Hack 8.0

Strings & String Processing
Computer Science I – Java

Department of Computer Science & Engineering
University of Nebraska–Lincoln

Introduction

Hack session activities are small weekly programming assignments intended to get you started on full programming assignments. Collaboration is allowed and, in fact, *highly encouraged*. You may start on the activity before your hack session, but during the hack session you must either be actively working on this activity or *helping others* work on the activity. You are graded using the same rubric as assignments so documentation, style, design and correctness are all important.

Exercises

To get more practice working with strings, you will write several methods that involve operations on strings. In particular, implement the following methods with the described behavior. You *must* use the given signatures.

1. Write a method that replaces instances of a given character with a different character in a string.

   ```
   public static String replaceChar(String s, char oldChar, char newChar)
   ```

   This method will replace any instance of the character stored in `oldChar` with the character stored in `newChar` in the string `s`.

2. Write a method that takes a string and removes all instances of a certain character from it.

   ```
   public static String removeChar(String s, char c)
   ```

3. Write a method that takes a string and splits it up to a *list* of strings. The split will be length-based: the function will take an integer n and will split the given string up into strings of length n. It is possible that the last string will not be of length n.

   ```
   public static List<String> lengthSplit(String s, int n)
   ```

   For example, if we pass `"Hello World, how are you?"` with $n = 3$ then it should return a list of size 9 containing the strings `"Hel"`, `"lo "`, `"Wor"`, `"ld,"`, `" ho"`, `"w a"`, `"re "`, `"you"`, `"?"`

Instructions

- Place all your methods into a source file named `StringUtils.java` along with proper documentation.
- In addition, create a main test driver program called `StringTester.java` that demonstrates at least 3 cases per function to verify their output. Hand in your tester.
- You are encouraged to collaborate with any number of students before, during, and after your scheduled hack session.
- You may (in fact are encouraged) to define any additional “helper” methods that may help you.
- Include the name(s) of everyone who worked together on this activity in your source file’s header.
- Turn in all of your files via webhandin, making sure that it runs and executes correctly in the webgrader. Each individual student will need to hand in their own copy and will receive their own individual grade.