University of Central Lancashire UCLan **School of Psychology and Computer Science**	**UCLan Coursework Assessment Brief**	2020-2021
	Module Title: Programming	
	Module Code: CO1401	Level 4
	Word Search	This assessment is worth **100%** of the overall module mark

THE BRIEF/INSTRUCTIONS

This assignment is based on a word search puzzle: the sort of puzzle where you have a table of letters and you have to find the word in the table. Examples of this type of puzzle can be found at the following website: http://thewordsearch.com/.

This is an individual project and no group work is permitted.

You will be assessed on your implementation of the solution which **must be produced using C++**.

Do not diverge from the assignment specification. If you do not conform to the assignment specification then you will **lose marks**.

You may conduct your own research into topics which have not been explicitly taught within the module (this will be required for higher marks) however, you should include a justification for its use alongside links to any sources in your program comments. **Failure to do so may result in a plagiarism investigation. Only use concepts that you are confident you understand.**

Learning Outcomes
This assessment has been designed to assess the following learning outcomes:

- Apply the principles of programming.
- Design an appropriate solution for a given problem.
- Implement a readable and maintainable software solution of their own design.
- Evaluate the quality of his or her developed software.

Marking Scheme
Use the marking criteria provided to guide your design for your solution. Please follow this carefully.
Marking bands are indicative and can be overridden at the marker's discretion with justification.

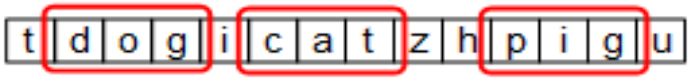
Pass Criteria:

- **10 marks:** The file 'search1.txt' is correctly read into an array. The file 'text1.txt' must also be read into a separate array.

- **20 marks:** Perform a comparison between the words from 'search1.txt' and the words from 'text1.txt' You need to find the occurrence of each word from 'search1.txt' in the string of characters from 'text1.txt'. Not all of the words occur in 'text1.txt'.
  For marks **up to 45%** you can use string pattern matching methods in the libraries, e.g. the *substr* and *find* methods of string. **Your grade will be capped at 45% if you choose to use these methods.**

- **10 marks:** Output the word, whether it has been found and, if found, the index of its location in the array. e.g.

  dog, Found, location 1
  pig, Found, location 10
  cat, Found, location 5
  rat, Not Found
  cow, Not Found



Continued on next page…

3rd Criteria:

- Up to **10 marks** will be given for following good programming practices:
- Your code must be properly indented and laid out so that it is readable.
  - Brackets must line up (and should normally be on a line of their own).
  - Indentation must be consistent.
  - Appropriate use of white space should be made.
  - Over-long lines of code or comments should be split up.
- You should have no 'magic numbers' but instead make proper use of constants.
- Variable names should be meaningful and no excessively long.
- Your code should be commented appropriated.

2:2 Criteria:

- In order to obtain a grade greater than 45% **you cannot use any pattern matching libraries**, this includes any of the STL string pattern matching methods and also the old style 'stdio.h' functions.

- **5 marks:** Developing your own pattern matching method.
  Hint: your pattern matching method needs to take into account the ends of the array – be careful about going out of bounds!

- **5 marks:** Use functions to `separate` your code into sensible, reusable parts.
  Comment these functions appropriately.

2:1 Criteria:

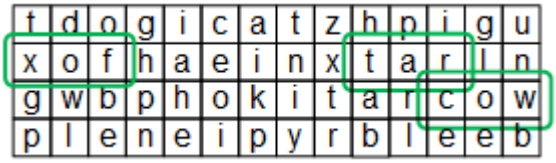
- **2 marks:** Read in the file 'text2D.txt' into a 2D array of size 4x14. You should do this in addition to the 1D array from 'text1.txt'

- **2 marks:** You should update your pattern matching method so it can identify words which have been written backwards as well as forwards, e.g. hidden in the text is the word 'rat' but it is written backwards 'tar'.

- **2 marks:** Write a filter method which **r**eplaces the words found by your pattern matching method with full stops, e.g. when you find the word 'cat' replace it with three full stops '…' (one for each character).

- **2 marks:** Write your filtered arrays (with the found words being replaced with full stops) to separate files named after the input files, e.g. the filtered text from 'text1.txt; is written to 'text1Filtered.txt', whilst 'text2.txt' is written to 'text2Dfiltered.txt'.
  You should ensure that your 2D array is written to file as a grid in the output file.

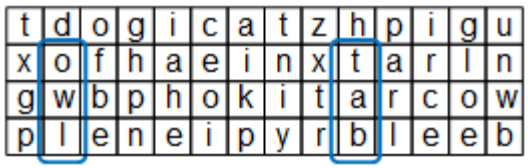
- **2 marks:** Update your screen output for your 2D array to include the line number of the word as well as the index of its location in the array, e.g.
  dog, Found, line 0, location 1
  pig, Found, line 0, location 10
  rat, Found, line 1, location 11



Continued on next page…

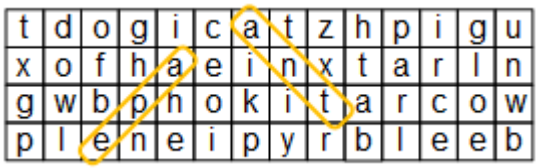
1st Criteria:

- **5 marks:** Read in the 'search2D.txt' file. This file contains additional words which will now overlap. Update your pattern matching and filtering methods to account for these overlapping words – you must take into account some of the characters will have already been replaced with full stops.
- **5 marks:** Update your pattern matching method to locate words which have been written **forwards and backwards,** as well as **up and down**.
  Remember to write your table to file.



High 1st Criteria:

- **10 marks:** Update your pattern matching function to also identify words **diagonally** in the table, as well as horizontally. Words may be written backwards, forwards but also up or down.
  For simplicity, diagonal words only occur downwards, but they can be on a diagonal down to the left or down to the right.



- **10 marks:** Update your pattern matching function to also identify orthogonal words. In this context, this refers to 'right-angled' words. These orthogonal words will only be 3 characters long and can only occur downwards however, they may occur downwards with one step to the left, or one step to the right.
  Hint: only a limited number of orthogonal patterns exist. It is possible to pre-generate the patterns and then use these to search, rather than attempt to calculate the search paths.

PREPARATION FOR THE ASSESSMENT

Before attempting this assessment, it is highly recommended that you revisit the "**Four L's**":

- **Lectures** – This includes the slides, notes and recording.
- **Lecture notes** – Any notes you took during the lectures.
- **Lab worksheets** – Read over all lab worksheets.
- **Lab projects** – Ensure all projects have at least stage one implemented.

Combined these provide all the necessary information for you to successfully complete this assessment. All resources are available on the CO1401 Blackboard area under *Module Materials*.

RELEASE DATES AND HAND IN DEADLINE

Assessment Release date: **05/07/2021** Assessment Deadline Date and time: **01/08/2021**

Please note that this is the final time you can submit – not the time to submit!
Your feedback/feed forward and mark for this assessment will be provided on **23/08/2021**

SUBMISSION DETAILS

Please take your time when reading this section, as this contains specific information on how you should submit your coursework.

- **You must use C++ to develop your solution.**
- Ensure your name and Student ID number (located on the back of your UCLan card) is stated at the top of your code.
- You should submit your **entire** project folder as a zip folder.
- **Do not only submit the .sln file as your code will not be included in your submission.**
- **Late submissions**: Except where an extension of the hand-in deadline date has been approved, work that is handed in within 5 working days late will receive a maximum mark of 40%. Work handed in later than this will receive 0%.
- **Academic Malpractice**: The consequences of academic malpractice in assessments are serious. This includes plagiarism, collusion and allowing other students to access your work. This will not be tolerated. Details surrounding the coursework regulations can be found in the University's "Assessment Handbook" located here.

Below are tips that you may find useful when working on this assessment:

- Do not leave this assessment to the last minute.
- If you have any questions regarding this coursework, ask the module leader or module tutors.
- Give yourself plenty of time to submit prior to the submission deadline.
- Use pen/cil and paper to work out the flow of your application.
- Revisit the "**Four L's**": **L**ectures, **L**ecture notes, **L**ab worksheets and **L**ab projects.

HELP AND SUPPORT

Please edit the below to describe how any questions arising from this assessment brief should be handled – e.g. tutorials in seminars, online forum, etc.

- Support will be provided via Microsoft Teams (CO1401 channel), and email. You will also have the opportunity to ask questions during lectures / labs.
- For support with using library resources, please contact our subject librarian Bob Frost RSFrost@uclan.ac.uk. You will find links to lots of useful resources in the My Library tab on Blackboard.
- If you have not yet made the university aware of any disability, specific learning difficulty, long-term health or mental health condition, please complete a Disclosure Form. The Inclusive Support team will then contact to discuss reasonable adjustments and support relating to any disability. For more information, visit the Inclusive Support site.
- To access mental health and wellbeing support, please complete our online referral form. Alternatively, you can email wellbeing@uclan.ac.uk, call 01772 893020 or visit our UCLan Wellbeing Service pages for more information.
- If you have any other query or require further support you can contact The <i>, The Student Information and Support Centre. Speak with us for advice on accessing all the University services as well as the Library services. Whatever your query, our expert staff will be able to help and support you. For more information, how to contact us and our opening hours visit Student Information and Support Centre.
- If you have any valid mitigating circumstances that mean you cannot meet an assessment submission deadline and you wish to request an extension, you will need to apply online prior to the deadline.