AMME5060 ADVANCED COMPUTATIONAL ENGINEERING

Major Project, 43% Course Weighting, Due Week 13 Friday*;
This assignment is estimated to take 20 hours to complete (for each group member).
* Presentation and Code Due in Friday Lab Session; Report and Code submitted online by 11.59pm

---

In small teams you will develop and test a high performance computing program to solve to an engineering problem of your choice. Considerable latitude will be given on the choice of project however it is recommended you focus on the computational and numerical methods used in this course so far.

General Project Guidelines:

1. Project should be computational. Your program should calculate solutions to some engineering problem or otherwise computationally obtain some relevant engineering result (i.e. optimise, minimise something etc). The problem could be very simple (e.g. 2D steady diffusion problem) or very complex (e.g Navier Stokes Solver).
2. The program you develop should be able to exploit high performance computing facilities. Your code should be parallelised using MPI and be able to run efficiently on at least 12 cores.
3. The program should be efficient and make sophisticated use of the strategies introduced in this unit. You need to test the performance of the code using timing statements to determine where program is spending the most time. You should investigate how to improve speed with different algorithms or coding and report on these investigations.
4. The project should contain methods, techniques, strategies or numerical modelling concepts that extend beyond the content covered in lectures. For example you could choose to use some advanced MPI strategies they we have not covered in class (or only covered briefly). You should choose one or at most two areas to direct your efforts to extend your knowledge and skills.
5. Marks are awarded for the project complexity and quality of your implementation, testing and reporting.

Project Ideas:

1. Choose a <u>complex engineering problem/equation</u> and write a parallel solver to produce solutions for a restricted set of boundary conditions. e.g. 2D/3D Navier Stokes solver (if you have completed AMME5202 or similar unit) or some other complex physical described by PDEs (e.g shallow water equations in 2D, wave equation, reaction diffusion/transport problem)

2. Implement and test an <u>advanced numerical method</u> and test on a simple generic problem such as the 2D diffusion problem. e.g. implement an advanced linear solver BICGStab or Multi-grid solver and test performance.

3. Implement and test an <u>advanced computational method</u> on a simple generic problem. e.g. test the performance of advanced MPI programming methods to improve code efficiency and code flexibility on a simple problem. For example efficient use of non-blocking communication on a problem with 3D domain decomposition using customised MPI features or try writing hybrid parallel code with both OpenMP and MPI to take advantage of the best features of both.

4. Something else. Please discuss with me. Most aspects of the project brief are negotiable. All projects should be approved by week 8 Friday (can change topic with further approval).

Project Approval:
All projects must be approved by Friday Week 8 in the laboratory session or Week 8 by appointment with me. Write a short (approx. 1/2 page) project description and allocate group member tasks. (see last page of this document).

Testing:
The code should be run on Artemis HPC facility demonstrate speed, parallel efficiency and accuracy. Training on how to do this will be provided in week 9. You will be required to demonstrate your code is functioning correctly by benchmarking against known results. You should provide tests cases in your final submission.

Group Selection:
Self select groups of 3 (I will consider other group sizes on basis of project complexity). Individual and group marks will be given. Your project must contain identifiable individual components. e.g. tasks could be partitioned among group members along these lines: numerical method and coding/parallel algorithm and MPI programming/engineering problem selection and governing equations and modelling/data structure, file format, file input-output and visualisation/testing and benchmarking. Individual marks will be awarded in proportion to task complexity and quality of work. All group members must contribute to code writing and report writing.

Assessment/Grading:
You will be assessed in three components:

1. **Code Submission, due Friday Week 13 (online at 11.59pm) :** You will be required to submit your code as a package with instructions on how to compile and run. The code should be written in Fortran and parallelised in MPI or OpenMP. The code should be extensively commented.
2. **Report Submission, Due Friday 11.59pm Week 13 :** You will be required to submit a single group project report. You are required to provide:
   (a) a description of the problem, the significance of the problem, what equations describe this problem and any assumptions you have made
   (b) your numerical method. Describe any limitations in this approach
   (c) your program, how it is structured, the inputs and outputs. Provide a brief user guide.
   (d) results of your testing to validate your code
   (e) results to demonstrate the efficiency of your program and parallel performance.
   (f) describe any coding strategies you have employed to make the code efficient.
   (g) individual contributions must be identified very clearly in the report and in a cover contributions statement (which must be signed by all members)
   (h) Reports should be concise. No formal page limit is set but please see me if your report is much more than 15 pages with figures and 12pt font.
3. **Presentation/Oral Exam, held on Friday Week 13:** A short presentation and question and answer session will be held in week 13 both during the timetabled laboratory time and also in some time slots outside laboratory time. As a group you will be required to briefly present your work. You will be asked individual questions about your work/method. This component will be like a short oral exam. Both group and individual marks will be awarded.

Final Submission:
Detailed submission guidelines will be provided closer to submission date

Marking Guide:
Grading is 50% individual contribution and 50% for group contribution.

	Mark /43
Project Complexity and Solution Quality:	22
Project complexity and difficulty Innovation and demonstration of self-directed learning of advanced concepts Design quality, including sophistication of solution, numerical methods, algorithms Efficiency of parallel strategy Efficiency of implementation, coding Overall quality of solution	
Report Submission:	6
Introduction and discussion of project aims Numerical methods are presented Computational methods are presented Brief user guide Discussion and interpretation of results and testing Discussion of computational efficiency and accuracy Presentation quality and clarity Individual contributions identified in detail	
Testing (in report):	5
Demonstration of computational efficiency Demonstration of computational accuracy Program verification (e.g bench-marking)	
Code and Package:	5
Clearly commented Compilation instructions are provided Test case(s) are provided Individual contributions are identified in detail Code structure (Efficient and readable code)	
Oral Presentation:	5
Understanding of problem and methods Ability to respond to questions	

Academic dishonesty:
We reserve the right to conduct a further oral exam on any submitted work. This would be used where there is reason to believe there is a breach of academic honesty requirements or if it is unclear what the individual student contributions are. In these cases the oral exam grade would be implemented as a multiplier on the grade determined using the marking scheme above. Further individual followup supplementary oral exams would be held over week 14-15.

You need to preface your report submission with a short statement, in your own words, that indicates you have read the University of Sydney requirements on academic honesty and that all work submitted is that of the group members and according to the contribution statement described above. If you have taken any code from another source or had assistance in writing your code you must in detail, exhaustively and completely describe these sources and contributions. No project will be graded until this statement is submitted.

Project Proposal (due week 8)

Write a short ($\sim 1/2$ page) project description and allocate group member tasks.

1. Describe the engineering problem, governing equations and boundary conditions you want to solve. Draw a picture if relevant. (e.g. Steady Heat equation on 1D rod with Neumann boundary conditions)
2. Describe how you are going to numerically solve this problem. (e.g. Finite difference on regular structured grid with Jacobi Solver etc)
3. Describe your aims for the program and where you intend of introduce complexity (e.g. implement advanced linear solver or custom MPI routines)
4. Provide a preliminary overview of the tasks required and allocate group members to each task.
5. Provide a preliminary time-line for each component
6. Discuss your problem with the tutors and myself
7. Allocate yourself to a Group in Canvas and give yourself an interesting group name.