Prof. G. Zachmann
Maximilian Kaluschke (mxkl@uni-bremen.de)
Thomas Hudcovic (hudo@uni-bremen.de)

University of Bremen
School of Computer Science
CGVR Group
April 3, 2024

Summer Semester 2024

Assignment on Massively Parallel Algorithms - Sheet 1

Due Date April 14, 2024

Exercise 1 (CUDA installation, *0 Credits*)

To install CUDA, you need to download the CUDA Toolkit. As of right now, only Linux and Windows are supported; Mac users are encouraged to work on the PCs in our lab. Please follow the appropriate installation guide down below. To test whether CUDA works as intended on your computer, we recommend that you run the deviceQuery and the bandwidthTest examples as instructed in the guide (They are also bundled with the canonical CUDA examples on the course page).

Windows: `https://docs.nvidia.com/cuda/cuda-installation-guide-microsoft-windows/`
Linux: `https://docs.nvidia.com/cuda/cuda-installation-guide-linux/`

Note: As our exercises are intended to be worked on by both Windows and Linux users, we use CMake as build system. Each assignment has a CMakeLists.txt file, among which you will find the line:
`set_property(TARGET ${name} PROPERTY CUDA_ARCHITECTURES 50 61 72 89)`
Depending on your Nvidia card and CUDA version, you might need to add your compatible CUDA architecture to it. You can find a list here: `https://developer.nvidia.com/cuda-gpus#compute`

For example the 89 corresponds to compute capability (cc) 8.9, which is the cc for the RTX4000 series. Also note that the newest version of the CUDA-Toolkit (12.x) requires an Nvidia GPU with cc >= 50

Exercise 2 (CUDA basics: Memory, *9 Credits*)

In the framework `cudaMallocAndMemcpy`, you'll find a .cu file of the same name.

a) Allocate two arrays `d_a` and `d_b` on the device of the same size as the array `h_a` on the host. You can use `sizeof(datatype)` to get the number of bytes for `datatype`. (*2 Credits*)

b) Copy `h_a` on the host to `d_a` on the device. (*2 Credit*)

c) Do a device to device copy from `d_a` to `d_b`. (*2 Credit*)

d) Copy `d_b` on the device back to `h_a` on the host. (*2 Credit*)

e) Free `d_a` and `d_b` on the device. (*1 Credits*)