

CSE 420 Fall 2016 Project 5

Due: 11:59 pm, Wednesday, April 27, 2016

Vectorization

Objective: Studying the advantages of vectorizing a program and various SIMD instructions used in the vector programming. [Please read the deliverables section before Submission Instructions.](#)

Problem 1 [CPU info] (20 points)

Identify whether your processor supports SIMD instructions. You can do this by checking the processor manual online if you know your processor model. If you are using the cluster type the following command

```
$cat /proc/cpuinfo
```

You can search the flag fields to see the SIMD support.

Question 1 (20 points)

Specify details about the processor model name, cache size and number of cores. Study closely about your processor and specify whether it supports SIMD and also specify what are the SIMD extensions supported.

Problem 2 [Vectorization and Timing Analysis](40 points)

[Note:](#) Follow the instructions to write a C program.

Step 1: Define three global arrays of size 512. Define two functions,

vec_add(int n) : (for i=0 to n ; i++) multiply a vector with 100, add it to the 2nd vector and store it in the 3rd vector.

vec_mult(int n) : (for i=0 to n ; i++) multiply 2 vectors and store the value in the 3rd vector.

Step 2: Write a main function calling the two function in a for loop (x = 0 to <=512; x+=8). **Pass the x value as arguments to the functions.**

Print out time taken to run each iteration of the function, this can be done by inserting start time and end time calls before each function and printing difference of end time and start time.

Note: You can use fopen and fclose commands to save the time for each iteration for function1 and function2 in two separate files. Save you C file as Project5.c. For timing analysis you can use wall clock time like *clock_gettime* or write a function or routine to printout the time.

Step 3: Compile and run the c file in SISD. Use following commands

```
$gcc Project5.c -o Project5
$ ./Project5
```

Step 4: Now vectorize the c program. O3 flag offers vectorization of the loops. Refer <https://gcc.gnu.org/onlinedocs/gcc/Optimize-Options.html>

```
$gcc -O3 Project5.c -o Project5_simd
$ ./Project5_simd
```

Question 1 (20 points)

Present your execution table. The table should be in the following format

	Vec_add		Vec_mult	
array size (n)	TIME-SISD	TIME-SIMD	TIME-SISD	TIME-SIMD

Question 2 (20 points)

Calculate the speedup of SIMD over SISD.

$$Speedup = \frac{Exec.time\ of\ SISD}{Exec.time\ of\ SIMD}$$

Include Speedup in the above table and plot speedup with respect to the array size(x axis) for both the functions. Briefly explain the optimization flags that -O3 uses to vectorize the loops by referring to the link in Step4.

Problem 3 [Understanding SIMD] (40 points)

Run the following commands to get the binary files.

```
$objdump -ds Project5.o > sisd.dis  
$objdump -ds Project5_simd.o > simd.dis
```

Compare the two .dis files. Search for the function names in both the .dis files and briefly explain the changes in the code. Justify the numbers in the table and the speedup graph obtained from Problem 2 with the explanation of the .dis code differences. Provide comparisons for both Vec_add and Vec_mult functions in your report.

NOTE: Refer SIMD lectures to get clear knowledge about the registers, vectorization and architectures.

Project Deliverables:

Your Project5_Anonymous.zip file should contain the following

- Report document
- Excel sheet (if the table is too big to include in the report)
- Project5.c
- Project5
- Project5_simd
- sisd.dis
- simd.dis

Submission Instructions

There should be **no name or student number** of the team members in any part of the file or program of the project submission (except the work.docx work distribution file). If a team violates this policy, the score for the project will be **ZERO**.

1) Write your answers to the questions, and pay attention to the required details as indicated in the “Output” for each question. You may use additional graphs and/or tables to explain your answers.

2) Zip all files into one file and submit it on the blackboard. Name the zip file with the Project3-Anonymous.

3) Submit only one project zip per team. Submit this zip file in **Submission\Evaluation section in the project directory**.

4) Submit a work.docx file, in which you write down the names of the people in the team, and their percentage contribution in the project. Grades will be divided among students, based on the percentage you quote. Include respective ASU IDs. **Submit this in the Work Distribution section in the project directory**. If a team does not submit the work file, the score for the project will be consider **ZERO**.

5) Project grading: 80 points for the project and 20 points for project evaluation (10 points for justification of your grade and 10 points for feeds to the authors of the project)