;

x1 = start\_x;

y1 = start\_y;

do

{

x2 = x1 +e\*y1;

y2 = y1 -e\*x2;

putpixel(maxx/2+(int)x2,maxy/2 -(int)y2,10);

x1 = x2;

y1 = y2;

} while ((y1 - start\_y)<e ||(start\_x - x1)>e);

}

void main()

{

clrscr();

CIRCLE obj;

obj.INIT\_GRAPHICS();

obj.DRAW\_AXIS();

obj.READ\_RADIUS();

getch();

}