CS 170: Introduction to Computer Science I – Spring 2023
Homework Assignment #3
Due by Tuesday, March 14, 2023 at 4:00 pm

Submission instructions

Submit your assignment through the QTest system, using exam ID: **CS170.hw3**.
No email submissions are accepted. No late submissions are accepted.

General instructions and hints

In those problems asking you to write a method, always call the method several times to test that it works properly with a variety of different values for the parameters. Test it on more examples than the ones shown in this handout. Even if the problem asks you for just one method, you can always write additional helper methods to simplify and organize your code. Make sure you write comments to explain your code.

Comment requirements: Always comment the top of each method (what the method does, the meaning of the input parameters, the meaning of the output value). Write comments within the methods to explain the strategy you are using to solve the problem, and to clarify blocks of code that may be difficult to understand.

Problem 1: Mid semester survey (2 points)

Go to the following website and complete the survey.

https://docs.google.com/forms/d/e/1FAIpQLScUtIlzdIEmJttbtjsoyb901sAb-NAAn3t3a0jx79qSD_1S0w/viewform

The answers to the survey will not be graded: you will receive credit just for completing it.

Problem 1 rubric:
2 points for completing the survey.

Problem 2: Average length (5 points)

Write a method named `avgLength` which takes an array of Strings as an input parameter and returns a double. The method should calculate and return the average length of all the strings in the array.

Examples:
```
avgLength(new String[]{"Hello", "Q"}) returns 3.0
avgLength(new String[]{}) returns 0.0
avgLength(new String[]{"Hello", "Goodbye"}) returns 6.0
```

Rubric for problems 2-9:
Programs that do not compile get zero points.
+5 correct implementation (up to 3 points for a partially correct solution. Zero points if the solution is far from correct.)
-1 incorrect method signature (method name, number of parameters, types of parameters, or return type)
-1 if there are fewer than 2 test cases
-1 if there are no comments, insufficient comments, or bad usage of comments

Problem 3: Sum of products (5 points)

Write a method named `sumOfProds` which takes an array of integers as input and returns an integer. The return value is calculated by summing up the products between each element and the following one.

Examples:
```
sumOfProds(new int[]{3, 4, 5}) returns 32        (3*4 + 4*5)
sumOfProds(new int[]{4, 1, 19, -6}) returns -91  (4*1 + 1*19 + 19*(-6))
sumOfProds(new int[]{}) returns 0
sumOfProds(new int[]{5}) returns 5
```

Problem 4: Without duplicates (5 points)

Write a method named `withoutDuplicates` that takes an array of integers and returns a copy of the array without any duplicate elements. The order of the original elements should be preserved.

Examples:
```
withoutDuplicates(new int[]{1, 2, 3}) returns {1, 2, 3}
withoutDuplicates(new int[]{1, 2, 1, 1, 3, 2, 3}) returns {1, 2, 3}
```

Problem 5: Angle (5 points)

Write a method named `angle(double[] x, double[] y)` that takes two arrays of doubles, and returns the angle (in degrees, 0.0 <= angle < 180.0) between the two arrays. To calculate the angle, you can use the geometric definition of the dot product between the arrays:
https://en.wikipedia.org/wiki/Dot_product#Geometric_definition
If one array is shorter than the other, treat the missing elements as zeros. If the norm of any of the two arrays is zero, the angle is zero.

Examples:
```
angle(new double[]{1, 2, 3}, new double[]{4, 5}) returns 54.243...
angle(new double[]{0}, new double[]{4, 5}) returns 0.0
angle(new double[]{3, 0}, new double[]{0, 3}) returns 90.0
angle(new double[]{2, 1, 5, -4}, new double[]{6, 3, 15, -12}) returns 0.0
```

Problem 6: Reverse copy (5 points)

Write a method named `reverseCopy` that takes an array of integers and returns a copy of the array with its elements in reverse order. The original array is not modified.

Example:
```
reverseCopy(new int[]{1, 2, 3}) returns {3, 2, 1}
```

Problem 7: Reverse in place (5 points)

Write a method named `reverse` that takes an array of integers and reverses the order of its elements. The original array is modified, and the method doesn't return anything.

Problem 8: Tally vowels (5 points)

Write a method named `tally` that takes a String and returns an array of 5 integers containing the frequencies of the 5 vowels (a, e, i, o, u) in the input string. Uppercase and lowercase vowels are counted in the same way.

Example:
`tally("HEY! Apples and bananas!") will return: {5, 2, 0, 0, 0}`

Problem 9: Student averages (5 points)

Write a method named `studentAverages` which takes a 2D array of integers as input. Each column in the 2D array is an assignment and each row is composed of grades for a particular student (see below for an example). Your method should return an array of doubles representing the grades for each student. You may assume each assignment is scored out of 100 points.

	Quiz 1	Quiz 2	Quiz 3
Maggie Simpson	50	100	0
Lisa Simpson	100	100	80

`studentAverages(new int[][]{{50,100,0}, {100,100,80}}) returns {50.0, 93.33333}`

Problem 10: Swear word filter (8 points)

Write a method named `swearFilter(String text, String[] swear)` that takes two parameters: a String containing some text, and an array of Strings containing a list of "swear words". Your method will return a String containing the text contained in the first String, where each "swear word" is replaced by its first character, followed by a number of stars equal to its number of characters minus two, followed by its last character. For example, if the swear words are "duck", "ship", and "whole", and the text contains the following story:

```
A duck was sailing on a ship shipping whole wheat bread. Duck that SHIP!!!
```

Your method would return:

```
A d**k was sailing on a s**p s**pping w***e wheat bread. D**k that S**P!!!
```

Your method should recognize both uppercase and lowercase characters in a swear word. If a swear word is less than 3 characters long, the filter has no effect for that swear word. You **cannot** use any pre-made search/replace methods such as indexOf, replaceAll, etc.

Rubric:
Programs that do not compile get zero points.
+8 correct implementation (up to 6 points for a partially correct solution. Zero points if the solution is far from correct)
-2 for swear-word matching being case-sensitive
-2 for not maintaining original upper/lower-case
-2 for not matching strings that contain swear words as substrings
-1 incorrect method signature (method name, number of parameters, types of parameters, or return type)
-1 if there are fewer than 2 test cases
-1 if there are no comments, insufficient comments, or bad usage of comments

Bonus points: Early submission

If you submit the entire homework no later than 48 hours before the deadline, and the total score on the rest of this homework assignment is at least 20 points, you will receive 2 bonus points. The bonus points will be added to the total score of this homework assignment.

Good luck and have fun!