CS311 Project 4

Emily Dunham

1 a design for your system

I began the project by researching ways to parallelize prime number finding, because I did not clearly understand the mathematics of how to implement the Sieve of Eratosthenes in a way that would work in parallel at all. Over the course of this research I stumbled across the article at http://create.stephan-brumme.com/eratosthenes/, which provided a working example of a block-wise sieve. The block-wise sieving function basically just finds all primes between its two input parameters, which means that it would parallelize extremely well.

The author of the example code also claimed to have experienced significant speed improvements even in a serial sieve by hard-coding a part which skips all multiples of primes 3 through 15. To determine whether those speed improvements actually matter, I plan to include them in my code but use preprocessor commands to make it so I can compile using none, some, or all of them. By including or excluding the code at compile time rather than run time, I'll avoid slowing it down by checking unnecessary conditionals when running none or only some of the attempted optimizations.

The project requirements specify that I need to use a bitmap, but don't appear to care about the dimensions or other details that I choose for it. For simplicity, I think I'd like to generate each row of the bitmap with a separate process, then just vary the dimensions (padding the final row with 0s if necessary) based on the number of concurrent processes or threads I'm testing with. Thus, all that the main function will have to do will be parse the command line arguments,

2 places your implementation deviated from this design

3 a work log, detailing what you did when

4 any challenges you overcame in completing this assignment

5 answers to the following questions:

5.1 what do you think the main point of this assignment is?

5.2 how did you ensure your solution was correct?

5.3 what did you learn?