```
/* Name:-
   Roll_No:- 01
   Div: A
   AIM:-Write C++/Java program to fill polygon using scan line algorithm. */


#include<graphics.h>
#include<iostream>

using namespace std;

void floodFill(int x, int y, int old, int fill)
{
          int current;
          current=getpixel(x,y);
          if(current==old)
          {
                    putpixel(x,y,fill);
                    //delay(1);
                    floodFill(x+1,y,old,fill);
                    floodFill(x-1,y,old,fill);
                    floodFill(x, y+1,old,fill);
                    floodFill(x,y-1,old,fill);
          }
}
void boundaryfill(int x, int y, int fcolor,int bcolor)
    {
     if(getpixel(x,y)!=fcolor && getpixel(x,y)!=bcolor)
        { //delay(1);
         putpixel(x,y,fcolor);
         boundaryfill(x+1,y,fcolor,bcolor);
            boundaryfill(x,y+1,fcolor,bcolor);
            boundaryfill(x-1,y,fcolor,bcolor);
            boundaryfill(x,y-1,fcolor,bcolor);
}
}

void scanline(int x1,int y1,int x2, int y2)
{
int x,y;
setcolor(RED);
rectangle(x1,y1,x2,y2);
x=x1+1;
```

```
y=y1+1;
x2--;
y2--;
setcolor(YELLOW);
while(y2>=y)
{
moveto(x,y);
lineto(x2,y);
y++;
//delay(500);
}
}

int main()
{
                int x,y,o=0,x1,y1,x2,y2,ch;
                //clrscr();
                int gd=DETECT,gm;

        cout<<"Enter the coordinates of rectangle:";
        cin>>x1>>y1>>x2>>y2;



        cout<<"\n Fill color in polygon.....\n";
                cout<<"1.Flood fill algorithm.....\n";
        cout<<"2.Boundary fill algorithm.....\n";
        cout<<"3.Scan line Algorithm.....\n";
        cout<<"Enter Ur Choice.....\n";
                cin>>ch;
          initgraph(&gd,&gm, NULL);
        switch(ch)
                {
                   case 1:  setcolor(RED);
                              rectangle(x1,y1,x2,y2);
                           x=(x1+x2)/2;
                           y=(y1+y2)/2;
                      floodFill(x,y,o,YELLOW);
                      break;

                   case 2:  setcolor(RED);
                           rectangle(x1,y1,x2,y2);
                           x=(x1+x2)/2;
```

```
                    y=(y1+y2)/2;
                boundaryfill(x,y,GREEN,RED);
                break;
            case 3:
                scanline(x1,y1,x2,y2);
                break;
        }
            getch();
        delay(500000);
            closegraph();
        return 0;
}
```