CSC 226 SUMMER 2018
ALGORITHMS AND DATA STRUCTURES II
ASSIGNMENT 4 - PROGRAM
UNIVERSITY OF VICTORIA

1 Programming Assignment (Optional)

This assignment is an optional programming assignment. If the grade on this assignment helps improve your overall assignment grade I will count it. Otherwise, I will not count it toward your overall assignment grade. The assignment is to implement an algorithm that finds the position of a string pattern (word) in an $n \times n$ character array (puzzle). A Java template has been provided containing two empty methods in the WordSearch class called WordSearch, which takes char[] word as an argument for any preprocessing needed on the pattern and a method called search which takes char[][] puzzle as an argument that searches for the pattern in the puzzle. (Note – this is how all three algorithms are implemented in the textbook.) The expected behavior of the class is as follows:

Input: A one dimensional character array of size m containg the pattern and a two dimensional character array of size $n \times n$ containing the puzzle.

Output: A Boolean value to determine if the word has been found or not.

There are also four global variables used for the coordinates of the starting point, (x_1, y_1), and ending point, (x_2, y_2), of the word in the puzzle. The main method has already been written and is used to read the word and puzzle from a text file and writing the output to the system.

You may add other methods to the WordSearch class or other classes to the WordSearch.java file but no other files please.

2 Input Format

The testing code in the main function of the template reads an input file containing all the necessary data. The input files have the following format: an integer, n, representing the dimension of the square grid of characters, followed by the word to search for, followed by n lines of n characters each, representing the puzzle. For example the input file, puzzle1.txt, contains the following:

```
10
rich
abcdefghij
klmnopqrst
uvwxyzrich
abcdefghij
klmnopqrst
uvwxyzabcd
efghijklmn
opqrstuvwx
yzabcdefgh
ijklmnopqr
```

3 Test Datasets

So far I have four, very egocentric, test files. I will include more as I make them.

4 Sample Run

The output of a model solution on the file above is given in the listing below.

```
Reading input values from puzzle1.txt.
Word: rich

Puzzle:
abcdefghij
klmnopqrst
uvwxyzrich
abcdefghij
klmnopqrst
uvwxyzabcd
efghijklmn
opqrstuvwx
yzabcdefgh
ijklmnopqr

Solution: Search word starts at coordinate (2,6) and ends at coordinate (2,9)
```

5 Evaluation Criteria

The programming assignment will be marked out of 25, based on a combination of automated testing and human inspection, based on the criteria in the table below.

Score (/25)	Description
0 - 5	Submission does not compile or does not conform to the provided template.
6 - 10	The program compiles and uses a brute-force algorithm and is incorrect.
10 - 15	The program compiles and either uses a brute-force algorithm and is correct.
15 – 20	The program compiles and uses an efficient algorithm (i.e. one from the text or equivalent) and is incorrect.
20 - 25	The program compiles and uses an efficient algorithm (i.e. one from the text or equivalent) and is correct.

To be properly tested, every submission must compile correctly as submitted, and must be based on the provided template. You may only submit one source file. **If your submission does not compile for any reason (even trivial mistakes like typos), or was not based on the template, it will receive at most 5 out of 25**. The best way to make sure your submission is correct is to download it from conneX after submitting and test it. You are not permitted to revise your submission after the due date, and late submissions will not be accepted, so you should ensure that you have submitted the correct version of your code before the due date. The conneX site will usually allow you to change your submission before the due date if you notice a mistake. After submitting your assignment, conneX will automatically send you a confirmation email. If you do not receive such an email, your submission was not received. If you have problems with the submission process, send an email to the instructor **before** the due date.