Exercise sheet 3 – Hardware performance

Goals:

- Basic knowledge about performance measures
- Practical benchmarking: Popular benchmarking suites and tools (CPU, (GPU), Memory, IO, Network)

Exercise 3.1: Clock cycle time

(a) Calculate the clock cycle time `t_CC (s)` of the Intel i9-11980HK processor at its base frequency (configurable TDP-up base frequency). You may find the CPU specification somewhere in the internet.

Proposal for solution: According to `https://www.intel.de/content/www/de/de/products/sku/213800/intel-core-i911980hk-processor-24m-cache-up-to-5-00-ghz/specifications.html`, the performance-core base frequency is at 3.30 GHz.

$$\begin{aligned}\texttt{t\_CC} = \frac{1}{\texttt{CR}} = \frac{1}{3.30\ \texttt{GHz}} = \frac{1}{3.30 \times 10^9} &= 303.03 \times 10^{-12}\ \texttt{s} \\ &= 303.03 \times 10^{-9}\ \texttt{milli seconds} \\ &= 303.03 \times 10^{-6}\ \texttt{micro seconds} \\ &= 303.03 \times 10^{-3}\ \texttt{nano seconds} \\ &= 303.03\ \texttt{pico seconds}\end{aligned}$$

(b) How many ADD instruction could theoretically be performed within one second, if an ADD instruction takes 0.25 cycles (according to `https://www.agner.org/optimize/#manual_instr_tab`)?

Proposal for solution:

$$\begin{aligned}\texttt{\#Add inst.} = \frac{\texttt{CR}}{0.25} = \frac{3.3 \times 10^9}{0.25} &= 13.2 \times 10^9 \\ &= 13.2\ \texttt{BE: milliard, AE: billion, DE: Milliarden}\end{aligned}$$

(c) How do the calculated numbers change, if the Intel i9-11980HK runs on its max turbo frequency?

Proposal for solution:

$$\texttt{t\_CC} = \frac{1}{\texttt{CR}} = \frac{1}{5\ \texttt{GHz}} = 200\ \texttt{pico seconds}$$

$$\texttt{\#Add inst.} = \frac{\texttt{CR}}{0.25} = \frac{5 \times 10^9}{0.25} = 20 \times 10^9$$
$$= 20\ \texttt{BE: milliard, AE: billion, DE: Milliarden}$$

Exercise 3.2: Time measurement and speed up

(a) Inspect the source in `sheet_03_performance/time_measurement/main.c`

(b) Compile the source in `sheet_03_performance/time_measurement` with: `gcc -O0 ...`

Proposal for solution:

```
gcc -O0 -o prog main.c
```

(c) Measure the time (real) when executing the compiled program.

Proposal for solution:

```
time ./prog
```
$t_{\texttt{O0}} = 1.997\text{s}$

(d) Repeat the compilation and time measurement with: `gcc -Ofast ...`

Proposal for solution:

```
gcc -Ofast -o prog main.c
time ./prog
```
$t_{\texttt{Ofast}} = 0.232\text{s}$

(e) Calculate the speedup.

Proposal for solution:

$$S = \frac{\texttt{O0}}{\texttt{Ofast}} = \frac{1.997}{0.232} \approx 8.6$$

Exercise 3.3: Popular benchmarking suites

(a) Use `geekbench5` to benchmark your CPU. You can download it from: `https://www.geekbench.com/`

Proposal for solution:

```
#start benchmark
./geekbench5
```

(b) Who has the highest single core and multi core score?

(c) Use PassMark PerformanceTest to benchmark your CPU. You may install the `libncurses5` library first. You can download it from: `https://www.passmark.com/products/pt_linux/index.php`

Proposal for solution:

```
#install required dependencies
sudo apt install libncurses5

#start benchmark
./pt_linux_x64
```

(d) Who has the highest integer and floating point MOPS?

Exercise 3.4: CPU benchmarking

(a) Use `7z` to benchmark a single core on your CPU

Proposal for solution:

```
#benchmark on 1 core
7z b -mmt1
```

(b) Use 7z to benchmark all cores on your CPU

Proposal for solution:

```
#benchmark on all cores
7z b
```

(c) Who has the highest single core and multi core MIPS value?

Exercise 3.5: Memory benchmarking

(a) Use sysbench to benchmark your memory with a single thread.

Proposal for solution:

```
#measure memory bandwidth (1 thread)
sysbench memory run
```

(b) Who has the highest single core memory throughput?

Exercise 3.6: IO benchmarking

(a) Use `fio` to benchmark the read throughput (without buffers) of your storage medium (HDD, SSD).

Proposal for solution:

```
#measure read io performance (without buffers)
fio -direct=1 -rw=read --size=1G -name=test
```

(b) Use `fio` to benchmark the write throughput (without buffers) of your storage medium (HDD, SSD).

Proposal for solution:

```
#measure read io performance (without buffers)
fio -direct=1 -rw=write --size=1G -name=test
```

(c) Who has the highest read/write IO throughput?

Exercise 3.7: Network benchmarking

(a) Use `speedtest-cli` to measure the internet speed.

Proposal for solution:

```
#benchmark internet bandwidth and ping
speedtest-cli
```

(b) What is the ping and the throughput?

(c) Optional: If you have Linux on your host: Use `wavemon` to measure the WLAN quality.

Proposal for solution:

```
#monitor wlan quality and benchmark read/write speed
wavemon
```

Exercise 3.8: Windows or macOS benchmarking

(a) Find and test similar tools for Window or macOS:

- CPU
- GPU
- Memory
- IO
- Network