

alpaka Parallel Programming – Online Tutorial

Lecture 20 – Thread Parallelism in alpaka

Lesson 21: 1D Work Division



CASUS

CENTER FOR ADVANCED
SYSTEMS UNDERSTANDING

www.casus.science



Printing the Block index

- Open `helloWorld` in your favourite editor
- Change the Kernel as shown on the right side
- Switch to your `build` directory and rebuild:

```
cmake --build . \  
--config Release
```

- Execute the example

```
// Change these lines:  
uint32_t threadIdx = idx::getIdx<Grid, Threads>(acc)[0];  
printf("Hello, World from alpaka thread %u!\n", threadIdx);  
  
// To these lines:  
uint32_t gridBlockIdx = idx::getIdx<Grid, Blocks>(acc)[0];  
printf("Hello, World from alpaka block %u!\n", gridBlockIdx);
```

Lesson 21: 1D Work Division

Printing the Thread index

- Open `helloWorld` again
- Change the Kernel as shown below:

```
// Add these lines:  
uint32_t blockThreadId = idx::getIdx<Block, Threads>(acc)[0];  
printf("Hello, World from alpaka thread %u in block %u!\n",  
       blockThreadId, gridBlockIdx);
```

- Switch to your `build` directory and rebuild:
`cmake --build . --config Release`
- Execute the example again

Computing the index

- Some algorithms may need to (re-)compute the indices
- Example: Computing the `gridThreadId`

```
uint32_t gridBlockIdx = idx::getIdx<Grid, Blocks>(acc)[0];  
uint32_t blockThreadExtent = workdiv::GetWorkDiv<Block, Threads>(acc)[0];  
uint32_t blockThreadId = idx::getIdx<Block, Threads>(acc)[0];  
uint32_t gridThreadId = gridBlockIdx * blockThreadExtent + blockThreadId;
```

Lesson 21: 1D Work Division

Printing the index

- Open `helloWorld` again
- Change the kernel as shown below:

```
// Add these lines:  
uint32_t blockThreadExtent = workdiv::getWorkDiv<Block, Threads>(acc)[0];  
  
uint32_t gridThreadId = gridBlockIdx * blockThreadExtent + blockThreadId;  
  
printf("Hello, World from alpaka thread %u!\n", gridThreadId);
```

- Switch to your `build` directory and rebuild:
`cmake --build . --config Release`
- Execute the example again

Lesson 21: 1D Work Division

Summary

- Indices are obtained by the `alpaka::idx::getIdx` functions
- Extents are obtained by the `alpaka::workdiv::getWorkDiv` functions
- Indices can be calculated if required



CASUS

CENTER FOR ADVANCED
SYSTEMS UNDERSTANDING

www.casus.science