```
// Draw the rectangle pattern using bresenham line drawing

#include <GL/glut.h>
#include <stdio.h>
#include <math.h>
#define round(a) (int)(a+0.5)

int X1,Y1,X2,Y2,xmax,ymax;

void init()
{
      glClearColor(0,0,0,0);
      glColor3f(1.0,1.0,0.0);
      gluOrtho2D(0,1024,0,768);
}

void setpixel(int x,int y)
{
      glBegin(GL_POINTS);
      glVertex2i(x,y);
      glEnd();
}

void bresenham(int X1,int X2,int Y1,int Y2)
{
      int s1,ex,s2,m,x,y,k=1;
      int dx,dy,p,temp;

      dx=abs(X2-X1);
      dy=abs(Y2-Y1);

      x=X1;
      y=Y1;

      s1=sign(X2-X1);
      s2=sign(Y2-Y1);

      if(dy>dx)
      {
              temp=dx;
              dx=dy;
              dy=temp;
              ex=1;
      }
      else
              ex=0;

      p=2*dy-dx;   //Initial value of decision parameter p0

      while(k<=dx)
      {
              setpixel(x,y);
```

```
                if(p>=0)
                {
                        if(ex==1)
                                x=x+s1;
                        else
                                y=y+s2;
                        p=p-2*dx;
                }
                if(ex==1)
                        y=y+s2;
                else
                        x=x+s1;

                p=p+2*dy;
                k++;
        }
}

int sign(int x)
{
        if(x>0)
              return 1;
        else if(x==0)
                return 0;
        return -1;
}

void primitives(void)
{
        int a,b,c,d,e,f;
        glClear(GL_COLOR_BUFFER_BIT);

        xmax=glutGet(GLUT_WINDOW_WIDTH);
        ymax=glutGet(GLUT_WINDOW_HEIGHT);

        bresenham(0,xmax,ymax/2,ymax/2);    // X - Axis
        bresenham(xmax/2,xmax/2,0,ymax);     //Y-axis

        bresenham(X1,X1,Y1,Y2);
        bresenham(X1,X2,Y2,Y2);
        bresenham(X2,X2,Y2,Y1);
        bresenham(X2,X1,Y1,Y1);

        a=(X1+X2)/2;
        b=(Y1+Y2)/2;

        bresenham(a,X1,Y1,b);
        bresenham(X1,a,b,Y2);
        bresenham(a,X2,Y2,b);
```

```
	bresenham(X2,a,b,Y1);

	c=((3*X1)+X2)/4;
	d=((3*X2)+X1)/4;
	e=((3*Y1)+Y2)/4;
	f=((3*Y2)+Y1)/4;

	bresenham(c,d,e,e);
	bresenham(c,c,e,f);
	bresenham(c,d,f,f);
	bresenham(d,d,f,e);

	glFlush();
}

int main( int argc,char **argv)
{
	printf("Enter the value of x1 : ");
	scanf("%d",&X1);
	X1=X1+512;

	printf("Enter the value of y1 : ");
	scanf("%d",&Y1);
	Y1=Y1+384;

	printf("Enter the value of x2 : ");
	scanf("%d",&X2);
	X2=X2+512;

	printf("Enter the value of y2 : ");
	scanf("%d",&Y2);
	Y2=Y2+480;

	glutInit(&argc,argv);
	glutInitDisplayMode(GLUT_SINGLE);
	glutInitWindowPosition(0,0);
	glutInitWindowSize(1024,768);

	glutCreateWindow("primitives");
	init();
	glutDisplayFunc(primitives);
	glutMainLoop();
}
```