CSE 1241 - COMPUTER PROGRAMMING I
Programming Assignment # 3
DUE DATE: 11/12/2023 - 23:00 (No extension)

In this homework, you will write a program which will take an input, if the word is either “**exit**” or “**quit**”, your program should terminate; otherwise, your program will execute the related methods based on the user choice.

Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice:
```

1. You should implement the following method for option 1 that takes an input string and an input character and returns the number of occurrences of the character in the given input string.

   - **`public static int numOfChars(String str, char ch)`**

   Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 1
Enter an input string: Hello
Enter an input char: e
The number of e in Hello is 1.
```

2. You should implement the following method for option 2 that takes an input sentence as a string and prints the words inside it. A word is a sequence of characters without any whitespaces and punctuation marks. The only punctuation marks that you have to consider are:

, (comma)	. (period)
! (exclamation mark)	? (question mark)
_ (underscore)	- (hyphen)
( ) (parentheses)	white-space

- **public static void printWords(String str)**

Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 2
Enter an input string: !!!The fourth CSE1141 homework is due on 28-Nov-2018 (Wednesday).!!!
The output is:
The
fourth
CSE1141
homework
is
due
on
28
Nov
2018
Wednesday
```

3. You should implement the following method for option 3 that finds a substring in a given string and delete it’s first occurrence if the type argument is 0 or delete it’s all occurrences if the type argument is 1. The method should return the updated string.

    - **public static String delete(String str, String subStr, int type)**

    Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 3
Enter an input string: I AM DOING MY HOMEWORK AT MY HOME.
Enter a substring: HOME
Enter a type: 0
I AM DOING MY WORK AT MY HOME.
```

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 3
Enter an input string: This homework is so easy!
Enter a substring: is
Enter a type: 1
Th homework so easy!
```

4. You should implement the following method for option 4 that finds a substring (subStr1) in the given string (str) and replaces it's all occurrences with the given substring as the third argument (subStr2).

    - **public static String replace(String str, String subStr1, String subStr2)**

    Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 4
Enter an input string: Introduction to Programming
Enter the first substring: o
Enter the second substring: x
Intrxductixn tx Prxgramming
```

5. You should implement the following method for option 5 that sorts the characters given in a string according to the sorting method provided by type argument. If type is equal to 0, sort the characters of the string in order of ASCII value from low to high and return the new string. If type is equal to 1, sort the characters of the string in this order: the lower-case letters, the upper-case letters, the digits and the other characters (these groups should internally be stored in order of ASCII value from low to high) and return the new string.

    - **public static String sortChars(String str, int type)**

Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 5
Enter an input string: CSe_1141
Enter a type: 0
1114CS_e

Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 5
Enter an input string: CSe_1141
Enter a type: 1
eCS1114_
```

6. You should implement the following method for option 6 that finds and returns a hash code of a given string based on an integer value.

   - **`public static int hashCode(String str, int b)`**

The hash code of a string is calculated based on the following formula:

$$hashCode(s,\ b) = s_0 * b^{(n-1)} + s_1 * b^{(n-2)} + \ldots + s_{n-1}$$

where s_i is **`s.charAt(i)`** and **`b`** is an integer given as the second argument.

Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 6
Enter an input string: ABC
Enter a value for b: 31
The hash code for ABC is 64578.
```

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: 6
Enter an input string: Welcome
Enter a value for b: 31
The hash code for Welcome is -1397214398.
```

Note: This computation can cause an overflow for long strings, but it is not problem.

7. Main method

To implement this program, you need to create an infinite loop in the "***main***" method; in each iteration of the loop, first ask the user to enter an option and request the inputs related with the option. Then, invoke related methods with the required arguments based on the option selected. Lastly, print the output of the selected option. If the input is "**exit**" or "**quit**", your program should terminate.

Example:

```
Welcome to our String Analyzer Program.
     1. Count number of chars
     2. Print the words in a sentence
     3. Delete substring
     4. Replace substring
     5. Sort characters
     6. Hash code of a string
Please enter your menu choice: exit
Program ends. Bye :)
```

Important Notes:

- ***You have to implement and use the methods listed above.***
- ***You can add more methods to your code if you need more.***
- ***You should print the values with two digits after the decimal point.***
- ***Your programs should execute correctly for different test cases.***
- ***Selected parts of your submissions will be graded! If you only submit the implementation of a single question, you might get a grade of 0!***

Submission Instructions

Please zip and submit your files using filename YourNumberHW3.zip
(ex: 150713852HW3.zip) to Canvas system (under Assignments tab).
Your zip file should contain the following 2 files:

1. Java source code (Pro_150713852.java)
2. Java class file (Pro_150713852.class)

Please use the ***default package*** *in Eclipse IDE* for the assignments. Otherwise, the submitted code may not be compiled on another computer

Submission Notes:

1. Write a comment at the beginning of your program to explain the purpose of the program. Write your name and student ID as a comment. Include necessary comments to explain your actions.
2. Select meaningful names for your variables and class name.
3. You are allowed to use the materials that you have learned in lectures & labs.
4. Do not use things that you did not learn in the course.
5. **Program submissions** should be done through the Canvas class page, under the assignments tab. Do not send program submissions through e-mail. E-mail attachments will not be accepted as valid submissions.
6. You are responsible for making sure you are turning in the right file, and that it is not corrupted in anyway. We will not allow resubmissions if you turn in the wrong file, even if you can prove that you have not modified the file after the deadline.
7. In case of any form of **copying and cheating** on solutions, all parts will get **ZERO** grade. You should submit your own work. In case of any forms of cheating or copying, both giver and receiver are equally culpable and suffer equal penalties.
   All types of plagiarism will result in zero grade from the homework.
8. No late submission will be accepted.