CS2.201: Computer Systems Organization
Professor: Dr. Deepak Gangadharan
Spring 2021
International Institute of Information Technology, Hyderabad

Assignment 2

Deadline: 23:55, 25/04/2021

Welcome to Assignment 2 of the Computer Systems Organisation Course. The aim of this assignment is to familiarize you with Cache Optimization.

Instructions: Read all the instructions below carefully before you start working on the assignment.

- This assignment has 3 tasks.
- You can evaluate the performance of your code using tools like perf, valgrind, cachegrind, etc.
- For task 2 and 3, you have to optimize the basic algorithm provided.
- Compare the performance statistics after each optimization and add it to final report.
- Printing of output will not graded in evaluations. It is not necessary to print the output.
- Suggestion 1: Edit the given code to take input from input generator.
- Suggestion 2: Use input size close to your system's cache size. This might help you see significant changes in performance report.
- Report should contain performance results and theoretical part of each task. It should also contain all the optimizations you did and the positive/negative effect they had on the performance.
- Deadline is final. No extensions.
- Total marks for assignment is 60.

Submission format: Strictly adhere to the following submission format. Failure to do so may result in an erroneous evaluation of your assignment.

- Zip all your codes and pdf of report in a folder named <*roll_number*> and name the zipped folder as <*roll_number*>_assign2.zip
- Folder should contain report_<*roll_number*>.pdf, 2.c, and 3.c

Task 0: 10 marks

System Details

Find and provide detailed information about your system. It should contain information regarding your system's operating system, kernel modules, file systems, processor, memory, PCI devices, USB devices, battery, sensors, storage, DMI, and benchmark score of at least one of CPU Blowfish, CPU cryptohash, CPU Fibonacci, CPU N-Queens, CPU Zlib, FPU FFT, FPU Raytracing and GPU Drawing.

Task 1: 25 marks

Matrix Multiplication

Given two matrices, A (mxn) and B (nxp), write a cache optimized code to multiply them. In addition, provide the reason for the improvement in performance.

1 $\leq$ m,n,p $\leq$ 3500

-600 $\leq$ Elements of A and B $\leq$ 600

Input/Output Format

- INPUT: m, n, p, A and B.
- OUTPUT: Print result of A*B or store it in a text file.

Task 2: 25 marks

Merge Sort

Given a basic C program for merge sort algorithm, optimize it for better cache usage. In addition, provide the reason for the improvement in performance.

Merge sort accesses data sequentially and the need of random access is low. However, it is slower comparative to the other sort algorithms for smaller tasks. Merge sort algorithm requires additional memory space of O(n) for the temporary array and it goes through the whole process even if the array is sorted.

Input/Output Format

- INPUT : The first line of the test case contains integers n (1 $\leq$ n $\leq$ 6*10^6) — the number of elements to be sorted. The second line of the test case contains n integers a_1,a_2,...,a_n (Size of a_i should be int), where a_i is the i^{th} element of the set.
- OUTPUT: Print the sorted array or store it in a text file.

Sample Test Case

Input :

5

3 4 6 2 5

Output:

2 3 4 5 6