



FIT3143 – Assignment 1

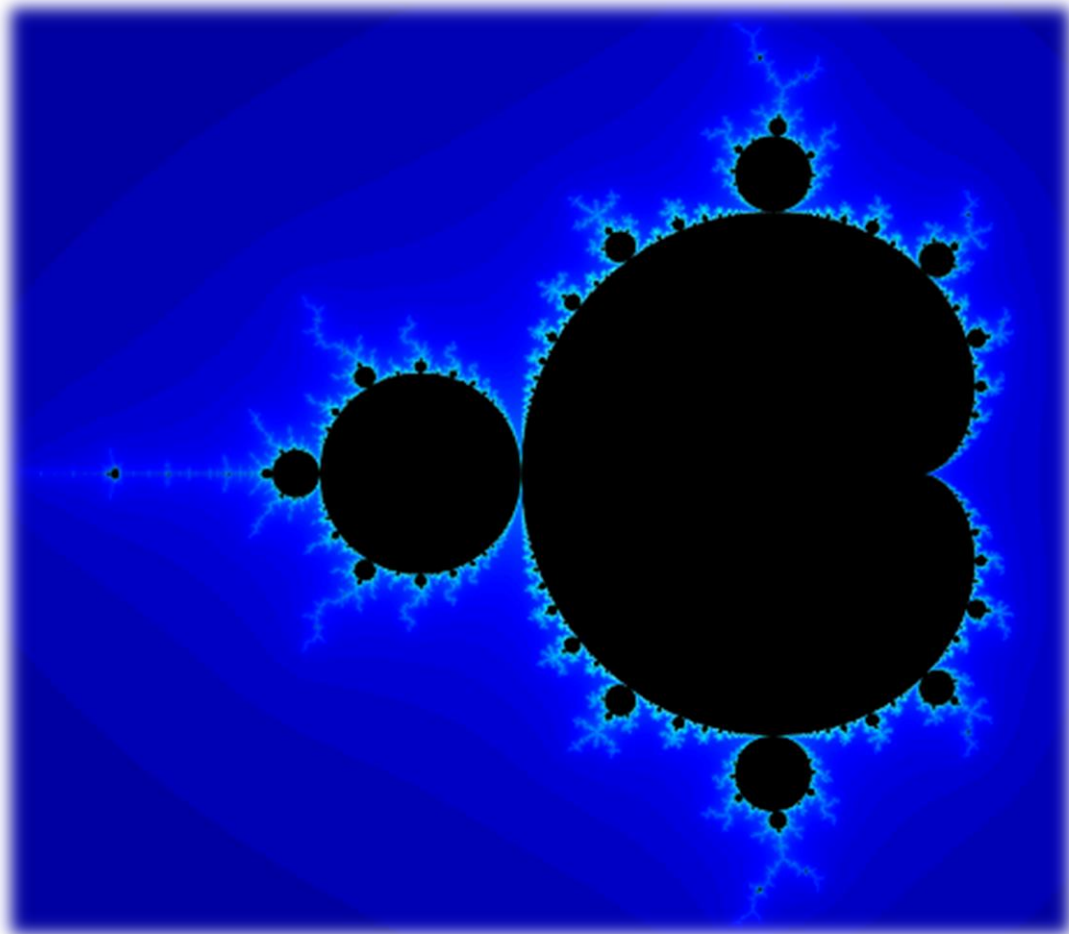
- Marks:** 20 marks and 20% of all marks for the unit.
- Due Date:** Week 6, Friday 6/September/2019, 11:55 PM Melbourne time.
- Submission:** (i) The assignment submission must be made through Moodle for this unit **by the due date**. (ii) The submission of this assignment must be in the form of a **single PDF file and a C code file**.
- Extensions:** *No extensions will be given.*
- Lateness:** *Late penalty of 10% per day* after the due date, including the weekends.
- Authorship:** This assignment is an individual assignment and the final submission must be identifiably your own work. Breaches of this requirement will result in an assignment not being accepted for assessment and may result in disciplinary action.
- Cover Sheet:** A completed individual assignment covered sheet is required with the submission. This coversheet should be appended with the assignment report (PDF file). Please refer to the submission details at the end of this file.



Assignment Question

The [Mandelbrot Set](#) is a visual representation of an iterated function $f(x) = x^2 + C$ on the complex plane, where C represents a constant complex number.

The file `mandelbrot.c` (provided with this assignment) is a C code to calculate the Mandelbrot set, and will yield a big version of the following picture:



The format of this picture is `.ppm`, and you can use the Gnome Image Viewer to display the picture. However, if the picture size is too larger, you could opt to install and use [GIMP](#) to display the picture.

If you are keen to find out more on the workings of the Mandelbrot algorithm, the following web links will help you:

1. <http://www.ddewey.net/mandelbrot/>
2. <http://www.mandelbrotset.net/tutorial.html>



Instructions:

1. Compile the **mandelbrot.c** code and run the program serially for 5 times on a multi-core computer. Use the **clock()** function in the code to measure the computational time of the program.
 - a) You should have 5 sets of measures time results.
 - b) From the time results, obtain an average time taken to complete the program execution.
2. Decide on a parallel partitioning scheme for this program. In your report, with the aid of diagram/s, describe the partitioning scheme. Justify the selection of the partitioning scheme.
3. Based on your partitioning scheme, modify the code to become a parallel program utilizing **Message Passing Interface (MPI)**
 - a) Note that all the data needs to be sent back to the master node to be written to the file.
 - b) Instead of using the **clock()** function, use the **MPI_Wtime()** function to measure the time taken to complete the program execution. The timing code must only be run on the master program. Be sure to place the timing code correctly so that the timings will validly calculate only the computational time and the communication time.
 - c) Run the MPI program 5 times on a multi-core computer. Specify the number of logical processes used in the table.
 - d) Obtain an average time taken to complete the program execution for part 5(c).
4. Plot your results (serial vs parallel program) on a graph. Evaluate your results for part 5 in your report to justify the selected partitioning scheme.



Submission:

Each implemented code must have a comment header, containing your name, ID number, and email address. Marks will be deducted if this information is missing.

Use the following table format in your report:

Computer specifications		<i>a) Specify CPU model (e.g., Intel Core i7-8700k)</i> <i>b) Specify number of logical processors (e.g., 8)</i> <i>c) Specify memory size (e.g., 8 GBytes of RAM)</i> <i>d) Specify network speed (e.g., 1 Gb/s)</i>			
Value of iXmax		8,000 (default)			
Value of iYmax		8,000 (default)			
Value of IterationMax		2,000 (default)			
	Serial program	Parallel Program			
		MPI			
		(e.g., 2 logical processors)	(e.g., 4 logical processors)	(e.g., 6 logical processors)	(e.g., 8 logical processors)
Run #1					
Run #2					
...					
Run #5					
Average time					

You can modify the table above to suit the representation of your results. For instance, you can include an extra test to run your MPI program on 10 logical processes or if you decide to run the program using the MonARCH cluster at Monash University (Details to access MonARCH will be provided in Moodle or via e-mail).

Compute the speed up and a plot a curve to observe the scalability of your algorithm for an increasing number of logical processors (i.e., sub-linear or linear speed up observation). Explain your observation in the report.

Report layout:

- I. Abstract (brief overview of the assignment, chosen partition method and summary of results).
- II. Introduction (Basic theory and mathematics of Mandelbrot Set).
- III. Theoretical speed up analysis for a Mandelbrot Set (using Bernstein's and Amdahl's analysis).
- IV. Partition Scheme for data parallelism (Include illustration to describe the selected scheme, and flowchart (or pseudocode) of the parallel algorithm).
- V. Results and Discussion of the implemented partitioning scheme (Compare the actual speed up with that of the theoretical speed up with reasoning).
- VI. Conclusion.



The report should be formatted using an [IEEE](#) template.

Please refer to the assessment rubric which is enclosed together with this specification. This rubric should assist you to attain a better understanding on the marking guide for the assignment.

Please do not attempt to plagiarise or collude your assignment work.

Note that it is a University requirement (<http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-plagiarism-collusion-procedures.html>) for students to submit an assignment coversheet for each assessment item.

Faculty Assignment coversheets can be found at <http://www.infotech.monash.edu.au/resources/student/forms/>. Please include the coversheet as the first page in your assignment report before submitting the report and code into Moodle.

Submission is to be made in Moodle for FIT3143. An assignment submission link will be provided. Each student is required to submit the following files:

- a) Assignment report (which includes the coversheet),
- b) Assignment code (.c file extension). If you are submitting multiple code files, please zip these files and submit the zipped file into Moodle.