```
#include<iostream>
#include<GL/glut.h>
#include<stdlib.h>
#include<math.h>
using namespace std;
int v, wxmin, wxmax, wymin, wymax, a[10][2];

void init(void)
{
	glClearColor(0.0,0.0,0.0,0.0);
	gluOrtho2D(-600.0,600.0,-600.0,600.0);
}

int accept_poly(int a[10][2])
{
	int i,n;
	cout<<"\nEnter the number of vertices : ";
	cin>>n;
	for(i=0;i<n;i++)
	{
		cout<<"Enter the coordinates"<<i + 1 << " : "	;
		cin>>a[i][0]>>a[i][1];
	}
	return n;
}
int leftclip(int a[10][2],int v,int wxmin)
{
	int i,t[10][2];
	int k=0;
	int x1,y1,x2,y2;
	for(i=0;i<v;i++)
	{
		if(a[i][0] >= wxmin)
		{
			t[k][0]=a[i][0];
			t[k][1]=a[i][1];
			k++;
		}
		if((a[i][0] > wxmin && a[(i+1)%v][0] < wxmin) || (a[i][0] < wxmin &&
a[(i+1)%v][0] > wxmin))
		{
			x1=a[i][0];
			y1=a[i][1];
			x2=a[(i+1)%v][0];
			y2=a[(i+1)%v][1];
```

```
                t[k][1] = y1 + (wxmin-x1) * ((float)(y2-y1)/(x2-x1));
                t[k][0] = wxmin;
                k++;
            }
        }
        for(i=0;i<k;i++)
        {
            a[i][0] = t[i][0];
            a[i][1] = t[i][1];
        }
        return(k);
}

int rightclip(int a[10][2],int v,int wxmin)
{
        int i,t[10][2];
        int k=0;
        int x1,y1,x2,y2;
        for(i=0;i<v;i++)
        {
            if(a[i][0] <= wxmax)
            {
                t[k][0]=a[i][0];
                t[k][1]=a[i][1];
                k++;
            }
            if((a[i][0] < wxmax && a[(i+1)%v][0] > wxmax) || (a[i][0] > wxmax
&& a[(i+1)%v][0] < wxmax))
            {
                x1=a[i][0];
                y1=a[i][1];
                x2=a[(i+1)%v][0];
                y2=a[(i+1)%v][1];
                t[k][1] = y1 + (wxmax-x1) * ((float)(y2-y1)/(x2-x1));
                t[k][0] = wxmax;
                k++;
            }
        }
        for(i=0;i<k;i++)
        {
            a[i][0] = t[i][0];
            a[i][1] = t[i][1];
        }
        return(k);
}
```

```
int topclip(int a[10][2],int v,int wymax)
{
	int i,t[10][2];
	int k=0;
	int x1,y1,x2,y2;
	for(i=0;i<v;i++)
	{
		if(a[i][1] <= wymax)
		{
			t[k][0]=a[i][0];
			t[k][1]=a[i][1];
			k++;
		}
		if((a[i][1] < wymax && a[(i+1)%v][1] > wymax) || (a[i][1] > wymax
&& a[(i+1)%v][1] < wymax))
		{
			x1=a[i][0];
			y1=a[i][1];
			x2=a[(i+1)%v][0];
			y2=a[(i+1)%v][1];
			t[k][1] = wymax;
			t[k][0] = x1 + (wymax-y1) * ((float)(x2-x1)/(y2-y1));
			k++;
		}
	}
	for(i=0;i<k;i++)
	{
		a[i][0] = t[i][0];
		a[i][1] = t[i][1];
	}
	return(k);
}

int bottomclip(int a[10][2],int v,int wymin)
{
	int i,t[10][2];
	int k=0;
	int x1,y1,x2,y2;
	for(i=0;i<v;i++)
	{
		if(a[i][1] >= wymin)
		{
			t[k][0]=a[i][0];
			t[k][1]=a[i][1];
```

```
                k++;
            }
            if((a[i][1] > wymin && a[(i+1)%v][1] < wymin) || (a[i][1] < wymin &&
a[(i+1)%v][1] > wymin))
            {
                x1=a[i][0];
                y1=a[i][1];
                x2=a[(i+1)%v][0];
                y2=a[(i+1)%v][1];
                t[k][1] = wymin;
                t[k][0] = x1 + (wymin-y1) * ((float)(x2-x1)/(y2-y1));
                k++;
            }
        }
        for(i=0;i<k;i++)
        {
        a[i][0] = t[i][0];
        a[i][1] = t[i][1];
        }
        return(k);
}

void draw_poly(int a[10][2],int v)
{
        int i;
        glBegin(GL_LINES);
        glVertex2i(wxmin,wymin);
        glVertex2i(wxmin,wymax);
        glVertex2i(wxmin,wymax);
        glVertex2i(wxmax,wymax);
        glVertex2i(wxmax,wymax);
        glVertex2i(wxmax,wymin);
        glVertex2i(wxmax,wymin);
        glVertex2i(wxmin,wymin);
        glEnd();
        glFlush();
        for(i=0;i<v;i++)
        {
                glColor3f(1.0,1.0,1.0);
                glBegin(GL_LINES);
                glVertex2i(a[i][0],a[i][1]);
                glVertex2i(a[(i+1)%v][0],a[(i+1)%v][1]);
                glEnd();
                glFlush();
        }
```

```
}
int main(int argc,char** argv)
{
	int cl;
	char ans;
	cout<<"\nEnter the co-ordinates of the window";
	cout<<"\n(Wxmin, Wymin) : ";
	cin>>wxmin >> wymin;
	cout<<"\n(Wxmax, Wymax) : ";
	cin>>wxmax >> wymax;
	v = accept_poly(a);
	glutInit(&argc,argv);
	glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
	glutInitWindowSize(800,800);
	glutInitWindowPosition(100,100);
	glutCreateWindow("Clipping");
	init();
	glClear(GL_COLOR_BUFFER_BIT);
	draw_poly(a,v);
	glFlush();
	do
	{
		cout<<"\nWhich clipping do you want to perform??\n1. Left\n2. Right\
n3. Top\n4.Bottom\nEnter your choice : ";
		cin>>cl;
		switch(cl)
		{
		case 1 :
			glClear(GL_COLOR_BUFFER_BIT);
			v=leftclip(a,v,wxmin);
			glClear(GL_COLOR_BUFFER_BIT);
			draw_poly(a,v);
			break;
		case 2 :
			glClear(GL_COLOR_BUFFER_BIT);
			v=rightclip(a,v,wxmax);
			glClear(GL_COLOR_BUFFER_BIT);
			draw_poly(a,v);
			break;
		case 3 :
			glClear(GL_COLOR_BUFFER_BIT);
			v=topclip(a,v,wymax);
			glClear(GL_COLOR_BUFFER_BIT);
			draw_poly(a,v);
			break;
```

```
            case 4 :
                glClear(GL_COLOR_BUFFER_BIT);
                v=bottomclip(a,v,wymin);
                glClear(GL_COLOR_BUFFER_BIT);
                draw_poly(a,v);
                break;
        }
        cout<<"\nDo you want to perform another clipping?? (Y/N) ";
        cin>>ans;
        }while(ans=='y'||ans=='Y');
        glFlush();
        glutMainLoop();
        return 0;
}
```