ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Computer Graphics**

**Lab Assignment#3**

**Submitted by:**

**Shreesha Pokharel**

013BSCCSIT038

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Anil K. Sah** |  |

Lecturer

Department of Computer Science

Date of submission: 14th August, 2015

**Statement: Implement the DDA line algorithm**

**Algorithm:**

Step1: Input x1, y1, x2, y2

Step 2: Calculate dx = x2 – x1

Step 3: Calculate dy = y2 – y1

Step 4: if(abs(dx)>abs(dy))

Set n=abs(dx)

else

Set n=abs(dy);

Step 5: calculate inx as inx=dx/n

Step 6: Calculate iny as iny=dy/n

Step 7: Set x=x1 and y=y1

Step 8: Plot pixel as

Image1->Canvas->Pixels[x][y]=RGB(100,50,100);

Step 9: Set i as 1

Step 10: check if i<n.

Yes: Set x=x+inx

Set y=y+iny

Plot pixel as Image1->Canvas->Pixels[x][y]=RGB(100,50,100);

Increase the value of i by 1 and go to step 10.

No: terminate.

Step 11: End.

**Source code:**

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

#include <vcl\vcl.h>

#pragma hdrstop

#include "Unit1.h"

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

#pragma resource "\*.dfm"

TForm1 \*Form1;

int x1,y1,x2,y2,dx,dy,n,x,y,i;

float inx,iny;

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

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

: TForm(Owner)

{

}

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

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

x1=StrToInt(Edit1->Text);

y1=StrToInt(Edit2->Text);

x2=StrToInt(Edit3->Text);

y2=StrToInt(Edit4->Text);

dx=x2-x1;

dy=y2-y1;

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

n=abs(dx);

else

n=abs(dy);

inx=dx/n;

iny=dy/n;

x=x1;

y=y1;

Image1->Canvas->Pixels[x][y]=RGB(100,50,100);

for(i=1;i<=n;i++)

{

x=x+inx;

y=y+iny;

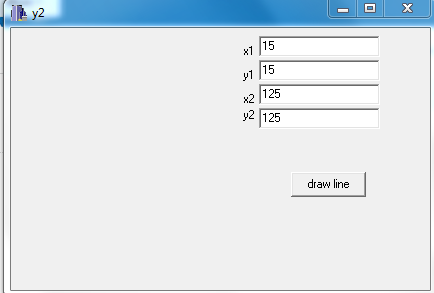
Image1->Canvas->Pixels[x][y]=RGB(100,50,100);

}

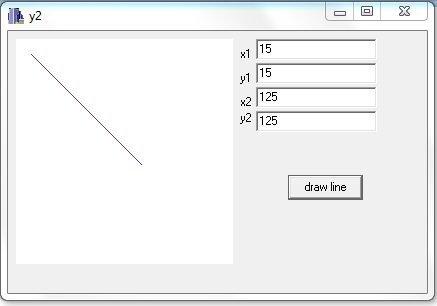
}

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

**Input:**

****

**Output:**



**Conclusion:**

Therefore, the DDA line algorithm was implemented as shown.

**Reference:**

[1] D. Hearn and M. Baker, Computer Graphics, second edition.