OPEN GL COMMANDS:

* To set the bit value corresponding to a specified screen position within the frame buffer:
  + **setPixel(x,y)**
* **Direct Scan Conversion:**

x = x1;

while (x<=xr){

ytrue=mx+b;

y=Round(ytrue);

PlotPixel(x,y);

X=x+1;

}

* **DDA Algorithm**

#include<stdlib.h>

#include<stdio.h>

#include <GL/gl.h>

#include <GL/glut.h>

float x1, x2, y1, y2;

void display(void) {

float dy, dx, step, x, y, k, Xin, Yin;

dx = x2 - x1;

dy = y2 - y1;

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

step = abs(dx);

} else

step = abs(dy);

Xin = dx / step;

Yin = dy / step;

x = x1;

y = y1;

glBegin(GL\_POINTS);

glVertex2i(x, y);

glEnd();

for (k = 1; k <= step; k++) {

x = x + Xin;

y = y + Yin;

glBegin(GL\_POINTS);

glVertex2i(x, y);

glEnd();

}

glFlush();

}

void myInit (void) {

glClearColor(0.0, 0.0, 0.0, 0.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

int main(int argc, char \*\* argv) {

printf("Value of x1 : ");

scanf("%f", & x1);

printf("Value of y1 : ");

scanf("%f", & y1);

printf("Value of x2 : ");

scanf("%f", & x2);

printf("Value of y2 : ");

scanf("%f", & y2);

glutInit( & argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(640, 480);

glutInitWindowPosition(100, 150);

glutCreateWindow("");

myInit ();

glutDisplayFunc(display);

glutMainLoop();

}

* **Bresenham’s Line Drawing Algorithm**

setPixel(xo,yo);

dx=xf-xo;dy=yf-yo;

dx2=2\*dx;dy2=2\*dy;

Po=dy2-dx;

K=0; pk=po;x=xo;y=yo;

Do while(x<xf){

x=x++;

if(pk<0)

pk=pk+dy2;

else

{ pk=pk+dy2-dx2;

y++;}

setPixel(x,y)

}

* **BRESNHAM’S CIRCLE DRAWING ALGORITHM**

Int x=0,y=0,d=3-2r;

While (x<=y){

setPixel(x,y);

if(d<0)

d=d+4x+6;

else{

d=d+4(x-y)+10;

y--;

}

x++;

}

* **MIDPOINT CIRCLE ALGORITHM**

Int x=0, y=r, p=1-r;

while(x<=y){

setPixel(x,y);

if(p<0)

p=p+2x+3;

else{

p=p+2(x-y)+5;

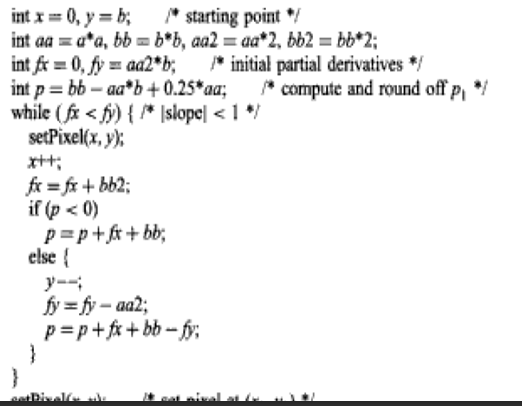
y--;

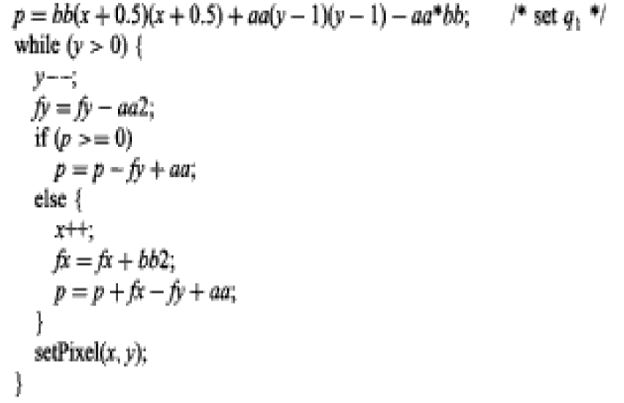
}

x++;

}

* **MIDPOINT ELLIPSE DRAWING ALGORITHM**

****

****