

Andreas Scholl (andreasscholl@stud.uni-frankfurt.de)

Kyrill Fries (kyrill.fries@stud.uni-frankfurt.de)

---

## High Performance Computing ( 2019/2020 )

### Exercise 1

---

#### Task 1 (7 points)

Wikipedia<sup>1</sup> defines a supercomputer as "a computer with a high level of performance compared to a general-purpose computer." But what criteria defines a high level of performance?

- a) Find out how many FLOPS the CDC 6600, one of the first supercomputers from 1964, was able to perform.
- b) What is the name of the fastest supercomputer in the world right now and how many FLOPS does it calculate theoretically and practically?
- c) Which benchmarking tool is used to determine the practical performance of the supercomputer from the previous task?
- d) Which position does the fastest german supercomputer hold on the Top500 list?
- e) Imagine you had to create your own ranking. Which criteria, other than operations per second, would you measure and why?

#### Task 2 (5 points)

The peak performance of a computer system is usually measured in operations or instructions per second. An example architecture for this is the BGCA (**B**est **G**raphics **C**ard **A**vailable) which is a GPGPU. Furthermore today these cards today are used for general purpose computation because of their programmability features. Assume that the BGCA consists of 512 parallel stream processors, which again consist of 8 ALUs (**A**rithmetic **L**ogik **U**nit). Every ALU is able to perform 3 floating point operations (single precision) per clock cycle (pipelined). The entire processor is clocked at 1 GHz.

- a) Write down, in your own words, what the peak performance of a system is. (1-2 sentences)
- b) Calculate the peak performance of the BGCA. Please write down the whole calculation.
- c) Give a possible reason why an application is not able to exploit the peak performance on such an architecture.

#### Task 3 (7 points)

Assume that you have developed a piece of software which solves a mathematical problem on a single processor in 72 ms. Your program achieves this by initially calculating some variables (not parallelizable) before it starts solving the problem set (parallelizable).

- a) You assume that the instantiation takes 20% of the programs runtime and that the solving of the problem set is perfectly parallelizable. What is the suspected speedup achieved when you run your program on 32 processors?

---

<sup>1</sup><https://en.wikipedia.org/wiki/Supercomputer>

- b) Since you don't like to assume things, you decided to test your program on 4 processors. This time the program needed 32 ms to finish. Calculate how long (in ms) each processor works on solving the problem set without instantiation and how many percent of the total runtime the instantiation really needs.
- c) Your boss found out about your little program and sold it to a company which requires the mathematical problem to be solved in 15 ms. Being a marketing genius your boss told them that this was no problem since you just need to add more processors until its fast enough. Explain to your boss if this requirement can be achieved, if at all, and what the maximum speedup and minimal time required is.
- d) To improve your original algorithm you make 50% of the instantiation phase perfectly parallelizable but at the same time limiting the system to a maximum number of 12 processors. Will this solution be enough to save the company?

## Homework

### Homework deliverables:

1. Homework sheet created using the provided  $\text{\LaTeX}$  template. [PDF format]
2. Working source codes of the homework, including all program, `.tex` and image files. [program and  $\text{\LaTeX}$  sources]
3. Cite your sources.

Task	Maximum points
Task 1	7
Task 2	5
Task 3	7

**Deadline: 30.10.2018 14:00**