

Concurrency: Multi-core Programming and Data Processing

ASSIGNMENT 01

Professor: Pascal Felber
Assistent: Isabelly Rocha

March 5, 2020

Question 1: Prime numbers

The goal of this exercise is to identify all prime numbers between 1 and n^{10} (where n is the number of cores in your computer). Implement two variations of a program which distributes work between n threads. The first variant should have each thread i testing numbers in the range $[(i - 1) * n^9 + 1, i * n^9]$. The second variant should have a shared counter used by threads to pick the next number to test. Ideally, each thread should take approximately the same time processing its n^9 numbers.

1. Run a program with 1 thread that identifies all prime numbers between 1 and n^{10} and measure how long the program takes to execute.
2. Run the first and second variants described above with n threads and measure how long they take to execute.
3. Report the runtime in milliseconds for both variations, the speedup achieved by each implementation and your machine specification.

Question 2: Producer-Consumer Problem

Suppose we have two types of threads: Producers and Consumers, sharing a circular buffer. Each Producer deposits data at a suitable position in the buffer, denoted by a variable in (i.e, the next position available in the buffer) and advances the variable in , while each Consumer retrieves the data item at the position denoted by the variable out and advances this variable. A producer cannot deposit its data if the buffer is full. Similarly, a consumer cannot retrieve any data if the buffer is empty. Write a program that can correctly coordinate the producers and consumers and their depositing and retrieving activities. For simplicity, the number of Producer threads is equal to the number of Consumer threads which is denoted by t . The buffer will have n integer elements. Variables t and n are program arguments.