Department of Computing and Mathematics

ASSESSMENT COVER SHEET 2024/25

Unit Code and Title:	6G7V0025 High Performance Computing & Big Data
Assessment Set By:	MK Bane, S Ajao
Assessment ID:	1CWK100
Assessment Weighting:	100
Assessment Title:	HPC-BD Assessment
Type:	Report
Hand-In Deadline:	See Moodle
Hand-In Format and Mechanism:	Single PDF upload to Moodle/Turnitin

Learning outcomes being assessed:

LO1	Understand challenges in large scale computation and in big data, and select and apply appropriate techniques across various system architectures.
LO2	Implement solutions for resource- and data-intensive problems using appropriate techniques and tools on various system architectures.
LO3	Apply a wide range of transferable skills and attributes applicable to real-world problems and scenarios

Note: it is your responsibility to make sure that your work is complete and available for marking by the deadline. Make sure that you have followed the submission instructions carefully, and your work is submitted in the correct format, using the correct hand-in mechanism (e.g., Moodle upload). If submitting via Moodle, you are advised to check your work after upload, to make sure it has uploaded properly. If submitting via OneDrive, ensure that your tutors have access to the work. Do not alter your work after the deadline. You should make at least one full backup copy of your work.

Penalties for late submission

The timeliness of submissions is strictly monitored and enforced.

All coursework has a late submission window of 7 calendar days, but any work submitted within the late window will be capped at 50%, unless you have an agreed extension. Work submitted after the 7-day late window will be capped at zero unless you have an agreed extension. See ‘Assessment Mitigation’ below for further information on extensions.

Please note that individual tutors are unable to grant any extensions to assessments.

Assessment Mitigation

If there is a valid reason why you are unable to submit your assessment by the deadline you may apply for Assessment Mitigation. There are two types of mitigation you can apply for via the module area on Moodle (in the 'Assessments' block on the right-hand side of the page):

- **Non-evidenced extension**: does **not** require you to submit evidence. It allows you to add a **short** extension to a deadline. This is not available for event-based assessments such as in-class tests, presentations, interviews, etc. You can apply for this extension during the assessment weeks, and the request must be made **before** the submission deadline. For this assessment, the non-evidenced extension is 2 days.

- **Evidenced extension**: requires you to provide independent evidence of a situation which has impacted you. Allows you to apply for a longer extension and is available for event-based assessment such as in-class test, presentations, interviews, etc. For event-based assessments, the normal outcome is that the assessment will be deferred to the summer reassessment period.

Further information about Assessment Mitigation is available on the dedicated Assessments page.

Personal Learning Plans

If you have a Personal Learning Plan (PLP) which states you can negotiate an extended deadline, make an appointment to see the Department's Disability Coordinator to discuss to discuss your needs, and where appropriate agree on a revised submission deadline.

Plagiarism

Plagiarism is the unacknowledged representation of another person's work, or use of their ideas, as one's own. Manchester Metropolitan University takes care to detect plagiarism, employs plagiarism detection software, and imposes severe penalties, as outlined in the Student Code of Conduct and Academic Misconduct Policy. Poor referencing or submitting the wrong assignment may still be treated as plagiarism. If in doubt, seek advice from your tutor.

As part of a plagiarism check, you may be asked to attend a meeting with the Module Leader, or another member of the module delivery team, where you will be asked to explain your work (e.g. explain the code in a programming assignment). If you are called to one of these meetings, it is very important that you attend.

Use of generative AI

Permitted – with changes

⚠ The use of generative AI is permitted in this assessment, but please make sure you follow these specific instructions:

- For the HPC Task Q2a you are required to use Github Copilot (and only Copilot is permissible – any other genAI will be treated as academic misconduct since you do not have permission to share the assessment codes). You should use Github Copilot to generate a portable distributed memory implementation of the provided example code, and you should record in the Appendix of your report the details of the version of Copilot and the actual prompts you use.
  Without use of genAI, you then need to critique the generated code in your words (see Q2b). You are required to run the code yourself and to explain (in your words) how the code performs. You may, additionally, tune the code yourself (see Q2c).

For any other uses of generative AI, you should also follow the instructions in the 'Are you allowed to use AI in assessments?' section of the AI Literacy Rise Study Pack or speak to your tutor. All submitted work must be your own original content.

If you are unable to upload your work to Moodle

If you have problems submitting your work through Moodle, you can send your work to the Assessment Management Team using the Contingency Submission Form. Assessment Management will then forward your work to the appropriate person for marking. If you use this submission method, your work must be sent **before the published deadline**, or it will be logged as a late submission. Alternatively, you can save your work into a single zip folder then upload the zip folder to your university OneDrive and submit a Word document to Moodle which includes a link to the folder. **It is your responsibility to make sure you share the OneDrive folder with the Module Leader, or it will not be possible to mark your work.**

Assessment Regulations

For further information see the Postgraduate Assessment Regulations on the Assessments and Results information pages

Formative Feedback:	The tutor/s are happy to give feedback on your progress. This is available by making an appointment with the course tutor/s
Summative Feedback:	Students will receive individual feedback relating to their marks, and a document will be published covering more generally how the assessment should have been tackled

High Performance Computing and Big Data

The assessment comprises a single Report discussing two tasks (of 50 marks each): an HPC Task and a Big Data Task. **You are required to submit a single PDF file to Moodle.** This file should contain both your HPC and Big Data reports and a single Appendix. The report should not include a table of contents, index, nor abstract. Your referencing scheme should follow Cite Them Right Harvard as explained in depth at Manchester Metropolitan University (no date). The report must be in English with good grammar and English spellings – you are highly recommended to use a grammar and spell checker. Marks will be lost for poor referencing and poorly written reports. The style of your report should be formal and written in the passive voice. Guidance on report writing is available from https://www.mmu.ac.uk/student-life/course/study-skills/online

See "Submission" section below for full details.

HPC Task

This task requires you to apply the skills and knowledge from the lectures, labs and recommended reading materials. You are provided with example source codes that determine the variance of a set of input numbers. You will need to examine (and improve) the performance of the provided codes using MS Azure Labs. You are required to write a report that clearly explains your methodologies, your results and your conclusions.

Details

You are given two versions of a C code that determines the variance of a set of input numbers. One of your labs will focus on explaining, compiling and running the serial version of this code. You will also be given an initial parallel implementation using the shared memory programming model. These are available from github at **`git@github.com:mkbane/mmu-hpc.git`** in the 2024-2025 subdirectory. If you cloned this repository prior to Monday 10 February 2025 you will need to use "git pull" to obtain the 2024-2025 subdirectory. Please ask tutors if need help.

You have 3 questions to undertake and in cases you should compile your code with zero optimisation (namely explicitly using the "-O0" flag. For your parallel implementations you should run on the appropriate number of cores, using a sound timing methodology for obtaining appropriate performance data (e.g. as explained during the course). As noted below you may amend source codes but it is imperative that your numerical accuracy is maintained. For the assessment you should use 250 million data points that are autogenerated, by making use of the provided autogenRoutines.c file.

Q1: OpenMP (15 marks)

Consider and discuss whether and how the OpenMP version can be improved in terms of performance, running the code as appropriate to yield data to back up your discussion. There are 15 marks available for this question, split 5 marks for methodology, 10 marks for improved performance, 15 marks for your discussion of what you have amended and why. For this question, you should follow the instructions in the 'Are you allowed to use AI in assessments?' section of the AI Literacy Rise Study Pack. All submitted work must be your own original content.

Q2: Distributed Memory Implementation (25 marks)

You are required to create a distributed memory implementation of the code, that is portable across leading HPC infrastructures. This comprises 3 stages

a. You should use Github Copilot to generate a portable distributed memory implementation of the provided example code and you should record in the Appendix of your report the details of the version of Copilot and the actual prompts you use.
Note that it is only permissible to use Copilot. Uploading the assessment codes to any other generative AI is equivalent to sharing the codes and permission has not been given to you to do so. Therefore, anybody using genAI tools other than Copilot will be reported for Academic Misconduct.
b. In your own words you should critique the generated code, which includes running and explaining the code's performance. For this part of the question, you should follow the instructions in the 'Are you allowed to use

AI in assessments?' section of the AI Literacy Rise Study Pack. All submitted work must be your own original content.

c. You can take the output from 2a and amend the code yourself to fix any issues. For this part of the question, you should follow the instructions in the 'Are you allowed to use AI in assessments?' section of the AI Literacy Rise Study Pack.

It is important you identify in your submission which elements of the code were from Copilot (as allowed within 2a) and which are your contributions.

The maximum marks are: 5 marks for provision of a working solution; 10 marks for discussion of how you obtained the working solution; 10 marks for discussion of performance

Q3: Scaling (10 marks)

For each of your parallel implementations, explain how you will estimate the runtime on 100 cores. Given your estimates, discuss the infrastructure required to achieve those estimates.

HPC Task Submission

For the HPC Task, you are required to submit a short, formally written report (3 pages excluding any title page, Figures, Tables or the Appendix) that answers each of the above questions. For all questions you should (i) state in the report the commands you use related to compiling and running; (ii) discuss the numerical accuracy of each run; (iii) explain in your report your timing methodology. This report (& appendix) will form part of the single PDF you submit for the assessment.

Furthermore, you must provide access to any *amended* source code, *all* scripts discussed, and *all* results data quoted in the report. Further details are given under "Submission" below.

Big Data Task

Lectures and labs will explain concepts and technologies around "Big Data", with hands on experience using technologies such as Hadoop, Apache Spark, Scala and Kafka. Your assessment task will be to take this experience, as well as the learning materials and further reading, to apply to a Big Data problem.

You have been given a Research Hypothesis that can be answered from the data provided. You will need to whether the research Hypothesis is true. This determination will primarily be by coding a Big Data solution. You can use any supporting reference to support your report.

Details

Provided dataset: The Amazon electronics products reviews dataset is available for your download from Kaggle. Note that the dataset contains millions of rows and cannot be analysed in basic desktop applications such as MS-Excel. However, it can be ingested directly into your application and Azure Labs using the following URL:

https://www.kaggle.com/api/v1/datasets/download/saurav9786/amazon-product-reviews

The dataset has the following attribute Information:

- userId : Every user identified with a unique id (First Column)
- productId : Every product identified with a unique id(Second Column)
- Rating : Rating of the corresponding product by the corresponding user(Third Column)
- timestamp : Unix Time of the rating (Fourth Column)

Step 1: Based on your lectures and labs, your Research Hypothesis is: **"In the second quarter of 2014, products given a review rating of 3 or more are significantly different compared to other products"** which you may refine to make more precise and/or testable from the data available in the provided dataset. You should only use the provided dataset for the Hypothesis.

Step 2: Write your report where, for the above Research Hypothesis, in a clear, concise and consistent manner, you should not exceed **1000 words** and:

- ➢ include a title page giving your name, MMU ID, signed declaration the work is your own
- ➢ state the Research Hypothesis and your test for determination of whether it is true
- ➢ explain the results and discuss your approach and what you have learned from the data
- ➢ detail your test on the required data and state whether the Hypothesis has been found true or not, or what you would need to do next to obtain a conclusive result

Step 3: End-to-End Big Data Pipeline - This section should not exceed **500 words for the technical report.** This section assesses your competency with the **core big data tools** and your ability to integrate them into an **end-to-end pipeline**.

You have been given a **public dataset** (or you may choose one of your own from a reputable open data source). Your task is to **design and implement** a **big data pipeline** that showcases your understanding of the following technologies within your Azure Labs environment or Google Colab if you prefer:

1. **Hadoop (HDFS)** – for distributed data storage.
2. **Spark** – for data processing or a machine learning task (Spark MLlib).
3. **Kafka** – for streaming data ingestion (can be real or simulated).
4. **Scala** – as the programming language for your Spark and Kafka integration code
5. **Data Source**: Pick a **public dataset**. For instance:
   a. **OpenWeather API** for streaming weather data, or
   b. **Mockaroo**-generated data

Marking Scheme

Maximum marks for the report are broken down into the following categories, with actual marks per category given in line with the "Marking Criteria" section of the Assignment Cover Sheet (available via Moodle):

- **8 marks** for concise/readable report
- **4 marks** for appropriate use/level of references
- **8 marks** for regarding Research Hypotheses (hypothesis statement, testing strategy,
- discussion of wisdom gained from the data, conclusions from testing)
- **10 marks** regarding coding solution to Research Hypothesis
- **20 marks** for design of a big data pipeline
  - o Technical Implementation (6 marks)
  - o Data Transformation & Analysis (6 marks)
  - o Integration & Project Cohesion (4 marks)
  - o Clarity and Documentation (4 marks)

Big Data Task Submission

The Big Data part of your report should not exceed 3 pages (excluding title page, references and appendices. Tables and figures should be put in the appendix), including key data plots to justify the points made in your report. This report (& appendix) will form part of the single PDF you submit for the assessment. All codes used should be included as an appendix or committed to a private GitHub repository shared with only the course tutors (m.bane@mmu.ac.uk, S.Ajao@mmu.ac.uk) and MUST **not** be publicly readable.

MARKING

Each of the HPC Task and Big Data Task sections above indicates the maximum mark for each required subtask. The actual mark assigned will be determined by applying the university's marking descriptors given in Table 1. For example, HPC Task Q1 subtask is worth a total of 15 marks and if you get a Pass for this element you will receive between 7.5 and 8.85 marks. Marks for all subtasks are summed and the final figure rounded up to an integer.

SUBMISSION

You can prepare your **HPC report (3 pages) and Big Data report (3 pages)** in MS Word, LaTeX or any preferred word processing software. You are required to have an Appendix (words not counted) that includes a URL to a directory on your MMU OneDrive (or github repository) that is shared only with the tutors. You must save your composite report as a PDF and upload to the assessment area on Moodle.

If you are sharing your work with the tutors via OneDrive, then you need to create a OneDrive directory containing files required by each Task (see above) and shared with only the course tutors (m.bane@mmu.ac.uk, S.Ajao@mmu.ac.uk) and MUST **not** be publicly readable. You must not update any files after you submit your report, and any files with a modification date later than the submission date of your report to Moodle will be investigated. Details on how to upload your work from Azure Labs to OneDrive will be provided and if you need assistance, you should ask the course tutors.

If you prefer to share your work with the tutors using github you may do so. This access should be via (i) inviting each course tutor (m.bane@mmu.ac.uk, S.Ajao@mmu.ac.uk) to be a contributor to your **private** github repository. Your repository must be private, and you must not update any files after you submit your report, and any files with a modification date later than the submission date of your report to Moodle will be investigated. If you require assistance on how to upload your work from Azure Labs to github, you should ask the course tutors.

Ensure your submitted report has a title page and the correct link to your OneDrive directory (or github repository). All reports will be automatically scanned for plagiarism and any exceeding a given threshold (or otherwise flagged) with be passed to MMU for further formal investigation.

References

Manchester Metropolitan University (no date) *Cite Them Right Harvard* Available at: https://www.mmu.ac.uk/library/referencing-and-study-support/referencing/cite-them-right-harvard (Accessed: 18 January 2024)

Marking Criteria

The criteria used in assessing each task is based upon the university’s standard marking grid, namely:

		Fail 0%-19%	Fail + 20%-44%	CF 45%-49%	Pass 50%-59%	Merit 60%-69%	Distinction 70%-85%	Distinction + 86%-100%
HPC/Big Data Programming Solutions	Overall	The work is missing major elements or contains serious errors.	Some elements of the brief have not been addressed. The work is superficial or limited.	An adequate attempt has been made and most elements of the brief have been addressed at a limited level.	The work represents a coherent solution to the assignment brief and demonstrates confidence in the development of parallel computing solutions.	The work addresses the brief rigorously and thoroughly and demonstrates fluency in the development of parallel computing solutions.	The work represents a sophisticated and original solution and shows evidence of reflective practice. The work addresses the brief rigorously and thoroughly and demonstrates fluency in the development of parallel computing solutions.	The work represents a creative and authoritative solution and shows evidence of reflective practice. The work addresses the brief rigorously and thoroughly and demonstrates fluency in the development of parallel computing solutions.
	Analysis and Design	Little or no analysis of the problem, with inappropriate or no design presented.	Limited or superficial analysis of the problem. Design is incomplete or inadequate and fails to address the brief.	Adequate analysis of the problem . Design is adequate.	Clear and careful analysis of the problem leading to a coherent design which fully addresses the brief.	Thorough and careful analysis of the problem and a clear design which fully addresses the brief.	Sophisticated critical analysis of the problem leading to a convincing design which meticulously addresses the brief.	Sophisticated and insightful critical analysis of the problem leading to a creative and convincing design which meticulously addresses the brief.

	Evaluation	Little or no evaluation of the outcomes of the exercise.	Inappropriate, incoherent or erroneous evaluation of the exercise.	Some critical evaluation of the exercise,	A clear, careful and confident evaluation which draws largely accurate conclusions.	A rigorous and precise critical evaluation of the exercise, which is fluently presented.	A sophisticated and meticulous critical evaluation of the exercise, which is convincingly presented, showing evidence of critical reflection.	An insightful and authoritative evaluation of the exercise, which is convincingly presented, showing evidence of critical reflection.
	Implementation	The implementation fails to compile or function. The source code is unclear, without comments and clearly named variables and functions.	The implementation contains major errors. The source code is unclear, without comments and clearly named variables and functions.	The implementation contains some significant errors. The source code is unclear, without comments and clearly named variables and functions.	The implementation is functional. The source code contains comments. Variables and functions have descriptive names.	The implementation demonstrates fluency in the programming language, frameworks and libraries, with well-chosen programming constructs. The code is well laid out and adheres rigorously to good coding and documentation practices. Comments are informative and at an appropriate level.	The implementation is a concise expression of the design. The implementation demonstrates fluency in the programming language, frameworks and libraries, with well-chosen programming constructs. The code is well laid out and adheres rigorously to good coding and documentation practices. Comments are	The implementation is a concise and elegant expression of the design. The implementation demonstrates fluency in the programming language, frameworks and libraries, with well-chosen programming constructs. The code is well laid out and adheres rigorously to good coding and documentation practices. Comments are

							informative and at an appropriate level.	informative and at an appropriate level.
Written/Report Elements	Overall	Communication of work is unclear and inappropriate to a defined audience and does not use appropriate strategies or media	Communication of work is unclear and inappropriate to a defined audience and does not consistently use appropriate strategies or media	Communication of the outcomes of their work is unclear and confused and does not consistently use appropriate strategies or media	The outcomes of their work are presented clearly and appropriately to a defined audience using a range of strategies and media	The outcomes of their work are presented confidently and coherently to a defined audience using a range of appropriately selected strategies and media	The outcomes of their work are presented convincingly and fluently to a defined audience using an interesting range of appropriately selected strategies and media	The outcomes of their work are presented creatively and persuasively to multiple audiences using a wide range of appropriately selected strategies and media
	Content	Most required elements of the Report are missing	Some required elements of the Report are missing	One or two required elements of the Report are missing	All of the required elements of the Report are included with some lacking in detail	All of the required elements of the Report are included in detail	All of the required elements of the Report are included in meticulous detail	All of the required elements of the Report are included in meticulous and authoritative detail
	Structure	The Report is devoid of meaningful structure	The Report is poorly structured and difficult to follow	The Report is lacking in structure and difficult to follow in places	The Report has a good basic structure and flow	The Report is well structured and flows effectively	The Report has a strong structure and flows very well	The Report has a strong structure and flows elegantly
	Referencing technique	Work does not implement the required referencing technique	Limited implementation of the required referencing technique	Work partially implements the required referencing technique	Work adequately implements the required referencing technique	Work thoroughly implements the required referencing technique	Work rigorously implements the required referencing technique	Work meticulous implements the required referencing technique

Table 1: Marking Descriptors