# CMP1124M – Algorithms and Complexity – Assessment 1

| Learning Outcome | Criterion | Pass | 2:2 | 2:1 | 1st |
|---|---|---|---|---|---|
| [LO1] Understand the time and space efficiency of algorithms and how to calculate/estimate/evaluate and improve them.<br><br>[LO2] Determine an appropriate algorithmic approach to a problem.<br><br>[LO3] Ability to select from a range of possible options, to provide justification for that selection, and to implement the algorithm in a particular context. | Report (40%) | A basic report which explains the operation of the application as well as the selection of searching and sorting algorithms, but does not go into detail regarding the structure and/or the choices made.<br><br>The report may not contain the correct sections. | The report explains the operation and structure of the application.<br><br>The searching and sorting algorithms used in the code are assessed and discussed briefly.<br><br>The report is correctly formatted. | The report details the operation and structure of the application.<br><br>The searching and sorting algorithms used in the code are explained and justified in their selection.<br><br>The report is correctly formatted. | The report fully details the operation and structure of the application.<br><br>The searching and sorting algorithms used in the application are fully explained and assessed for their use in this application.<br><br>The report is correctly formatted.<br><br>Greater marks can be achieved in this section by the comprehensiveness of the report. |
| [LO1] Understand the time and space efficiency of algorithms and how to calculate/estimate/evaluate and improve them.<br><br>[LO2] Determine an appropriate algorithmic approach to a problem.<br><br>[LO3] Ability to select from a range of possible options, to | Implementation of the program, selection of sorting and searching algorithms (60%) | A basic solution that implements searching and sorting operations on an array and provides console output, which is appropriately formatted.<br><br>One searching and one sorting algorithm are used in the code. The arrays analysed are of 256 length, and the number of steps are shown. | A basic solution that successfully implements searching and sorting operations using the console screen and generates appropriately formatted output.<br><br>One searching and two sorting algorithms are used in the code. The arrays analysed are of 256 and 2048 length, and the number of steps are shown. | An implementation that includes a solution which shows searching and sorting operations on a console output screen.<br><br>Two searching and three sorting algorithms are used in the code. The arrays analysed are of 256 and 2048 length, and the number of steps are shown. | An implementation that includes a solution which shows searching and sorting operations on a console output screen.<br><br>Two searching and four sorting algorithms are used in the code. The arrays analysed are of 256 and 2048 length, and the number of steps are shown. |


| | | | | | |
|---|---|---|---|---|---|
| provide justification for that selection, and to implement the algorithm in a particular context. | | | Fair program structure and some code comments are present. | Clear program structure and appropriate comments are present. | Greater marks can be achieved by performing a comparative evaluation.<br><br>The program code is well-structured and commented with high standard naming conventions.<br><br>The best available searching and sorting algorithms are used to enhance and maximise the effectiveness of the application.<br><br>Greater marks can be achieved by completing the enhanced additions as set out in the brief. |
| **Weighting** | Weighting is indicated on individual criteria. | | | | |