Due: Thursday, March 5th by the start of class. Late turn-in is March 10th (Spring Break!) so plan accordingly.

Be sure to include the appropriate collaboration statement as comments at the top of your submitted program. Failure to do so will result in a 10 point deduction.

Move into your CS170 directory:
\[ cd \text{cs170} \]

Make a directory for hw3:
\[ mkdir \text{hw3} \]

And move into that directory:
\[ cd \text{hw3} \]

You can now start up gedit:
\[ gedit \& \]

Name your program \texttt{Homework3.java}. Write the following methods. Follow the specified naming conventions EXACTLY (capitalization, parameter order, etc).

Strategy: I HIGHLY recommend you write one method at a time, test it, and verify it works before moving on to another method. While it is not required, I also recommend you write some of these methods with while loops and some with for loops to get practice with both of them.

1. Write a method named \texttt{countMyZs} which takes a String as an input parameter. The method should count the number of Z's (both upper case or lower case) in the input string, and if they exist, replace them with dashes. This method should \textbf{return an integer} representing the number of Z's and \textbf{print the new string} with the dashes to the terminal. You may not use any String class methods except \texttt{length} and \texttt{charAt}.

   Examples:
   \begin{itemize}
   \item \texttt{countMyZs("ZzABCzZZabc")} returns 5 and prints "--ABC---abc"
   \item \texttt{countMyZs("HELLO")} returns 0 and prints "HELLO"
   \item \texttt{countMyZs("ZZZZ")} returns 4 and prints "----"
   \end{itemize}

2. Write a method named \texttt{hungrySevenMonster}. This monster only eats multiples of seven. Have this method take an integer as input and return the number of multiples of 7 which are less than or equal to the input parameter. If the input parameter is negative, your method should return -1. Examples:

   \begin{itemize}
   \item \texttt{hungrySevenMonster(45)} returns 6 since there are 6 multiples of 7 (7, 14, 21, 28, 35, 42) which are less than or equal to 45
   \item \texttt{hungrySevenMonster(6)} returns 0
   \item \texttt{hungrySevenMonster(-100)} returns -1
   \end{itemize}

3. You are helping an English professor write a program to perform some basic literary analysis. Write a function named \texttt{avgLength} which takes a String as input. This function should count the words in the input string and compute the average length of the words in the sentence. For the purposes of this method, you should ignore punctuation. You may assume that words are
always separated by a single space. Your method should return a double indicating the average word length in a given String. You may not use any String class methods except length and charAt. Don't worry if you answer has more/less significant digits.

avgLength("In a hole in the ground, there lived a...") returns 3.625
(average of 2, 1, 4, 2, 3, 6, 5, 5, and 1)

avgLength("@#$% $%^ *&!!") returns 0.0 (all words are punctuation only)

4. Write a function named countdown which takes 3 integer parameters. In order, they are: the number to start counting at, the number to count to and the amount to count by (decrement). You method should print the countdown for the given parameters. For example:

countdown(9, 0, 3) would print:
9
6
3
0

countdown(8, 4, 2) would print:
8
6
4

countdown(10, 5, 2) would print:
Impossible!

Your code must account for situations which are not possible. For example it is not possible to countdown from 10 to 5 by subtracting 2. So a call to countdown(10, 5, 2) is not valid. Likewise, it is not possible to count down if the 2nd parameter is greater than or equal to the first parameter. If the countdown is not possible given the input values, the function should print out “Impossible!”.

Hint: there is simple math you can do to check if a combination of input values is valid.

5. Write a function named printFibonacci. A Fibonacci sequence is produced by adding the two preceding numbers together to produce the next integer. Your method should take in 2 integer parameters and produce the next 10 numbers in a Fibonacci sequence which starts with the addition of the 2 parameters. Your function should work for any integer inputs. Examples:

printFibonacci(1, 2) prints
3, 5, 8, 13, 21, 34, 55, 89, 144, 233

printFibonacci(1, 9) prints
10, 19, 29, 48, 77, 125, 202, 327, 529, 856

Submission;
Make sure to include comments and your collaboration statement. Then, from your cs170/hw3 directory, run the command:

/home/cs170001/turnin hw3 Homework3.java

You will see a message containing “+++ALLOWED” if your submission was successful. If you had previously submitted, you will be asked to verify you wish to overwrite a previous submission.