

Coursework Assignment Brief

Re-Assessment - Undergraduate

Academic Year 2021-22

Module Title:	Data Structures and Algorithms UG1	
Module Code:	CMP4272	
Assessment Title:	Report	
Assessment Identifier:	CWRK001	Weighting: 50%
School:	School of Computing and Digital Technology	
Module Co-ordinator:	Iain Rice	
Re-assessment hand in deadline date:	Monday 25 th July 2022 at 12pm (midday)	
Return of Feedback date and format	20 Working days after submission.	
Support available for students required to submit a re-assessment:	Timetabled support sessions will be arranged for the period immediately preceding the hand-in date	
NOTE:	At the re-assessment attempt the mark is capped and the maximum mark that can be achieved is 40%.	
Assessment Summary	<p>This assignment is a report in the form of a technical report. The technical report should:</p> <ol style="list-style-type: none"> 1. Explain an algorithm you have written and implemented which uses appropriate discrete data structures to solve a computational problem 2. Analyse the efficiency and correctness of your algorithm <p>The report should be submitted electronically to Moodle before the above deadline.</p>	

IMPORTANT STATEMENTS

Undergraduate Regulations

Your studies will be governed by the BCU Academic Regulations on Assessment, Progression and Awards. Copies of regulations can be found at <https://icity.bcu.ac.uk/Academic-Services/Information-for-Students/Academic-Regulations-2018-19>

For courses accredited by professional bodies such as the IET (Institution of Engineering and Technology) there are some exemptions from the standard regulations and these are detailed in your Programme Handbook

Cheating and Plagiarism

Both cheating and plagiarism are totally unacceptable and the University maintains a strict policy against them. It is YOUR responsibility to be aware of this policy and to act accordingly. Please refer to the Academic Registry Guidance at <https://icity.bcu.ac.uk/Academic-Registry/Information-for-Students/Assessment/Avoiding-Allegations-of-Cheating>

The basic principles are:

- Don't pass off anyone else's work as your own, including work from "essay banks". This is plagiarism and is viewed extremely seriously by the University.
- Don't submit a piece of work in whole or in part that has already been submitted for assessment elsewhere. This is called duplication and, like plagiarism, is viewed extremely seriously by the University.
- Always acknowledge all of the sources that you have used in your coursework assignment or project.
- If you are using the exact words of another person, always put them in quotation marks.
- Check that you know whether the coursework is to be produced individually or whether you can work with others.
- If you are doing group work, be sure about what you are supposed to do on your own.
- Never make up or falsify data to prove your point.
- Never allow others to copy your work.
- Never lend disks, memory sticks or copies of your coursework to any other student in the University; this may lead you being accused of collusion.

By submitting coursework, either physically or electronically, you are confirming that it is your own work (or, in the case of a group submission, that it is the result of joint work undertaken by members of the group that you represent) and that you have read and understand the University's guidance on plagiarism and cheating.

You should be aware that coursework may be submitted to an electronic detection system in order to help ascertain if any plagiarised material is present. You may check your own work prior to submission using Turnitin at the [Formative Moodle Site](#). If you have queries about what constitutes plagiarism, please speak to your module tutor or the Centre for Academic Success.

Electronic Submission of Work

It is your responsibility to ensure that work submitted in electronic format can be opened on a faculty computer and to check that any electronic submissions have been successfully uploaded. If it cannot be opened it will not be marked. Any required file formats will be specified in the assignment brief and failure to comply with these submission requirements will result in work not being marked. You must retain a copy of all electronic work you have submitted and re-submit if requested.

Learning Outcomes to be Assessed:

3. Write simple algorithms using appropriate discrete data structures to solve computational problems
4. Use appropriate methods to analyse the efficiency and correctness of algorithms.

Assessment Details:

Title: Report

Type: Coursework

Style: Technical Report

Throughout the semester you will have written technical reports explaining, presenting and analysing algorithms, including algorithms you have written yourself. You should have submitted a technical report as a first submission in May. Based on the feedback you were given you must re-work this report to meet the required learning outcomes. You **do not** need to start a new report if you submitted one previously. If you did not previously submit in May then ensure you use the technical report template provided on Moodle.

The report you submit should:

1. Present and explain an algorithm you have written and implemented, which uses appropriate discrete data structures to solve a computational problem
2. Analyse the efficiency and correctness of the algorithm you have implemented

Suggested topics and suitable algorithms for the technical report can be found on Moodle on the Assessment page. **You must choose from these suggested topics** unless you have prior approval from your module lecturer.

The technical report has a limit of 1500 words, excluding the report pro-forma formatting, any figures and program source code. Any program source code included in your technical report must be written in Python.

You must submit electronic copies of your technical report to Moodle, in either Microsoft Word or PDF format, before the corresponding submission deadline.

Rationale:

The assessment allows students to demonstrate their theoretical knowledge and practical skills developed throughout this module. The report holistically assesses the design, implementation and analysis of an algorithm in a format which is used widely across industry and academia both within and outside of computing. This will provide you with a valuable set of technical and writing skills which will be useful in your future career.

Spreading the assessment work over the whole semester reduces pressure on students towards the end of the semester, allowing time for feedback on earlier assessments to be received and incorporated into later assessments. You are encouraged to use other students' technical reports from classes as a learning resource, and to ask questions and give peer feedback.

Additional information:

You are encouraged to read and refer to information and code from other sources. However, information from other sources must be referenced appropriately using the BCU-Harvard referencing style.

Only use information from trustworthy sources; Wikipedia is a good source of information for data structures and algorithms and textbooks are very likely to be correct, but answers on StackOverflow may contain errors, for example.

For advice on writing style, referencing and academic skills, please make use of the Centre for Academic Success: <https://icity.bcu.ac.uk/celt/centre-for-academic-success>

Workload:

One technical report, 1500 words. The report should take approximately 50 hours of work, including background research and preparation.

Transferable skills:

At the point of completion of this technical report you should be able to:

- Communicate technical information
- Search for, read and refer to relevant and trustworthy sources
- Develop and implement a solution to a computational problem
- Identify potential issues within a program

Marking Criteria:

The report is marked in two distinct parts where 25% of the marks are given for the algorithm explanation and its' subsequent analysis.

Criteria	Write simple algorithms using appropriate discrete data structures to solve computational problems (LO3)	Use appropriate methods to analyse the efficiency and correctness of algorithms (LO4)
Weight	25%	25%
0 – 29%	<p>The algorithm does not solve an appropriate problem, or has serious errors. There is little or no discussion of how the algorithm works.</p> <p>No discrete data structure has been used, or the choice of data structure is inappropriate.</p>	<p>The analysis is limited and seriously flawed.</p>
30 – 39%	<p>The algorithm solves part of an appropriate problem. There may be substantial aspects of the problem which are not attempted or explained, or errors in the solution.</p> <p>The explanation is unclear or missing important details about how the algorithm works.</p>	<p>An attempt has been made to analyse an algorithm, but appropriate methods of analysis were not used, and the results of the analysis may be incorrect or meaningless.</p>
40 – 49%	<p>A rudimentary algorithm solving a basic problem. There may be some errors which could be corrected with further work.</p> <p>There is a limited discussion of how the algorithm works. The choice of data structure is inappropriate, or unjustified.</p>	<p>An attempt has been made to measure the running time of the algorithm for some inputs, but the methodology is unclear or the measurement may be inaccurate. There is a limited discussion of some other issues relating to efficiency.</p> <p>Analysis of the algorithm's correctness is vague, or not attempted.</p>
50 – 59%	<p>The algorithm solves an appropriate problem, though it may have minor errors or fail to account for special cases. There is an explanation of how the algorithm works.</p> <p>The choice of data structure may be inappropriate or poorly justified.</p>	<p>The running time of the algorithm has been measured accurately for an appropriate range of inputs, and the methodology has been explained. There is some discussion of other issues relating to efficiency.</p> <p>There is a basic or informal analysis of the algorithm's correctness.</p>

60 – 69%	<p>The algorithm correctly solves an appropriate problem. There is a clear explanation of how the algorithm works.</p> <p>At least one appropriate data structure has been used, and the choice has been adequately justified.</p>	<p>The efficiency of the algorithm has been accurately measured using an appropriate methodology, which has been explained. The measurements may include more than one metric.</p> <p>There is an analysis of the algorithm's correctness, which may specify pre- and post-conditions for part of the algorithm.</p>
70 – 79%	<p>The algorithm correctly solves a challenging problem. There is a clear explanation of how the algorithm works, and the explanation makes clear references to the relevant parts of the source code.</p> <p>Appropriate data structures have been used, and justification is given for each with reference to the specific problem.</p>	<p>The efficiency of the algorithm has been accurately measured using an appropriate methodology, with multiple metrics and a clear explanation. The asymptotic complexity of the algorithm is given. The efficiency may be compared with appropriate alternative algorithm(s).</p> <p>There is a formal analysis of the correctness of at least part of the algorithm.</p>
80 - 90%	<p>A well-designed algorithm which correctly solves a challenging problem. There is a clear, detailed explanation of how the algorithm works, with clear references to the relevant parts of the source code.</p> <p>Appropriate data structures have been used, and justification is given for each with reference to the specific problem.</p>	<p>The efficiency of the algorithm has been accurately measured using an appropriate methodology, with multiple metrics and a clear, detailed explanation. The asymptotic complexity of the algorithm is given. The efficiency has been compared with appropriate alternative algorithm(s).</p> <p>There is a detailed formal analysis of the correctness of the algorithm.</p>
90 – 100%	<p>An excellent algorithm written, explained and evaluated to the highest standards.</p>	<p>An excellent analysis of the efficiency, complexity and correctness of an algorithm, conducted and explained to the highest standards.</p>

Submission Details:

You must submit electronic copies of your technical report to Moodle, in either Microsoft Word or PDF format, before the corresponding submission deadlines.

Regulations:

If you submit an assessment late at the first attempt then you will be subject to one of the following penalties:

- if the submission is made **between 1 and 24 hours** after the published deadline the original mark awarded will be reduced by **5%**. For example, a mark of 60% will be reduced by 3% so that the mark that the student will receive is 57%. ;

- if the submission is made between **24 hours** and **one week (5 working days)** after the published deadline the original mark awarded will be reduced by 10%. For example, a mark of 60% will be reduced by 6% so that the mark the student will receive is 54%.
- **if the submission is made after 5 days following the deadline, your work will be deemed as a fail and returned to you unmarked.**

The reduction in the mark will not be applied in the following two cases:

- The mark is below the pass mark for the assessment. In this case the mark achieved by the student will stand
- Where a deduction will reduce the mark from a pass to a fail. In this case the mark awarded will be the threshold (i.e.40%)

Please note:

- **If you submit a re-assessment late then it will be deemed as a fail and returned to you unmarked.**

Feedback:

Marks and Feedback on your work will normally be provided within 20 working days of its submission deadline via Moodle.

Where to get help:

Students can get additional support from the library for searching for information and finding academic sources. See their iCity page for more information

<http://libanswers.bcu.ac.uk>

The Centre for Academic Success offers 1:1 advice and feedback on academic writing, referencing, study skills and maths/statistics/computing. See their iCity page for more information: <https://icity.bcu.ac.uk/celt/centre-for-academic-success>

See also the My Assignment Planner tool: <http://library.bcu.ac.uk/MAP2/freecalc-mail/>

Fit to Submit:

Are you ready to submit your assignment – review this assignment brief and consider whether you have met the criteria. Use the checklist provided below to ensure that you have done everything needed.

Checklist:

#	Item	Completed
1	I have read the assignment brief, its learning outcomes and marking criteria. I have clarified anything that I am not sure about it with the module leader.	
2	I have used the technical report template available on Moodle or an alternative which contains the relevant formatting and sections.	
3	I have used the spellchecker and proofread the report correcting errors several times.	
4	I have read carefully all the deadline dates for this submission.	
5	I have included all the required sections in my report.	
6	I have clearly shown which algorithm I have chosen to design, implement and analyse in my report.	
7	I have included a copy of my Python code in the appendix of my report.	
8	I have tested my algorithm implementation and made sure that it can be successfully executed.	
9	All work that is not my own is correctly referenced (BCU- Harvard Referencing) in the report.	
10	I have uploaded all the required submissions in plenty of time to the correct module Moodle submission point.	

Good Luck, Teaching Team 2021