Digital Systems

EEE4084F

Practical 3 – Dynamic pThreads

[30 Marks]

Introduction

The focus of this practical is dynamic partitioning of data for multi-threaded applications. It continues from Practical 2.

The Algorithm

Keep a shared-memory data structure to keep track of the block number. In the simplest case, each block consists of one row.

Each thread takes possession of one block at a time, updating the global block number in the process. When it is finished processing that block, it takes possession of the next available block, and so forth until there are no more blocks left.

The block number is a shared resource, and must therefore be protected by means of mutual exclusion.

Part 1: Experiments

Make use of your code from Practical 2. Experiment with different thread counts and different block sizes. Comment on the time performance of each case and compare your results with those obtained using the static partitioning methods of practical 2.

Part 2: Report

Compile your experiments and findings into an IEEE-style conference paper. The page limit is 3 pages.

This case, where you expand on previous work, is typical in academia. Make reference to your report of Practical 2 (and put it in your list of references), but don't repeat yourself unnecessarily. It should work out that this report is shorter, because most of what you need to say has already been said in the previous report. When you compare results from the static case, make sure you reference your previous report (i.e. avoid self-plagiarism).

Submit your paper to the Vula Assignment for this practical.