**CPSC 3300**

**Project 3 -- Speed up Matrix Multiplication**

Due date: 11:59PM, April 24

Submit your project to canvas

Grading Weights: 10% title and author info

10% abstract

10% related work

30% methodology

30% experimental results

10% bibliographic references and citation style

This is an individual or team of two project assignment.

An implementation that gives an incorrect product matrix gets 0 points for methodology on this project.

**Summary:** In this project you are going to optimize an implementation of sequential Matrix Multiply provided in matmul\_naive.c. The goal is to obtain a maximum overall speedup as you can with multiple optimization techniques. Note that you can’t use parallel computing for this project.

Start with the provided naïve implementation, and add optimization techniques one by one to hand tune the program and make it run faster. Common optimization techniques include register variable, loop interchanging, loop unrolling, blocking, prefetching. These techniques are commonly used by programmers and compilers.

Name the optimized code matmul\_opt.c, compare the optimized code against the naïve implementation.

Submit matmul\_opt.c and a report named “Matrix Multiplication Optimization.pdf” both to Canvas. The required sections and contents for the report is provided at the end of this document.

**Matmul:** C = A x B where A is a general m x k matrix, B is a general k x n matrix, and C is the product m x n matrix. For simplicity we assume m = k = n or square matrices in this project.

**Compare hand-tuned program against naïve program.** Always use -O0 (-[Big O][zero]) to turn off optimization in compilers when compiling the hand-tuned program and the naïve program. Compare the execution times of these two programs for various matrix sizes and calculate speedups, which is defined as Timenaive/Timeopt.

**Timing:** Use timing functions in the code to measure the performance. Example timing functions include gettimeofday and getrusage in Unix/Linux environments. Report the performances, speedups, and comparisons in figures or/and tables. Multiple matrix sizes should be examined in the experiments.

The timing and comparisons can be reported in a table as follows or in figures. In the table, T1 and T2 represent two different optimization techniques. Your method should include as many effective techniques as possible, so that the achieved speedup is the highest. Particularly, your method should include the blocking technique and use different block sizes. Assume that blocking is denoted as T4, the following table includes the timing info for two block sizes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Technique / Matrix Size** | **1000** | **2000** | **3000** | **4000** | **5000** |
| **Naïve + (-O0) (**Timenaive**)** |  |  |  |  |  |
| **Transformed code with T1 +**  **(-O0) (**Timeopt**)** |  |  |  |  |  |
| Timenaive/Timeopt |  |  |  |  |  |
| **Transformed code with T1 & T2 + (-O0) (**Timeopt**)** |  |  |  |  |  |
| Timenaive/Timeopt |  |  |  |  |  |
| **…** |  |  |  |  |  |
| **Transformed code with T1 & T2 & T3 & T4 (blk=32) + (-O0) (**Timeopt**)** |  |  |  |  |  |
| Timenaive/Timeopt |  |  |  |  |  |
| **Transformed code with T1 & T2 & T3 & T4 (blk=48) + (-O0) (**Timeopt**)** |  |  |  |  |  |
| Timenaive/Timeopt |  |  |  |  |  |

The report should be at least 4 pages (2 ½ pages single spaced) or if working as a team of two, 8 pages (5 pages single spaced).

**Required sections of the report:**

1. *Title and author info:* Your name should appear towards the top of the first page of your report. I suggest 14-point font for title, and 12-­point font for the rest of your document. If you work in teams of two, include both names on the on the paper and both people should submit the same paper to Canvas. The document should use a single column with 12 point font. Margins should be 1 inch from all sides. Use numeric headings and subheadings as necessary.
2. *Abstract:* On the first page of your document, provide an abstract that summarizes your work, approach, and findings. This should include exact numbers for your achieved speedup. This should provide the reader with an overview of your contribution without having to read the paper. This should be no longer than 200 words.
3. *Related Work:* If you utilized algorithms or approaches that you did not invent, you should cite them accordingly in your paper. This is to be your own work, where you apply optimizations by hand to the code given by the instructor. You may use other algorithms and techniques provided you understand and can explain how they work, and you implement the code yourself. Here you should simply identify others’ work and discuss its relationship to your optimized matrix multiply. This section should be less than 300 words. Save the particulars of your approach for the next section.
4. *Methodology:* This section should describe exactly the method you used to optimize the application. You should provide a step-by-step accounting of your approach. Why you did what you did, and how it might be different or the same from previous approaches. You should discuss your methods for attempting to speed up the given code and how you believe they will affect the code generally. Save exact numbers for the next section.
5. *Results:* Here you should provide the empirical timing measurements for the matrix multiply you implemented verse the base version(s). You should summarize the details of the system upon which implementation took place (architecture, platform) and the timing methods you used (and why you chose both arch and timing method). You should provide charts and tables as necessary to show your results and discuss the results in the context of your methodology from the last section. This section should contain mostly tables or graphs (like those found at the end of this document) and some analysis of what worked, why you think it worked, and what didn’t work.
6. *References:* The IEEE Journal citation format is strongly recommended. You must cite all references. Plagiarism is unacceptable. Formatting details can be found at:

http://www.ece.uiuc.edu/pubs/ref\_guides/ieee.html