Lab 1

```
/* program to draw a line by using build in function line ()*/

    #include<stdio.h>

    #include<graphics.h>

    #include<conio.h>

    #include<dos.h> //for including delay function


    void main()

    {

    int gd = DETECT,gm;

    //gd = detects best available graphics driver, gm=graphics mode


    initgraph(&gd,&gm,"C:\\TurboC3\\BGI"); // initializing graph mode


    line(100,100,200,200); //draw a line segment

    getch();


    }
```

Lab 2

DDA line drawing Algorithm

```
void lineDDA (in x1, int y1, int x2, int y2)
{
        int dx, dy, steps, k;
        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;   /* first point to plot */
        y = y1;
        putpixel(round(x), round(y),1);          //1 is color parameter
        for (k = 1;k <= steps;k++)
        {
                x = x + incrx;
                y = y + incry;
                putpixel(round(x),round(y),1);
        }
}
```

/* C- Program for DDA line drawing algorithm */

```
#include <graphics.h>

#include <stdio.h>

#include <conio.h>

#include <math.h>

void main() {

        int gd = DETECT, gm = DETECT, s, dx, dy, m, x1, y1, x2, y2;

        float xi, yi, x, y;

        clrscr();

        printf("Enter the starting point x1 & y1\n");

        scanf("%d%d", &x1, &y1);
```

```
        printf("Enter the end point x2 & y2\n");
        scanf("%d%d", &x2, &y2);
        initgraph(&gd, &gm, "C:\\TURBOC3\\BGI");
        cleardevice();
        dx = x2 - x1;
        dy = y2 - y1;
        if (abs(dx) > abs(dy))
                s = abs(dx); else
                s = abs(dy);
        xi = dx / (float) s;
        yi = dy / (float) s;
        x = x1;
        y = y1;
        putpixel(x1, y1, 4);
        for (m = 0; m < s; m++) {
                x += xi;
                y += yi;
                putpixel(x, y, 4);
        }
        getch();
}
```

Lab 3

Bresenham line Drawing Algorithm

```
void lineBresenham (int x1, int y1, int x2, int y2){
        int dx = abs(x2-x1), dy=abs(y2-y1);
        int  pk, xEnd;
        pk=2*dy-dx;
        //determine which point to use as start, which as end
        if(x1>x2){
                x = x2;
                y = y2;
                xEnd = x1;
        }
         else {
                x = x1;
                y = y1;
                xEnd = x2;
        }
        putixel (x,y,1);
        while (x < xEnd)
        {
                x++;
                if(pk<0)
                        pk=pk+2*dy;
                else
                {
                        Y++;
                        pk= pk+2*dy-2*dx
                 }
                putpixel (x,y,1);
         }
}
```

/* C- Program for Bresenham Line Drawing Algorithm */

```
#include<graphics.h>

#include<stdio.h>

#include<conio.h>


void main()

{

int x,y,x1,y1,delx, dely, m, grtr_d,smlr_d,d;
```

```
int gm,gd=DETECT;
initgraph(&gd, &gm, "C:\\TurboC3\\BGI");

printf("enter initial coordinate = ");
scanf("%d%d",&x,&y);

printf("enter final coordinate = ");
scanf("%d%d",&x1,&y1);
delx=x1-x;
dely=y1-y;

grtr_d=2*dely-2*delx; //when d>0

smlr_d=2*dely; //when d<0

d=(2*dely)-delx;

do{
putpixel(x,y,1);
if(d<0){
d=smlr_d+d;
}
else{
d=grtr_d+d;
y=y+1;
```

```
}
x=x+1;

}while(x<x1);
getch();

}
```