Programming Parallel Architectures (CCE3015)

Assignment 1 – Problem Research and Planning

Due: 17 Mar 2020, 16:00

Prof. Johann A. Briffa

Instructions

- This assignment is to be attempted individually. It is essential that the work you submit and present consists only of your own work; use of copied material will be treated as plagiarism. Discussion is only permitted on general issues, and it is absolutely forbidden to discuss specific details with anyone.

- Your assignment submission consists of the following deliverables:

  1. A report, submitted on the VLE. This **must** be a single PDF file less than 20 MiB in size. Be particularly careful with the sizes of included images, which can easily cause the PDF to be too big.
  2. An archive containing the software developed as part of the assignment, submitted on the VLE. This **must** be a single archive file less than 20 MiB in size. If your assignment requires the submission of data files that exceed this limit, please discuss this with me beforehand.

  If either of these submissions is late, the whole unit of assessment will be considered a late submission, even if any part was submitted on time. Other methods of submission will not be considered. Refer to the relevant assessment form for details of the marking scheme.

- The submitted archive should:
  - Contain all relevant Eclipse projects and any other requested material.
  - Contain only the sources and project settings (i.e. remove all binaries and build folders).

- Each solution requested should be presented as a separate Eclipse project.

- The name of each Eclipse project must start with your username. This may be followed by a dash and any text you wish.

- Textual answers to any questions must be included in the report. Each answer must clearly identify the question number to which it refers.

- In your submission, include only files and projects that *directly* answer the questions asked. Submission of irrelevant material may lead to a reduction in the grade obtained.

- Any submitted Eclipse projects should work *without modification* on NVIDIA Nsight Eclipse Edition as installed on the lab computers, with default settings (i.e. must not depend on any settings specific to your login). If your projects do not work in this way, it may lead to a reduction in the grade obtained.

- This assignment is to be submitted on the VLE by the above deadline; late submissions will be rejected and assessed with a grade of zero.

- If there are extenuating circumstances which do not allow you to complete the assignment on time, you are required to follow the procedure specified in the regulations.

Research Exercise

This exercise asks you to research a classic or current algorithm from scientific computing that is suitable for parallelization, consider the difficulties involved in implementing a parallel solution and plan a strategy for parallelizing the program. Choose a scientific computing domain of your preference and within this domain find an algorithm of non-trivial computational complexity that is suitable for parallelization. It is worth taking into account the following points:

- It is expected that the parallel port to be on CUDA; no alternative parallel platform will be allowed.
- The problem that you choose has to be sufficiently complex to benefit from a parallel solution.
- At the same time, it needs to be simple enough for you to implement within the constraints of this class (i.e. within the time allocated to the second assignment and with the resources available in the labs).
- The algorithm you choose must be amenable to parallelization; not all interesting problems can be divided this way!
- If you're unsure where to start looking, the following fields are good starting points:
  - Image processing (filtering, denoising, etc.)
  - Image/video compression
  - Error correcting codes (the decoding algorithms)
  - Physics/mechanics/fluids simulations
- Given the constraints of this class, it is helpful if you focus on a specific problem rather than a general one.
- As a guide to the level of complexity expected, consider the example problem provided of Lab 3. Your chosen problem should not be simpler than this. It is recommended that the problem should not be much more complex.
- Furthermore, in choosing a problem to work on, keep in mind that you are expected to spend a total of 25 hrs on this assignment and a further 45 hrs on the second assignment.
- Many of you will be considering simple image processing operations such as interpolation, blurring, etc. While in principle these are sufficiently complex, this really depends on various details, such as whether there are any user-selectable options (e.g. neighbourhood size). In any case, if the number of arithmetic ops per pixel is low, you will need to take into account memory bandwidth.
- Whatever problem you choose, make sure there is no readily available solution already (it is more likely that a solution exists for the simpler and more common algorithms). Possible places to look include the CUDA SDK itself (not the toolkit, but the package of sample code).

The questions to be answered start below.

1. Describe the context of the problem and the specific algorithm you have decided to tackle. The algorithm description must be sufficiently detailed to allow someone to implement it. Use of standard software documentation methodology is recommended. Include references as necessary. **[20 marks]**
2. Justify your choice, taking into account the points mentioned above. This justification needs to consider:
   - whether the problem can be parallelized sufficiently,
   - whether there are non-obvious details affecting how the problem is parallelized and that can be used to optimize the solution, and
   - whether there are any previous parallel implementations, in which case the justification also needs to explain how the proposed solution is different. **[10 marks]**
3. Plan for a parallel implementation:
   (a) Identify the operations that can be performed in parallel. **[20 marks]**
   (b) Identify any required synchronization points. **[10 marks]**
   (c) Discuss how your chosen approach maps to the parallel platform's architecture. **[10 marks]**

4. Write a serial implementation of this algorithm. This solution should not require any libraries other than the C++ STL and the Boost C++ libraries as installed in the labs. Your solution will be assessed according to the following criteria, as evidenced by the solution and its documentation:

   (a) Project setup that works *without modification* on NVIDIA Nsight Eclipse Edition as installed on the lab computers, with default settings. Your setup must include a debug build that disables compiler optimization and includes debug symbols and assertions, as well as a release build that enables compiler optimizations and excludes assertions. The Eclipse project name must start with your username, followed by a dash and any text you wish. **[5 marks]**

   (b) Quality of implementation; all compiler warnings should be properly resolved, the program should be complete and works correctly under all input conditions; performance improvement should only be possible with aggressive optimization. **[15 marks]**

5. Test and verify your serial implementation. Include your test data and results. **[10 marks]**

Notes:

- The report should be written in double-column IEEE transactions style, with a font size of not less than 11 pt. Templates for LaTeX and Word can be found in the IEEE Digital Author Toolbox site [1].
- The report should not exceed six pages in length, including any title, figures, references, and appendices. If a longer report is submitted, only the first six pages will be assessed.
- The grade achieved is not proportional to the report length. It is not *expected* that reports should be any specific length, and as long as the questions are answered fully, the maximum mark is achievable.

References

[1] "IEEE author digital toolbox," Jan. 2012. [Online]. Available: http://www.ieee.org/publications_standards/publications/authors/authors_journals.html