**ST.XAVIER’S COLLEGE**

MAITIGHAR, KATHMANDU



Computer Graphics

Assignment #3

Submitted By:

Aashish Raj Shrestha

013BSCCSIT002

2nd year/ 4th semester

Submitted to:

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

**STATEMENT**

Write a program to draw a line using DDA algorithm in C++ builder.

­

**ALGORITHM**

DDA Line (X1, Y1, XN, YN):

Description: Here X1 and Y1 denote the starting x – coordinate and y – coordinate of the line

And XN and YN denote the ending x – coordinate and y – coordinate.

1. Set M = (YN – Y1) / (XN – X1) [Calculate slope of line]

2. Repeat For I = X1 to XN

3. If (M <= 1) Then

4. Set DX = 1

5. Set DY = M \* DX

6. Else

7. Set DY = 1

8. Set DX = DY / M

[End of If]

9. Set X1 = X1 + DX

10. Set Y1 = Y1 + DY

11. Call PutPixel(X1, Y1)

[End of For]

12. Exit

**SOURCE CODE**

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

#include <vcl\vcl.h>

#pragma hdrstop

#include "DDA.h"

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

#pragma resource "\*.dfm"

TForm1 \*Form1;

int x1,y1,x2,y2;

void draw();

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

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

: TForm(Owner)

{

}

void draw()

{

int dx, dy, steps, m;

float incrx, incry,x,y;

dx=x2-x1;

dy=y2-y1;

if(abs(dx)>abs(dy))

steps=abs(dx);

else

steps=abs(dy);

incrx=dx/steps;

incry=dy/steps;

x=x1;

y=y1;

Form1->Canvas->Pixels[x][y]= RGB(255,0,0);

for(m=1;m<=steps;m++)

{

x=x + incrx;

y=y + incry;

Form1->Canvas->Pixels[x][y]=RGB(255,0,0);

}

}

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

{

x1=StrToInt(Edit1->Text);

y1=StrToInt(Edit2->Text);

x2=StrToInt(Edit3->Text);

y2=StrToInt(Edit4->Text);

draw();

}

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

**OUTPUT:**

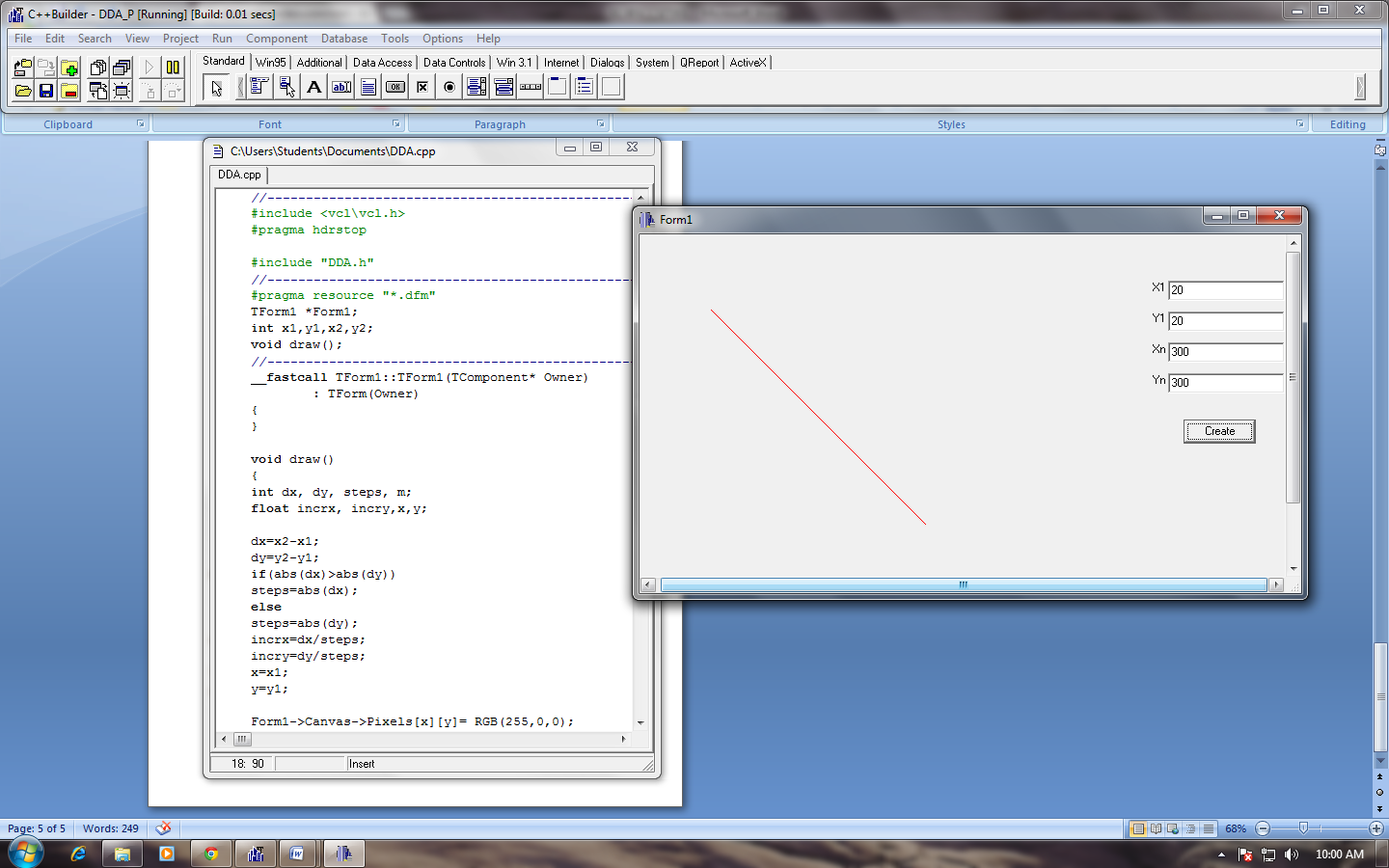
****

Figure I: Line Drawn using DDA Algorithm

**CONCLUSION**

Hence, a program to draw a line was implemented using DDA algorithm in C++ builder.