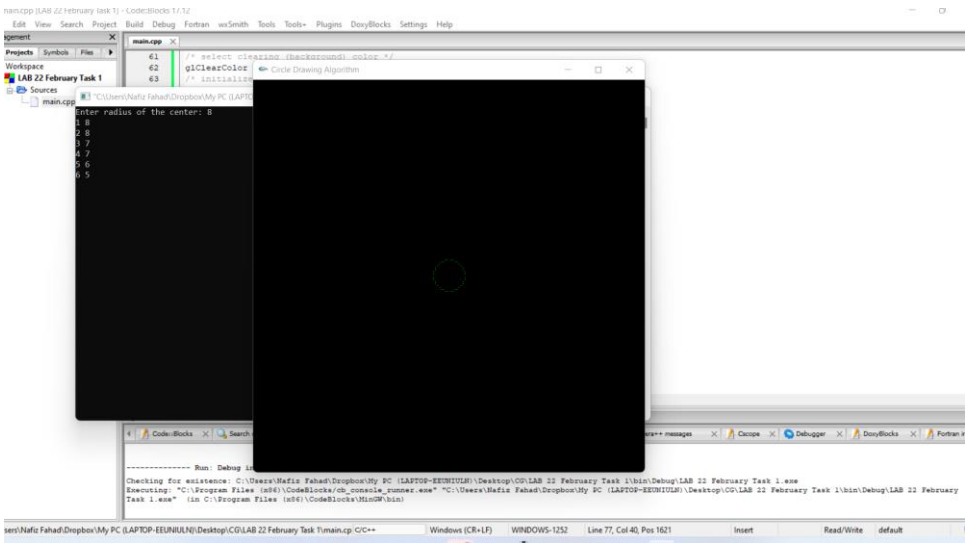
CODE TASK 1 and 2 Together

```
#include <iostream>
#include <GL/gl.h>
#include <GL/glut.h>
using namespace std;

int x = 0, y, x2, y2, p, r;

void display(void)
{
/* clear all pixels */
glClear (GL_COLOR_BUFFER_BIT);

glColor3f (0.0, 1.0, 0.0);

glBegin(GL_POINTS);
p = 1-r;

while(x<y){
   if(p<0){
      x = x+1;
      y=y;
      cout << x <<" " << y << endl;
      p = p+(2*x)+1;
   }
   else {
      x = x+1;
      y = y-1;
      cout << x <<" " << y << endl;
      p = p+(2*x)+1-(2*y);
```

```
    }


    /*glVertex3i(x,y,0.0);
    glVertex3i(y,x,0.0);

    glVertex3i(-x,y,0.0);
    glVertex3i(-y,x,0.0);

    glVertex3i(-x,-y,0.0);
    glVertex3i(-y,-x,0.0);

    glVertex3i(x,-y,0.0);
    glVertex3i(y,-x,0.0);*/

    glVertex2i(x,y);
    glVertex2i(y,x);

    glVertex2i(-x,y);
    glVertex2i(-y,x);

    glVertex2i(-x,-y);
    glVertex2i(-y,-x);

    glVertex2i(x,-y);
    glVertex2i(y,-x);
}

glEnd();
glFlush ();
}
void init (void)
{
```

```
/* select clearing (background) color */
glClearColor (0.0, 0.0, 0.0, 0.0);
/* initialize viewing values */
glMatrixMode(GL_PROJECTION);
glLoadIdentity();// replaces current matrix with identity matrix
glOrtho(-100.0, 100.0, -100.0, 100.0, -10.0, 10.0); // defines the size of graph paper
}
int main(int argc, char** argv)
{
   cout << "Enter radius of the center: ";
   cin >> r;
   y=r;
glutInit(&argc, argv);
glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
glutInitWindowSize (600, 600);
glutInitWindowPosition (100, 100);
glutCreateWindow ("Circle Drawing Algorithm");
init ();
glutDisplayFunc(display);
glutMainLoop();
return 0; /* ISO C requires main to return int. */
}
```

OUTPUT TASK 1



OUTPUT TASK 2