

Aditya Krishna P  
185001010  
GM Lab - Test

Code :

```
//  
// main.cpp  
// labtest  
//  
// Created by test1 on 23/10/21.  
//  
#include<GLUT/GLUT.h>  
#include <iostream>  
using namespace std;  
  
float coordSign(float n){  
    if(n == 0)  
        return 0;  
  
    if(n > 0)  
        return 1;  
  
    else return -1;  
  
}  
  
void Circle(int xc, int yc, int r){  
    int p, x, y;  
    x = 0;  
    y = r;  
    p = 1-r;  
    while(x<y)  
    {  
        x++;  
        if(p<0){  
            p = p + 2*x + 1;  
        }  
        else{  
            y--;  
            p = p + 2*x + 1 - 2*y;  
        }  
    }  
    glBegin(GL_POINTS);
```

```

        glVertex2i(x+xc,y+yc);
        glVertex2i(y+xc,x+yc);
        glVertex2i(-y+xc,x+yc);
        glVertex2i(-x+xc,y+yc);
        glVertex2i(-x+xc,-y+yc);
        glVertex2i(-y+xc,-x+yc);
        glVertex2i(y+xc,-x+yc);
        glVertex2i(x+xc,-y+yc);
        glEnd();
    }
    glFlush();
}

```

```

void Line(int x0, int y0, int x1, int y1){
    int x = x0;
    int y = y0;
    int dx = abs(x1-x0);
    int dy = abs(y1-y0);

    int sign_x = coordSign(x1-x0);
    int sign_y = coordSign(y1-y0);

    int swap = -1;

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

    int p = (2*dy)/dx;
    int fca = 2*dy;
    int fcb = (2*dy)-(2*dx);

    glBegin(GL_POINTS);
    glVertex2i(x,y);

    for(int i=1;i<dx;i++){
        if(p<0){
            if(swap == 1){

```

```

        y += sign_y;
    }
    else{
        x += sign_x;
    }

    p = p + fca;
}
else{
    y += sign_y;
    x += sign_x;
    p = p + fcb;
}
glVertex2i(x,y);
}
glEnd();
glFlush();
}

```

```

void myDisplay(){
    Line(10, 20, 30, 40);
    Line(30,40,65,40);
    Line(65, 40, 85, 20);
    Line(85, 20, 85, 10);
    Line(85,10,74,10);
    Circle(70, 10, 6);
    Line(64,10,34,10);
    Circle(30, 10, 6);
    Line(10, 20, 10, 10);
    Line(10, 10, 30, 10);
}

```

```

void myinit(){
    glClearColor(0.0, 0.0, 0.0, 0.0);
    glColor3f(1.0f,1.0f,1.0f);
    glPointSize(2);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(0.0,640.0,0.0,480.0);
    glClear(GL_COLOR_BUFFER_BIT);

}

```

```
int main(int argc, char * argv[]) {  
    // insert code here...  
    glutInit(&argc, argv);  
    glutInitDisplayMode(GLUT_SINGLE|GLUT_RGB);  
    glutInitWindowSize(640,480);  
    glutCreateWindow("Car");  
    glutDisplayFunc(myDisplay);  
    myinit();  
    glutMainLoop();  
  
    return 1;  
}
```

Screenshot :

