



Concurrent and Parallel Systems (SET10108)

Semester 1 –
2023/2024

Learning Outcomes Covered:	LO1,LO2,LO3,LO4
Assessment Type:	Practical Assessment / Demonstration
Overall module assessment	100%
For this assessment:	60%
Assessment Limits:	4000 words for report
Submission Date:	Wednesday, 06 December 2023
Submission Time:	17.00
Submission Method:	Via Moodle
Turnitin:	Not Applicable
Module leader:	Babis Koniaris
Tutor with Direct Responsibility:	Babis Koniaris, Amjad Ullah

- You are advised to keep a copy of your assessment solutions.
- Please note regulation Section B5.3.b regards component weighting.
- Late submissions will be penalised following the University guidelines as follows: Up to 5 working days late the grade will be capped at P1, and F5 after 5 working days.
- Extensions to the submission date may only be given by the Module Leader for exceptional circumstances. – by submitting appropriate request form from [Extenuating circumstances](#).
- Feedback on submissions will normally be provided within three working weeks from the submission date.

The University rules on Academic Integrity will apply to all submissions. The [student academic integrity regulations](#) contain a detailed definition of academic integrity breaches which includes use of commissioned material; knowingly permitting another student to copy all or part of his/her own work

You must not share your work with other students - this includes posting any of your work in any repository that is accessible to others (such as GitHub) and applies also after you have completed the course. You must not ask coursework-related questions in online for a (such as Stackoverflow) and you must not use ChatGPT or other generative AI tools – this would constitute academic misconduct as it would be commissioning material.

By submitting the report, you are confirming that:

- It is your own work except where explicit reference is made to the contribution of others.
- It has not been submitted for any module or programme degree at Edinburgh Napier University or any other institution.
- It has not been made with the assistance of Artificial Intelligence (AI) tools **[except where and how as has been clearly stated]**.

Academic Skills Support: In advance of submission, you can access the support of the academic skills team. They can help you with any aspect of the assessment that you might struggle with, that is

not content related. For example, they can help with time-management, effective reading and note-making, and any aspect of academic writing that you might struggle with. This support is provided through workshops and individual appointments which are bookable online via MyNapier: [Improve your Academic & Study Skills \(napier.ac.uk\)](https://www.napier.ac.uk/improve-your-academic-study-skills).

They are also able to provide formative feedback, within 5-days of your deadline, on a draft section, helping you to identify any issues of focus, structure or academic integrity in your writing. Please email Annemarie Douglas for any specific academic skills support you require: a.douglas@napier.ac.uk

By submitting the report, you are confirming that it is your own work. Please include the following declaration on the first page of your final report:

Declaration

I declare, except where explicit reference is made to the contribution of others*, that this assignment is the result of my own work and has not been submitted for any module or programme degree at the Edinburgh Napier University or any other institution. This is in accordance with Edinburgh Napier University's Academic Integrity Regulations.

***IMPORTANT:** Contribution of others may include use of Artificial Intelligence (AI) tools (details of which can be found in the Guidelines for Students on AI & Writing Assistant Tools). Please declare here whether you have used such tools, and to what extent:

- ☐ NO I have not used such tools
- ☐ YES I have used such tools and I have provided details below and included sample prompts and responses in an appendix.

If you answered YES here, please, in around 100 words, describe how (and at which points) you have used such tools to support your completion of this assessment.

This is the second part of the assessment for this module.

1 Coursework 2: N-Queens

You should write a C++ program to meet the specification below.

1.1 Specification

The N-queens problem is a generalisation of the eight queens puzzle, which is the problem of placing eight chess queens on an 8x8 chessboard so that no two queens threaten each other; thus, a solution requires that no two queens share the same row, column, or diagonal. The N-Queens problem generalises this puzzle by placing N queens on a NxN chessboard, for which solutions exist for all natural numbers N except 2 and 3. It is a standard constraint solving problem, and you can easily find several implementations online. One such implementation is given to you, in the github repository for the module's labs (it's titled "cw2").

Your task is to modify the implementation so that it runs in parallel, using OpenMP (in one variation) and GPUs (in another variation). The task is not trivial, as the implementation uses recursion, and OpenMP and GPUs don't work well with recursion. So, you have to reformulate the algorithm so that it works without recursion, and execute your modified algorithm in OpenMP (easier) and GPUs (harder). Your program is expected to run for N values between 4 and 10, and you can attempt more than 10 if your performance is tolerable.

An easily parallelisable solution is a brute force approach, where you can test every potential board combination, independently, for validity, and identify the valid solutions. For such an approach, you should investigate how you can enumerate all solutions, using a loop and a function that converts an integer (which might need to be 64-bit for larger N values) to a solution configuration.

You should report performance, and it should not surprise you if performance using OpenMP or GPU is slower than the serial version. Be methodical in your performance measurements, by executing methods multiple times, averaging results and comparing the 3 methods (serial, OpenMP and GPU) for each of the N values.

1.2 Report

Your report should include a section per method (OpenMP and GPU), where you present how you parallelised the problem, and your results, compared to the serial version. In a final section you can report your findings and observations by comparing

and discussing all three methods together. You should also include your hardware setup in the report. The report should be no longer than 4,000 words.

1.3 Marking Scheme

Marks for this coursework will be allocated as follows:

- OpenMP 25%: OpenMP: implementation correctness (5%), quality (5%) and documentation in report (5%). Testing different variants is encouraged (e.g using different scheduling methods), and can get you up to 10% more.
- GPU 25%: GPU implementation, using OpenCL or CUDA: implementation correctness (5%), quality (5%) and documentation in report (5%). Testing different variants is encouraged (e.g using different workgroup sizes), and can get you up to 10% more.
- Report 10%: Suitable performance analysis, presentation of results and report writing.

2 Submission

You should submit your assignment via Moodle. The assignment deliverable needs to contain the source code and everything else needed (e.g. CMakeLists.txt file) to build the project and generate an executable, as well as a report discussing the design of the application, your hardware setup and the results, including graphs showing timings, scalability, etc. Please name your file as SET10108_\$YOUR-ID_coursework2.zip (rar/7z are also fine).

Good luck!