



UNIVERSITY OF  
LINCOLN

## Lincoln School of Computer Science

### Assessment Item Briefing Document

**Title:** CMP1124M Algorithms and  
**Complexity:** Assessment 2

**Indicative Weighting:** 70%

#### Learning Outcomes:

**On successful completion of this assessment item a student will have demonstrated competence in the following areas:**

- [LO1] Understand the time and space efficiency of algorithms and how to calculate/estimate/evaluate and improve them.
- [LO2] Determine an appropriate algorithmic approach to a problem.
- [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.

#### Requirements

This assignment asks you to design and implement a Search and Sort application. In particular, you are to create a Console Application, which will help with the analysis of Weather Data. This assignment has mandatory (1-4) and additional (5-9 and a couple more) tasks, which will allow you to achieve higher marks.

A set of files is provided: “Year.txt”, “Month.txt”, “WS1\_TMax.txt”, “WS1\_TMin.txt”, “WS1\_AF.txt”, “WS1\_Rain.txt”, “WS1\_Sun.txt”, “WS2\_TMax.txt”, “WS2\_TMin.txt”, “WS2\_AF.txt”, “WS2\_Rain.txt” and “WS2\_Sun.txt”.

The files correspond to real averaged climate data of two Weather Stations WS1 (Lerwick), and WS2 (Ross on Wye) taken from the Met Office Historical Data web site. The “WS\*\_TMax.txt”, “WS\*\_TMin.txt”, “WS\*\_AF.txt”, “WS\*\_Rain.txt” and “WS\*\_Sun.txt”, correspond to the Mean Daily Temperature Maximum (in C), Mean Daily Temperature Minimum (in C), Days of Air Frost, Total Rainfall (in mm) and Total Sunshine duration (in hours), respectively.

The values of all the files are currently sorted in ascended order (i.e. oldest to newest) in respect to the content of the **Year** and **Month** files.

Initially, read the files “Year.txt”, “Month.txt”, “WS1\_TMax.txt”, “WS1\_TMin.txt”, “WS1\_AF.txt”, “WS1\_Rain.txt” and “WS1\_Sun.txt”.

Your Console Application should be able to provide the following functionality to the user:

1. Select which individual Array or String Array is to be analysed.
2. Sort in ascending or descending order and display the selected Array or String Array.
3. Search according to the content of the Year, and display ALL the corresponding values of Year, Month, WS1\_TMax, WS1\_TMin, WS1\_AF, WS1\_Rain and WS1\_Sun. If the value of

the Year does not exist then provide an error message.

4. Search according to the content of the Month, and display ALL the corresponding values of Year, Month, WS1\_TMax, WS1\_TMin, WS1\_AF, WS1\_Rain and WS1\_Sun. If the value of the Month does not exist then provide an error message.
5. When sorting in ascending or descending order the Month file you should display the months as they appear during the calendar year (i.e. in ascending order they should be: January, February, March, April, May, June, July, August, September, October, November and December), not as they appear in alphabetical order (i.e. in ascending order they are: April, August, December, February, January, July, June, March, May, November, October and September).
6. Find the maximum and minimum values of WS1\_TMax, WS1\_TMin, WS1\_AF, WS1\_Rain and WS1\_Sun. You should then display ALL the corresponding values of Year, Month, WS1\_TMax, WS1\_TMin, WS1\_AF, WS1\_Rain and WS1\_Sun.
7. Repeat Task 6, but for the median values of WS1\_TMax, WS1\_TMin, WS1\_AF, WS1\_Rain and WS1\_Sun.
8. Your Console Application should be in position to rearrange and display the content of all the other Arrays and String Arrays in respect to the one that is sorted.
9. For additional marks, your Console Application should be in position to input the files “WS2\_TMax.txt”, “WS2\_TMin.txt”, “WS2\_AF.txt”, “WS2\_Rain.txt” and “WS2\_Sun.txt”. Then Repeat Tasks 1 to 8 and display the corresponding values for both WS1 and WS2.

### **Enhancing your submission for top marks.**

Undertake a comparative evaluation for the all the searching and sorting tasks by using different Searching or Sorting algorithms. You should display the number of steps that each algorithm performed.

It is strongly recommended that you complete the console version first to ensure that you have a working application. For further marks in this assignment, you should rewrite your code so that it does not output to the console screen, but creates and opens a formatted web page, which displays in columns with the appropriate titles the information that you have to display.

A short (up to 1 minute) video of your application running should also be produced and uploaded to YouTube.

### **Useful Information**

This assessment is an individual piece of work. Your work must be presented according to the Lincoln School of Computer Science guidelines for the presentation of assessed written work.

Please make sure you have a clear understanding of the grading principles for this component as detailed in the accompanying Criterion Reference Grid.

If you are unsure about any aspect of this assessment component, please seek the advice of a member of the delivery team.

### **Submission Instructions**

The deadline for submission of this work is included in the School Submission dates on Blackboard.

You should submit your work as a single “.ZIP” file to the “*Assessment 2 – Source Code Upload*” section, and a report (in .PDF format only) to the “*Assessment 2 – Report Upload* section”. Use of other compression formats such as RAR files will be penalised.

- a) The ZIP file which is uploaded to *Assessment 2 – Source Code* should contain the project files, accompanying input files, any output files, executable and source code files for your application. The project should be able to be opened in Visual Studio (or any other IDE that you have used – *Mono* for example).
- b) The pdf report to be uploaded to *Assessment 2 – Report Upload* should contain:
  - a. **A contents page**
  - b. **A basic design for the application (1 page) including:**
    - i. A written description of the application.
    - ii. Comment about the implementation of tasks 3 to 9.
  - c. **A description of the algorithmic choices you made for the application (~1 page) including:**
    - i. Justification of selecting and implementing particular searching and sorting algorithms for your application.
    - ii. An evaluation of the time and space efficiency of the searching, and sorting parts, as well as your program overall.
  - d. **The URL of the video**
    - i. The short video should be captured with free software such as Screencast-O-Matic (<http://www.screencast-o-matic.com>). Download and install the software. Using the ‘free version’, follow the instructions to capture your video. When the video is complete (no greater than 1 minute in length), select ‘Upload to YouTube’. Upload your video as an ‘unlisted’ video (this allows us to see the video, but only when you tell us of its URL). Full, illustrated instructions for this process will also be available. The video should show your application running while you describe what you have done, how you have implemented it and how it works. You **will not** be assessed on the quality of the recording.
  - e. **A Reference list showing items you have used in your learning that are correctly cited in the body of the report**

*DO NOT include this briefing document with your submission.*