University of Basel

Lecturer:
Prof. Dr. Florina M. Ciorba

Computer Lab. U1.001
Spiegelgasse 1,
CH-4051 Basel

Assistants:
Ahmed Eleliemy, M.Sc
Ali Mohammed, M.Sc

http://informatik.unibas.ch/fs2017/high-performance-computing/

High Performance Computing (17164-01) **Spring Semester 2017**

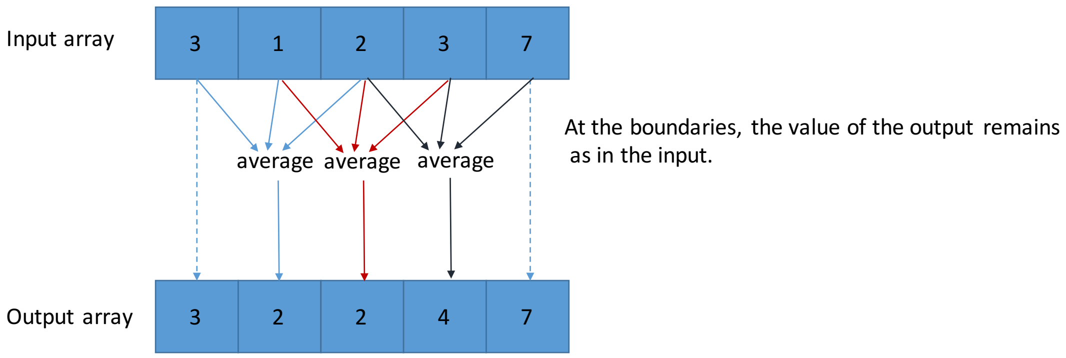
Assignment 5: Introduction to GPU programming *(20 Points)*

Starting Date: May 11, 2017
Deadline: May 30, 2017 - 23:59:59

Objectives:
1- Understand the memory transfers between the host (CPU) and the device (GPU).
2- Understand how to launch 2D kernels.
3- Exploit the massive parallelism offered by GPUs.

1 Memory transfers between the host and the device *(5 Points)*

Given the file *T1.cu* that contains a simple kernel function. The figure below describes how this kernel should work.


Input array
3
1
2
3
7
average
average
average
At the boundaries, the value of the output remains as in the input.
Output array
3
2
2
4
7

a. Identify the root-causes that make this kernel not work as described in the figure. *(2.5 Points)*

b. Propose, discuss and implement what is necessary to make this kernel work correctly *(2.5 Points)*

2 Launching 2D kernels and exploiting the massive parallelism *(15 Points)*

Given the file *T2.cu* which contains a basic skeleton for developing a matrix multiplication program on GPU, show the right way of launching the matrix multiplication kernel in 2D fashion.
Attention: The target device (GPU) on which the code will be executed has a limit of 1024 threads per block.

a. Implement the five TODO parts in the given skeleton within the *main* function *(5 Points)*

b. Implement the TODO part in the *matrix_mult_kernel* function *(5 Points)*

c. Run and obtain the total program execution time for
*Matrix size N*N = 32*32, 64*64, 128*128, 512*512, and 1024*1024* *(5 Points)*

Ensure that your optimizations do not affect the correctness of the results.
The delivered solution should be in one tar file.