ETH zürich

**High-Performance Computing Lab for CSE** **2020**

Student: Sina Klampt Discussed with: Alain Hügli, Anna Hutter

## Solution for Project 1

Due date: 09.03.2020 (midnight)

**HPC Lab for CSE 2020 — Submission Instructions**
**(Please, notice that following instructions are mandatory: submissions that don't comply with, won't be considered)**

- Assignments must be submitted to Moodle (i.e. in electronic format).
- Provide both executable package and sources (e.g. C/C++ files, Matlab). If you are using libraries, please add them in the file. Sources must be organized in directories called:
  *Project_number_lastname_firstname*
  and the file must be called:
  *project_number_lastname_firstname.zip*
  *project_number_lastname_firstname.pdf*
- The TAs will grade your project by reviewing your project write-up, and looking at the implementation you attempted, and benchmarking your code's performance.
- You are allowed to discuss all questions with anyone you like; however: (i) your submission must list anyone you discussed problems with and (ii) you must write up your submission independently.

In this project you will practice memory access optimization, performance-oriented programming, and OpenMP parallelizaton on Euler.

# 1. Explaining Memory Hierarchies *(30 Points)*

## 1.1. Identifying Parameters

Using the commands given on the exercise sheet, I was able to find the following values for the memory hierarchy on the computer node of the Euler cluster:

| | |
|---|---|
| Main memory | 32 GB |
| L3 cache | 6 MB |
| L2 cache | 256 kB |
| L1 cache | 32 kB |

## 1.2. Running Membench

In this exercise we were supposed to run the membench program on our local machine and on the Euler cluster.

The first plot (Fig. 1) represents the results on my local machine. I have a Intel Core i7 processor with 16 GB of Main Memory, 196 kB of L1 cache, 2 MB of L2 cache and 8 MB of L3 cache.

The second plot (Fig. 2) represents the results using Euler. I have used the Xeon Gold 6150 CPU. When we look at the time axis, we can easily see that the performance of the Euler cluster is much better than on my local machine.

Figure 1: Plot for local machine

Figure 2: Plot for Euler

### 1.3. Memory Access Patterns

In the first case we had: csize = 128 and stride = 1. Since one integer corresponds to 4 Bytes, we have 128 times 4 which equals 512. Thus we need to look at the red line corresponding to 0.5 kB.

In the second case we had: csize = $2^{20}$ and stride = csize/2. This time we get csize = $2^{22}$ Byte (4 MB) and stride = $2^{21}$ Byte (2 MB). Thus we have to look at the light blue line.

### 1.4. Temporal Locality

asdf

## 2. Optimize Square Matrix-Matrix Multiplication *(70 Points)*

### 2.1. Optimizing Part One

### 2.2. Optimizing Part Two