

Heterogeneous and Cloud Computing

Homework #7

Max Points: 25

Due: Mon, Nov 10 2014 By 11:59 PM (EST)

Email-based help cut-off: 24 hours prior to due date

Objective: Develop an OpenCL sorting program that can sort a list of words and can operate on different devices.

Submission: Once you have successfully tested your program and verified it meets all the formatting requirements, upload:

1. The C++ source file(s) for this homework.

Grading Rubric:



This is a graduate course and consequently the expectations in this course are higher. Accordingly, the C++ program submitted for this homework must be operational in order to qualify for earning a full score.

NOTE: Program that do not compile, have methods longer than 25 lines, or just some skeleton code will be assigned zero score.

-1 point: For each style violations as reported by the C++ style checker (a slightly relaxed version from Google Inc. automatically downloaded by `Makefile` from Niihka)

-3 points: If the program does not include suitable comments at appropriate points in each method to elucidate flow of thought/logic in each method.

-1 point: For each warning message generated by `g++` when compiling with `-Wall` (report all warnings) flag.

NOTE: The C++ program must perform only reading and writing of words. **Your program will be assessed based on how efficiently it organizes and manages the data.** It is up to you to design and implement your program efficiently.

Homework Program Requirement

Develop a C++ program that uses OpenCL to sort a list of words from a given input file and write the stored list of words to a given output file, specified as the following command-line arguments:

1. The first command-line argument is one of the three strings “CPU”, “GPU”, or “ACC” indicating the OpenCL device to be used is a CPU, GPU, or Accelerator. Use the corresponding device on the first platform on which the device is available.
2. The second command-line argument is the input file name with a list of words, one word on each line.
3. The third command-line argument is the output file name. The output file should contain the sorted list of words.

NOTE: The C++ program must perform only reading and writing of words. **Your program will be assessed based on how efficiently it organizes and manages the data.** It is up to you to design and implement your program efficiently.

Sample Outputs:

A sample output is shown below (and additional files are supplied for testing):

```
$ ./hw7 GPU words.txt words_sorted.txt
$ cat words_sorted.txt
1
1
2
2
3
3
4
4
5
5
five
five
four
four
one
one
three
three
two
two
$
```

Submission

Once you have successfully tested your program and verified it meets all the requirements, upload:

1. The C++ source file(s) for this homework.

Verify that your program meets all the requirements as stated in the grading rubric. No credit will be given for submitting skeleton code or programs that do not meet the base case. Ensure your C++ source files are named with the appropriate conventions. Ensure your solution meets the style guidelines by running it through the style checker. **Upload each file individually. Do not upload zip/7zip/tar/gzip or any other archive file formats.**