CSCI 1301 – Programming Principles I
Georgia Southern University
Department of Computer Science
Fall 2024

Assignment 7
Point Value: 20 points
Due: Friday November 1, 2024, start of lab

NOTE: Gradescope will give input test cases for any problems that require it. Use the Scanner's nextDouble(), nextInt(), etc. methods discussed in class to prompt the user for input and Gradescope will act as the user when your program requests it. In other words, just code user input like normal—as if a human were going to test your program.

Description

Write a Java program that asks the user to enter a string, then a character. Your program should determine how many times that character occurs in the string using iteration (loops) as well as the reverse of the string.

Your program should ask the user for strings to test until they enter STOP in all capitals. If they enter STOP, exit **before** asking for a character to test. **Do NOT use break/continue in this project.**

Your project should contain at least three methods:

1. A method named countCharacters() that takes a String and a character as parameters (in that order). This method returns an integer that represents the number of times the character showed up in the string.
    a. The String parameter should be the string to test, the character is the character to test (case-sensitive), and the returned integer is the number of times the character was found in the passed string.
2. A method named printCount() that takes a String, a char, and an integer as parameters (in that order) and prints the number of occurrences to the console. This method does not return any values.
    a. The String parameter should be the string that was tested, the character is the character that was counted, and the returned integer is the number of times the character was found in the string.
    b. Do NOT call countCharacters() from inside this method. The count should be determined in the main method and passed as a parameter to printCount().
3. A method named reverseString() that takes a String as a parameter and returns a String that is the reverse of the passed String.
    a. The string that is reversed should be the user-entered string.
    b. Concatenation and iteration will be your friends in solving this problem.
    c. Think about the problem before implementing. Do not just Google a solution and use it. For example, many online solutions will use a StringBuilder or StringBuffer to

accomplish this, which we have not discussed, and you will not see until Chapter 10 in CSCI 1302. If your solution uses one, it is obvious that you have copy/pasted a solution from online.

HINTS

- Use the String's length to control any counting loops.
- Refresh yourself on how to compare strings, get individual characters, and get string input from the console; all in Chapter 4.
- Do not worry about trying to error check the user-entered String or character, assume that they do it correctly each time. All Gradescope tests will provide proper input as long as the code uses token-based (NOT line-based) input.

Expected Output/Sample Run

```
Enter a string: Georgia
Enter a character: g
g occurs in Georgia 1 times
The reverse of Georgia is aigroeG

Enter a string: Mississippi
Enter a character: s
s occurs in Mississippi 4 times
The reverse of Mississippi is ippississiM

Enter a string: igloo
Enter a character: z
z occurs in igloo 0 times
The reverse of igloo is oolgi

Enter a string: STOP
```

Code

Use the provided template for this assignment. Make any necessary modifications to classes and class headers to complete this assignment.

```
public class PAssign07 {
	public static void main(String[] args)  {
		// add your code here

	}
}
```

Deliverables

Name your program PAssign07.java. Programming Assignment 7 is to be individual work. Submit the program by the specified due date. Submit each file to its corresponding assignment on Gradescope.

See and follow the Programming Assignment Format document for submission requirements.

Use a utility similar to https://www.diffchecker.com/ and the Expected Output to compare your program's output with the requested output as well as the unit tests provided within Gradescope. Programming is in the details, so double check punctuation, spacing, and case if your output does not match. When copying and pasting, be aware that Microsoft Word sometimes replaces normal quotes with Smart Quotes, which may need to be edited.

Last modified: October 21, 2024