Programming Assignment 3

Getting Started

This assignment emphasizes the use of string methods, and is based on the material in Sections 1.3 *The class String* and Chapter 2 of the textbook. Review in-class examples and Handouts 2 and 3.

Programming Project

WordContext: Reveal a word in context **worth 10 points**

For this project you'll write a program that will ask the user to enter text and a phrase from the text and will display where the phrase appears in the text. Two sample interactions below show how the program is supposed to run:

> Please enter some text: **IBM Watson passed its first test on Jeopardy! in February 2011, but the real test will be in applying the underlying systems, data management and analytics technology in business and across different industries.**
> Please enter a word or a phrase to find in context: **technology**
>
> The phrase appears in the following context: [analytics TECHNOLOGY in]

Here is another interaction:

> Please enter some text: **Bullfighting is the only art in which the artist is in danger of death and in which the degree of brilliance in the performance is left to the fighter's honor. (Hemingway)**
> Please enter a word or a phrase to find in context: **in danger**
>
> The phrase appears in the following context: [is IN DANGER of]

More specifically, the program should:

1. Ask the user to enter one line of text. **In that text the words must be separated by no more than a single space.**
2. Ask the user to enter a phrase (note that it may contain one or more words, as shown above).
3. Display the phrase in the context in which it occurs for the first time in the text as the word right before the phrase, the phrase (in upper case letters), the word right after the phrase.

For this week's project, you can make the following important **simplifying assumption:** the user will enter a phrase that does indeed occur within the text, moreover, the phrase will not be in the beginning and not in the very end of the text. This means that if your program does not work for the cases in which the text starts or ends with the searched phrase, that's ok. Note also that the program should only display the *first* occurrence of the phrase.

Other requirements on the program: to receive full credit, your program should

- display the phrase within the context in all upper case letters, while the rest of the text should be displayed unchanged;
- have exactly one space between the words in the output;
- allow case-insensitive searching, for instance, user entering **test** should cause the same output as **TEST**.

Hints: To avoid having to enter long text every time you test your program, at the first stages of developing your program hardcode the text by assiging it to a variable directly in the code. Later on, when you've tested the program with different phrases, you can update it to read the text from the user.

Grading. The grading schema is as follows

- 1 point for correctly accepting the input (i.e. the test and the phrase),
- 4 points for correctly displaying the phrase in context,
- 1 point for displaying the phrase within the context in upper case letters,
- 2 points for case insensitive search,
- 2 point for good programming style. The **style requirements** are as follows:

Include *introductory comments* listing the name of the author, the date the file was created and a description of the purpose of the program. Also provide comments within the program code for any part of code that may be difficult to follow and would benefit from explanatory text.

Use variable names that reflect the purpose of the variable, and add comments if more information would be helpful (for example, if a variable has a valid data range, that should be noted in a comment).

Indent code within curly braces for improved readability (Eclipse can do most of the indenting automatically, just use `Source-> Correct Indentation` or `CTRL-i`. Proper indentation makes it easier for you to debug, because then the layout of the code matches the functionality.

Good luck and Happy Programming!