

Name:Harshada Gopal Rayate

Roll No:19 SEDA

Subject :CGAVR

Assignment No : 01

Bresenham Line Algorithm

```
#include <stdio.h>
#include <GL/glut.h>
#include <math.h>
```

```
// Function to plot a pixel
void setPixel(int x, int y) {
    glBegin(GL_POINTS);
    glVertex2i(x, y);
    glEnd();
}
```

```
// Bresenham's Line Drawing Algorithm
void bresenham(int x1, int y1, int x2, int y2) {
    int dx = x2 - x1;
    int dy = y2 - y1;
    int sx = (dx > 0) ? 1 : (dx < 0) ? -1 : 0; // Step in x direction
    int sy = (dy > 0) ? 1 : (dy < 0) ? -1 : 0; // Step in y direction
    dx = abs(dx);
    dy = abs(dy);
```

```
    if (dx > dy) {
        int err = dx / 2;
        while (x1 != x2) {
            setPixel(x1, y1);
            err -= dy;
            if (err < 0) {
                y1 += sy;
                err += dx;
            }
            x1 += sx;
        }
    } else {
        int err = dy / 2;
        while (y1 != y2) {
            setPixel(x1, y1);
            err -= dx;
            if (err < 0) {
                x1 += sx;
                err += dy;
            }
            y1 += sy;
        }
    }
```

```
}  
}  
setPixel(x2, y2); // Ensure the last point is plotted  
}
```

```
// Display function to clear and draw the line  
void display() {  
glClear(GL_COLOR_BUFFER_BIT); // Clear the screen
```

```
bresenham(50, 50, 300, 200);
```

```
glFlush();  
}
```

```
// Initialization function  
void init() {  
glClearColor(0.0, 0.0, 0.0, 0.0); // Set background color to black  
glColor3f(1.0, 1.0, 1.0); // Set drawing color to white  
gluOrtho2D(0, 640, 0, 480); // Set the coordinate system  
}
```

```
int main(int argc, char** argv) {  
glutInit(&argc, argv); // Initialize GLUT  
glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB); // Set display mode  
glutInitWindowSize(640, 480); // Set window size  
glutInitWindowPosition(100, 100); // Set window position  
glutCreateWindow("Bresenham's Line Drawing"); // Create window with title
```

```
init(); // Call initialization function  
glutDisplayFunc(display); // Register display function
```

```
glutMainLoop(); // Enter the GLUT event loop  
return 0;  
}
```

