Computer Graphics - Assignment 1

CS 4830 — Dr. Mihail
Department of Mathematics Computer Science
Valdosta State University

January 25, 2019

1 Introduction

In this assignment you will implement Bresenham's line drawing algorithm. You will use the skeleton program given to you. The functionality implemented handles the user drawing a line by first clicking on a point in the canvas, then by moving the mouse to a second point and releasing the button triggers an event that calls the Bresenham line drawing algorithm.

The first point of the line is where the user clicked, the second point is set to where the user released the mouse. The drawing (setting a pixel) is done on the HTML canvas. Your implementation has to handle all possible situations, including all octants and vertical and horizontal lines.

2 Implementation Details

2.1 Mathematics

The algorithm can be implemented in the first octant. For any other octant, your program should first identify it, then transform the problem to the first octant. Such a mapping is given in Section 3.3 of the following Wikipedia article: `https://en.wikipedia.org/wiki/Bresenham%27s_line_algorithm`

In the first octant, the main idea of the algorithm is simple: as x increases by 1, making the decision whether y increases is done via accumulating an error. The error can be thought of as a deviation (along the y axis) from the true line. If the error is strictly positive, then y increases, otherwise remains unchanged. The derivation of this algorithm is shown in the course slides.

A line can be written in slope-intercept form as follows:

$$y = f(x) = mx + b \tag{1}$$

where m is the slope, or rise over run: $\frac{\Delta y}{\Delta x}$. After some simple algebraic manipulation, one can write the equation of the line as follows:

$$f(x, y) = 0 = (\Delta y)x - (\Delta x)y + (\Delta x)b \tag{2}$$

or

$$f(x, y) = 0 = Ax + By + C \tag{3}$$

where $A = \Delta y$, $B = -\Delta x$ and $C = (\Delta x)b$ are constants.

Given a line from P_0 to P_1, $P_0 = \{x_0, y_0\}$ and $P_1 = \{x_1, y_1\}$, the starting is on the line, i.e., $f(x_0, y_0) = 0$. Given a slope ≤ 1, the idea is to evaluate which point should be chosen at the next step $(x_0 + 1, y_0)$

or (x_0+1, y_0+1). This is done by looking at the midpoint, or $D = f(x_0+1, y_0+\frac{1}{2})$. If D is positive, then the ideal line is below the midpoint, and closer to (x_0+1, y_0+1).

We can also evaluate f at the midpoint using differences between points:

$$D = f(x_0+1, y_0+\frac{1}{2}) - f(x_0, y_0) \tag{4}$$

after simplification $D = A + \frac{1}{2}B$, or $D = \Delta y - \frac{1}{2}\Delta x$. Due to the fact that we only care about the sign of D in making the decision to increase y, we can algebraically manipulate equations (multiplying by 2) to only work with integers.

Initially, the error is $D = 2\Delta y - \Delta x$. A point is plotted at P_0, then depending on the sign of D, y either increases or stays the same.

3 Program Requirements

- You will ONLY modify one function, bresenham(p0, p1), and uncomment the call to i from display().
- You have to correctly determine octant and output it in the HTML paragraph.
- The algorithm cannot use any floating point arithmetic for line drawing.
- The algorithm has to work in any situation (e.g., horizontal, vertical, 0 length lines).

4 Hints

- Please note the coordinate system in HTML's canvas has the positive Y-axis pointing down.

5 Due Date

This assignment is due before midnight on Sunday, February 3rd.

6 Grading Rubric

- 1st octant works **50%**
- All octants work **50%**