

Exercise 6.1 War and Peace

This week introduced 2D array, char and String along with some methods to manipulate above types. Make sure you understand the difference between primitive data type and reference data type. The exercises of this week are combination of for loop, array and String. Once you are familiar with methods, you should find it easy.

Problem Description:

Write a Java method that given a string text, returns true if the string "war" and "peace" appear the same number of times in text. All input letters are in lowercase.

Key words:

a string text(no patterns)

the same number of times(count occurrences)

lowercase(no need to transform)

Test cases :

"war" and "peace" will be underlined

warAndPeace("there was never a good war, or a bad peace") → true

one war and one peace

warAndPeace("war what is it good for") → false

one war

warAndPeace("asdhfjwarsajkfhkpeaceadkjfhkjswarsklfkjwarfsd")-> false

Solution Design

We need to count the occurrences of String "war" and String "peace". Notice that I use String instead of word. That is because the input of this method is merely a String(a sequence of characters) not plain text. So, there is no grammar structure guaranteed and it could be a sequence of random characters with no spaces while "war" or "peace" can occur at any place.

Next I will introduce two different design strategies.

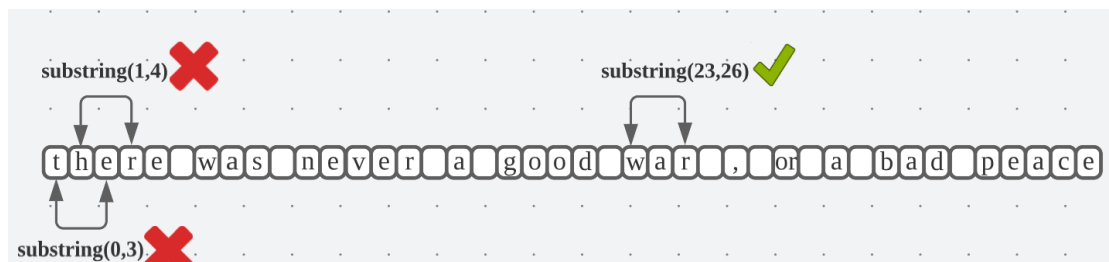
First is "forloop+substring+equals" searching strategy.

Second is "whileloop+indexOf" searching strategy.

Please make sure you understand the methods I listed in both strategies.

"forloop+substring+equals"

To search "war", we need to compare any three consecutive characters in String with "war". That is comparing any `substring(i, i + 3)` for all *i* between 0 and the length of text with "war".



Once you find one "war", you can increment the variable used to count the occurrence of "war". Similar way to search "peace". One loop to count "war" and one loop to count "peace". Make sure you set the condition correctly.

"whileloop+indexOf"

A useful method wrapped within String is `indexOf`. Notice this method is overloading and has four different ways to pass parameter. We are going to use

`indexOf(String str)` to search the first *str* and `indexOf(String str, int fromIndex)` to find left *str*. Please make sure you understand this method. it return the index of first occurrence of *str* starting from *fromIndex*.

In above case, `text.indexOf("war")` and `text.indexOf("war", 2)` return 23. And `text.indexOf("war", 24)` returns -1 as there is no "war" after index 24.

Since we don't know how many loops the search will take. We use while loop.

Once we found a "war" at index *i*. we can set *fromIndex* to *i*+2. And next round search from *i*+3.

One loop to count "war" and one loop to count "peace".

Check implementations