Programming Assignment #5

CS 200, Fall 2015

Due Thursday, October 8

Assignment requirements

Implement the Midpoint line drawing algorithm. I will provide you with the header file `DrawLineMidpoint.h`, which declares the function prototype:

```
void DrawLineMidpoint(Raster &r, int x0, int y0, int x1, int y1);
```

(the header file `Raster.h` has been included). You are to implement the function `DrawLineMidpoint`, which draws the line segment between the points (x_0, y_0) and (x_1, y_1) to the frame buffer defined by the specifed instance of the `Raster` class. Here are some requirements for the assignment.

- You must implement the midpoint line drawing algorithm.
- Only the header file `Raster.h` may be included. In particular, you may **not** use the `cmath` or `algorithm` header files.
- Floating point variables and operations may **not** be used. Only integer variables and integer arithmetic may be used.
- No explicit multiplications may be used. The only arithmetic operations allowed are addition, subtraction, incrementation, and decrementation. Bit shifting may also be used.
- No implicit multiplications may be used inside of any loop. I.e., you may only call the `GotoPoint` function outside of a loop.

Note that part of your grade will be based on the efficiency of your coding. As a general strategy, you should (1) strive to avoid redundant computations by precomputing values when possible, and (2) minimize the number of operations needed to perform a given task.

Your submission for this assignment should consist of a single source file, which should be named `DrawLineMidpoint.cpp`. You man not include any header files other than `Raster.h`.

Implementation table (not to turn in)

To aid in the implementation, you will need to fill out the table on the last page. Depending on the locations of the starting point (x_0, y_0) and ending point (x_1, y_1) of the line segment, the midpoint algorithm takes a different form. Define

$$\Delta x \doteq x_1 - x_0, \quad \Delta y \doteq y_1 - y_0, \quad m \doteq \frac{\Delta y}{\Delta x}$$

and

$$F(x, y) \doteq 2[y\Delta x - x\Delta y - (y_0\Delta x - x_0\Delta y)]$$

The form of the algorithm depends on 3 factors: (1) the sign of Δx, (2) the sign of Δy, and (3) whether $|m| \leq 1$ or $|m| > 1$, for a total of $2^3 = 8$ cases to consider. In each case, we need to identify:

- the two choices for next pixel to be set: (u_a, v_a) or (u_b, v_b)
- the signs of $F(u_a, v_a)$ and $F(u_b, v_b)$
- the value $D \doteq F(u_m, v_m)$, where (u_m, v_m) is the midpoint between the two pixel choices
- which pixel to choose if D is positive (the other will be chosen if D is negative).

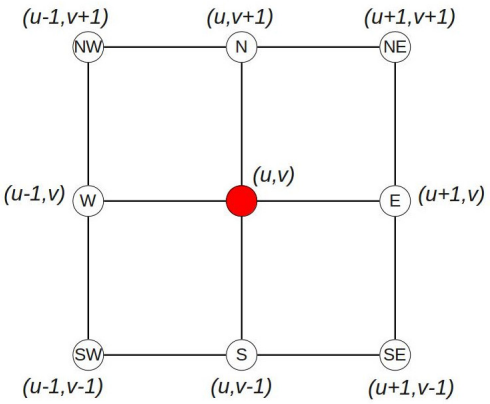
Note that if (u, v) is the last pixel that was set by the algorithm, then (u_a, v_a) and (u_b, v_b) are of the form $(u \pm 1, v \pm 1)$. However, to compute the successive values of D using the DDA idea (which allows us to avoid any multiplications), we need to know

- the initial value D_1 of D
- the amount ΔD_+ to increment D by if the previous value of D was positive
- the amount ΔD_- to increment D by if the previous value of D was negative

In the table on the last page, the two pixel choices (u_a, v_a) and (u_b, v_b) are denoted using the usual map direction notation:

$$\begin{aligned} \text{N} &= (u, v+1) \\ \text{NE} &= (u+1, v+1) \\ &\text{etc.} \end{aligned}$$

The $+$ or $-$ sign next to the direction label indicates whether the value of F at the corresponding pixel is positive or negative in sign.



A remark on ambiguous pixels

If the value of the decision variable D is zero, the midpoint falls on the line segment, so that either pixel choice is valid. You are free to choose which one. However, you should be consistent: the scan conversion of the line segment from (x_0, y_0) to (x_1, y_1) should set the same pixels as the line segment from (x_1, y_1) to (x_0, y_0).

Δx	Δy	$\lvert m\rvert \le 1$	(u_a, v_a)	(u_b, v_b)	$D > 0$	D_1	ΔD_+	ΔD_-
$+$	$+$	T	E $(-)$	NE $(+)$	E	$\Delta x - 2\Delta y$	$-2\Delta y$	$2\Delta x - 2\Delta y$
$+$	$+$	F						
$+$	$-$	T						
$+$	$-$	F						
$-$	$+$	T						
$-$	$+$	F						
$-$	$-$	T						
$-$	$-$	F						