Ryan Scheel

Project 2

This project was a challenge. I made some poor assumptions at first which significantly complicated my life. I thought I needed to be handle as many threads as there were cells, which turns out is much more complicated than what I had to do in the end, which was just handle as many threads as there are rows.

I learned an important lesson that I should ask more clarifiying questions instead of assuming requirements though. Unfortunately I’m now turning this in very late as a result. I learned a lot about threads though with help from Andy, and had some interesting bugs related to buffer overflow and fgets that Tim helped me debug.

Design Description:

My design is fairly straightforward. I break up the grid by rows, based on the number of threads. If there are 32 threads and 32 rows then each thread does one row. If there are an odd number of threads then the last thread takes on a few extra rows of cells to calculate.

There are no mutexes protecting the grids as each thread only reads from one grid and writes to it’s own place on the next grid, so they don’t interfere with each other.

There are mutexes to keep track of how many cells are calculated, and to determine if we have reached the end of the generations. These help keep the threads from getting too far ahead in generations, as only one generation can be done at a time.

Speedup:

For speedup I used the input_6000.txt file, as speedup is more noticeable with the larger inputs. With smaller inputs the benefits of multithreading is not as noticed with the added printing.
My laptop is a dual core with hyper threading so I chose to compare using 1 worker thread to 2 worker threads to 4 worker threads to 8 worker threads.

# of workers	# of generations	gridsize	time to calc
1	50	6000	66.2 seconds
2	50	6000	40.3 seconds
4	50	6000	30.9 seconds
8	50	6000	31.186 seconds

As you can see there is a significant speedup between 1 worker thread and 2. There is a less but noticeable improvement between 2 and 4. Between 4 and 8 there is actually a very slight slow down, most likely due to my laptop not being able to natively support 8 threads.