

1. Define a method which returns the sum of three rounded numbers. If the right most digit of the number is less than 5, then round off it's value to the previous multiple of 10 otherwise if the right most digit of the number is greater or equal to 5, then round off to the next multiple of 10.

Write the method with the following specifications:

Name of method `sumOfRoundedValues()` // which accepts three integer value as argument and return the sum of three rounded numbers.

Arguments: three argument of type integer

Return Type: an integer value

Example

If $a = 23$, $b = 34$, $c = 66$

$20 + 30 + 70 = 120$

If $a = 23$, $b = 37$, $c = 55$

$20 + 40 + 60 = 120$

Specifications: The value returned by the method `sumOfRoundedValues()` is determined by the following rules:

If any of the given number is negative or zero, return -1.

If any of the given numbers right most digit is of the number is less than 5, then round off its value to the previous multiple of 10 otherwise if the right most digit of the number is greater or equal to 5, then round off to the next multiple of 10.

2. Create a method that accepts a number and modifies it such that the each of the digit in the newly formed number is equal to the difference between two consecutive digits in the original number. The digit in the units place can be left as it is.

Note: Take the absolute value of the difference. Ex: $6-8 = 2$

Method Name	<code>modifyNumber</code>
Algorithm	<ol style="list-style-type: none">1. Convert number into String2. Extract each char using <code>charAt</code> method Convert char to int and find the difference3. Create new <code>StringBuffer</code> object and keep adding the difference4. Finally convert <code>StringBuffer</code> to int
Argument	<code>int number1</code>
Return Type	<code>int</code>
For example.	Input: 45862 Output:13242

3. You are asked to create an application for registering the details of jobseeker. The requirement is:

Username should always end with _job and there should be atleast minimum of 8 characters to the left of _job. Write a function to validate the same. Return true in case the validation is passed. In case of validation failure return false.

Method Name	validateUserName
Argument	String userName
Return Type	boolean

4. Create a method which can perform a particular String operation based on the user's choice. The method should accept the String object and the user's choice and return the output of the operation.

Options are

A: Add the String to itself

*B: Replace alternate positions with **

C: Remove duplicate characters in the String

D: Change alternate characters to upper case

Method Name	changeString
Argument	String string, char ch
Return Type	String

5. Create a method that accepts a String and checks if it is a positive string. A string is considered a positive string, if on moving from left to right each character in the String comes after the previous characters in the Alphabetical order.

For Example

ANT is a positive String (Since T comes after N and N comes after A)

APPLE is not positive since L comes before P in the alphabetical order.

The method should return true if the entered string is positive

Method Name	checkPositive
Argument	String
Return Type	boolean
Algorithm	Step 1: Convert to Char array. Step 2: Iterate through array, subtract 1st two characters (A-N). This will give the ASCII difference Step 3: If result is negative, then return false and break. Else continue to next loop

6. Write a method which accepts a String and moves all the lower case 'a' to the beginning of the String.

Method Name	rearrangeCharacters
Argument	String
Return Type	String
Algorithm	1. Convert the string to a character array

	<p>2. Create a StringBuffer object</p> <p>3. Create a variable(count) to store the number of 'a' present</p> <p>4. Iterate over the character array and if the character is 'a' increment count for 'a' else add the character to the StringBuffer object.</p> <p>5. Finally insert the count number of 'a' to the beginning of the StringBuffer object</p>
--	---

7. Write a method which accepts an email in a form of String and validate that email values based on the following rules

1) There should be only one @

2) There should be one com

3) There should be only 5 characters between @ and .

Method Name	validateEmail
Argument	String
Return Type	Boolean

8. Write a method which accepts two string parameters, find out the count the number of occurrence of second parameter in the first parameter

Method Name	countWord
Argument	String and String
Return Type	Int
Example	<p>First parameter: hello world, world is wonderful</p> <p>Second parameter: world</p> <p>Output:2</p>