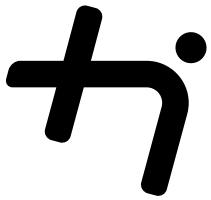
Technische Hochschule Ingolstadt
Prof. Dr.-Ing. Richard Membarth



27. März 2023

GPU Programming
Assignment 1

Submission deadline for the exercises: 27. März 2023

1.1 The GPU

a) How does occupancy relate to latency hiding? Which factors limit occupancy?
b) What is a multiprocessor? How do blocks relate to multiprocessors?

1.2 Warp Scheduling

a) `compute()` is a simple CUDA kernel with execution times given for the individual statements. Three warps (w0, w1, and w2) executing this program are launched on a multiprocessor which can execute one warp at a time.

Insert into the table below one possible execution order a hardware warp scheduler trying to hide memory latency could choose. A warp can be in one of the following states: ready, execute, suspended, or exited.

We assume that issuing of a memory request itself does not take any time; the values given in comments correspond to the time it takes until the memory request has been served. You can assume that different memory requests do not influence each other.

```
1 __global__ void compute(float* in, float* out) {
2     int tid = blockIdx.x * blockDim.x + threadIdx.x;      // 10 cycles
3     float v = 42.0f * in[tid];           // 10 cycles + 30 cycles (mem)
4     consume(tid, v);                                      // 10 cycles
5 }
```

ready	w0 w1 w2	w1 w2													
exec.		w0 (ln 2)													
susp.															
exited															
	$t = 0$	10	20	30	40	50	60	70	80	90	100	110	120	130	140