DeviceQuery, bandwidthTest, and warming up the device

Exercise 1:

As in the previous weeks, you need to connect to the LSF front-end node first, i.e. `ssh login3.hpc.dtu.dk`, or with the `lsf10sh` command from within a ThinLinc session.

From there you can use `voltash` to get an interactive session on a GPU node.

In order to use CUDA you need to load the following modules (gcc > 6.x is not supported)

```
module load cuda/9.1
module load gcc/6.3.0
```

These can be inserted into your `.bashrc` for convenience. Please note that `module load cuda/9.1` will also succeed on nodes without GPUs (you can work on such nodes, but running GPU code will produce an `cudaErrorInsufficientDriver` error).

1. Run `nvidia-smi` to check the GPUs on your node and whether they are used.
2. Run `/appl/cuda/9.1/samples/bin/x86_64/linux/release/deviceQuery` to see the details of the GPUs (you should do this several times this week to become familiar with it).

   Run `lscpu` and `free` to check the CPUs on your node and the size of the main memory.

   Make a note of the main differences between the CPUs and the GPUs by writing down the key specifications (core counts, clock rates, cache sizes, main memory sizes).
3. Run `/appl/cuda/9.1/samples/bin/x86_64/linux/release/bandwidthTest` to measure the effective bandwidth for transferring data CPU $\leftrightarrow$ GPU and GPU $\rightarrow$ GPU. Use options `--memory=pageable` and `--memory=pinned` to see the difference between having normal vs. pinned memory (we will learn later what this means).
4. The CUDA C file `myDeviceQuery.cu` is provided on CampusNet. Compile this file using the `nvcc` compiler and run it.

   ```
   nvcc -I/appl/cuda/9.1/samples/common/inc myDeviceQuery.cu -o myDeviceQuery
   ```

   Look into the source code and familiarize yourself with it.
5. The CUDA C file `warmUpDevice.cu` is provided on CampusNet. Compile this file using the `nvcc` compiler and run it.

   ```
   nvcc --compiler-options="-fopenmp" warmUpDevice.cu -o warmUpDevice
   ```

   Look into the source code and familiarize yourself with it.