COMP30820 Java Programming (Conv)

Programming Test – Example Questions

1 Instructions

- Download file `test_examples.zip` from Moodle. Unzip the file and copy the package into Eclipse.
- Rename the **package** as follows: `test_examples_12345678` where `12345678` is your student number.
- Write your methods in the classes provided (one method per class). Each class contains a `main` method with some test cases included to enable you to test your code.
- At the end of the test, upload your solutions to Moodle:
  - Right-click on the package name and select *Export*. A dialogue box will appear – select *Next*.
  - You will then be asked to specify the name of the archive file which will be created – use the following naming convention: `test_examples_12345678.zip` where `12345678` is your student number. Select *Finish*.
  - Upload the zip file using the link under section `XXX` on Moodle. (*Note there is no need to upload solutions to these example questions.*)
- Please note:
  - This is an open book test. You may refer to the lecture notes, the textbook, and any notes you may have prepared yourself. Access to the Web is **not** permitted except for the Java API page and the course Moodle page.
  - **Important** — this is **not** a team/group exercise — each student must submit her/his own work. Please ask if you have any questions about this. See the course Moodle for information on the UCD plagiarism policy.

2 Marking Scheme

For each question, the following marking scheme will be used:

- 1 mark for each correct test case. Some test cases are given to you in the code you have downloaded from Moodle; a number of additional test cases will also be used to grade your solution.

3 Questions

Answer all questions.

Question 1

Write a method to replace every occurrence of a given character in a string. The first occurrence of the character should be replaced with `'1'`, the second occurrence with `'2'`, etc. Use the following method header:

```
public static String replace(String str, char ch)
```

Test cases:

- Invoking `replace("Mississippi", 'i')` should return: `"M1ss2ss3pp4"`.
- Invoking `replace("Hello World", 'w')` should return: `"Hello World"`.
- Invoking `replace("1234", '2')` should return: `"1134"`.

Write this method in file `Q1.java`.

Question 2

Write a method to find the **distinct** lowercase letters (`'a'` to `'z'`, inclusive) in a string. The method should return a reference to a character array containing the distinct lowercase letters. Use the following method header:

```
public static char[] getDistinctLCLetters(String str)
```

Test cases:

- Invoking `getDistinctLCLetters("")` should return: a reference to an empty character array (i.e. an array with zero elements).
- Invoking `getDistinctLCLetters("BEE")` should return: a reference to an empty character array (i.e. an array with zero elements).
- Invoking `getDistinctLCLetters("Bee")` should return: a reference to a character array with a single element: `'e'`.
- Invoking `getDistinctLCLetters("Hello World!")` should return: a reference to a character array with the following elements: `'d'`, `'e'`, `'l'`, `'o'`, `'r'` (note: the elements can be in any order in the array).

Write this method in file `Q2.java`.