**Assignment 2**

**Due date: 2-12-2022**

**(Question 1 to 10 are as assignment and remaining are for practice Perform these without using string standard functions)**

**1.** Write a C++ program to reverse a given string.   
Example:  
Sample Input: Pakistan  
Sample Output: natsikap

**2.** Write a C++ program to change every letter in a given string with the letter following it in the alphabet (ie. a becomes b, p becomes q, z becomes a). Example:  
Sample Input: Pakistan Zindabad  
Sample Output: Qbkistbn aindbbbd

**3.** Write a C++ program to capitalize the first letter of each word of a given string. Words must be separated by only one space.

Example:  
Sample Input: cpp string exercises  
Sample Output: Cpp String Exercises

**4.**Write a C++ program to find the largest word in a given string.

Example:  
Sample Input: C++ is a general-purpose programming language.  
Sample Output: programming

**5.** Write a C++ program to sort characters (numbers and punctuation symbols are not included) in a string.    
Example:  
Sample Input: python  
Sample Output: hnopty

**6.** Write a C++ program to check whether the characters e and g are separated by exactly 2 places anywhere in a given string at least once.

Example:  
Sample Input: eagerer  
Sample Output: eagerer -> 1

**7.** Write a C++ program to count all the vowels in a given string.

Example:  
Sample Input: eagerer  
Sample output: number of vowels -> 4

**8.** Write a C++ program to count all the words in a given string.    
Example:  
Sample Input: Python  
Sample Output: number of words -> 1

**9.** Write a C++ program to check whether two characters present equally in a given string.

Example:  
Sample Input: aabcdeef  
Sample Output: True

**10.** Write a C++ program to check if a given string is a Palindrome or not.    
A palindrome is a word, number, phrase, or other sequence of characters which reads the same backward as forward, such as madam, racecar.  
Example:  
Sample Input: madam  
Sample Output: True

**11.** Write a C++ program to find a word in a given string which has the highest number of repeated letters.   
Example:  
Sample Input: Print a welcome text in a separate line.  
Sample Output: Word which has the highest number of repeated letters. separate

**12.** Write a C++ program to insert a dash character (-) between two odd numbers in a given string of numbers.    
Example:  
Sample Input: 1345789  
Sample Output: Result-> 1-345-789

**13.** Write a C++ program to change the case (lower to upper and upper to lower cases) of each character of a given string.    
Example:  
Sample Input: Pythpn  
Sample Output: pYTHON

**14.** Write a C++ program to find the numbers in a given string and calculate the sum of all numbers.    
Example:  
Sample Input: Pakistan12 from 2008  
Sample Output: Sum of the numbers: 2011

**15.** Write a C++ program to convert a given non-negative integer to English words.    
Example:  
Sample Input: 12  
Sample Output: Twelve  
Sample Input: 29  
Sample Output: Twenty Nine

**16.** Write a C++ program to find the longest common prefix from a given array of strings.    
Example:  
The longest common prefix is: Pa  
The longest common prefix is: J  
The longest common prefix is:

**17.** Write a C++ program to find all combinations of well-formed brackets from a given paris of parentheses.    
Example:  
n = 2  
[[]] [][]  
n = 3  
[[]] [][] [[[]]] [[][]] [[]][] [][[]] [][][]

**18.** Write a C++ program to find the length of the longest valid (correct-formed) parentheses substring of a given string.   
Example:  
Original Parentheses string: [[]  
Length of longest parentheses: 2  
Original Parentheses string: [[]]]  
Length of longest parentheses: 4  
Original Parentheses string: ]]]][[[[  
Length of longest parentheses: 0

**19.** Write a C++ program to reverse only the vowels of a given string.    
A vowel is a syllabic speech sound pronounced without any stricture in the vocal tract. Vowels are one of the two principal classes of speech sounds, the other being the consonant.  
Example:  
Original string: Pakistan  
After reversing the vowels of the said string: Pikastin

Original string: Python  
After reversing the vowels of the said string: Python  
Original string: Hello  
After reversing the vowels of the said string: Holle  
Original string: USA  
After reversing the vowels of the said string: ASU

**20.** Write a C++ program to find the length of the longest palindrome in a given string (uppercase or lowercase letters).   
Original string: adcdcdy  
Length of the longest palindrome of the said string: 5  
Original string: aaa  
Length of the longest palindrome of the said string: 3  
Original string: aa  
Length of the longest palindrome of the said string: 2  
Original string: abddddeeff  
Length of the longest palindrome of the said string: 9  
Original string: PYTHON  
Length of the longest palindrome of the said string: 1

**21.** Write a C++ program to check whether a given string is a subsequence of another given string. Return 1 for true and 0 for false.    
Example:  
word1: apple  
subse1: apl  
Is subse1 is the subsequence of word1? 1  
word2: apple  
subse2: ppe  
Is subse2 is the subsequence of word2? 1  
word3: ACGGTGTCGTGCTATGCTGATGCTGACTTATATGCTA  
subse3: CGTTCGGCTATGCTTCTACTTATTCTA  
Is subse3 is the subsequence of word3? 1  
word4: CGTTCGGCTATCGTACGTTCTATTCTATGATTTCTAA  
subse4: CGTTCGGCTATGCZTTCTACTTATTCTA  
Is subse4 is the subsequence of word4? 0

**22.** Write a C++ program to remove all special characters from a given string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original string: abcd $ js# @acde$  
New string after removing the special characters from the said string:  
abcd js acde  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-22.php)

**23.** Write a C++ program that count the number of unique characters of two given strings. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original Strings:  
String1: Python  
String2: Java  
Total number of unique characters of the said two strings: 9

**24.** Write a C++ program to count number of duplicate characters in a given string.    
Example:  
Original String:  
Total number of unique characters of the said two strings.  
Number of duplicate characters in the said string: 36

**25.** Write a C++ program to find the longest sequence of consecutive ones in a given binary string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original Binary String:  
1100110001  
Longest sequence of consecutive ones of the said binary string:  
11  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-25.php)

**26.** Write a C++ program to check a given string is a title cased string or not. Return "True" if the string is title cased string otherwise "False". [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original String:  
The Quick Brown Fox.  
Check the said string is a title cased string or not!  
True  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-26.php)

**27.** Write a C++ program to insert a space when a lower character follows an upper character in a given string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original String:  
TheQuickBrownFox.  
Insert white spaces between lower and uppercase Letters in the said string:  
The Quick Brown Fox.  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-27.php)

**28.** Write a C++ program to extract the first specified number of vowels from a given string. If the specified number is less than number of vowels present in the string then display "n is less than number of vowels present in the string". [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Input a string: Input a number:  
Extract the first n number of vowels from the said string:  
n is less than number of vowels present in the string!  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-28.php)

**29.** Write a C++ program to print a given integer with commas as thousands separators. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Input a number:  
Print the said integer with commas as thousands separators:  
5,000  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-29.php)

**30.** Write a C++ program to identify the missing letter in a given string (list of alphabets). The method returns, "There is no missing letter!" if no letters are missing from the string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original string: abcdef  
Identify the missing letter in the said string:  
There is no missing letter!  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-30.php)

**31.** Write a C++ program to check if a given string contains only uppercase or only lowercase letters. Return "True" if the string is uppercase or lowercase, otherwise "False". [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original string: ABCDEF  
Check whether the said string is uppercase or lowercase: True  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-31.php)

**32.** Write a C++ program that takes a string and reverses the words of three or more lengths in a string. Return the new string. As input characters, only spaces and letters are permitted. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
Example:  
Original string: The quick brown fox jumps over the lazy dog  
Reverse the words of three or more lengths of the said string:  
ehT kciuq nworb xof spmuj revo eht yzal god  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-32.php)

**33.** Write a C++ program to verify that the letters in the second string appear in the first string. Return true otherwise false. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
**Test Data:**  
("CPP", "Cpp") -> false  
("Java", "Ja") -> true  
("Check first string", "sifC") ->true  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-33.php)

**34.** Write a C++ program that removes a specific word from a given string. Return the new string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
**Test Data:**  
("Exercises Practice Solution", "Solution") -> "Exercises Practice"  
("Exercises Practice Solution", "Practice ") -> "Exercises Solution"  
("Exercises Practice Solution", " Solution") -> " Practice Solution"  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-34.php)

**35.** Write a C++ program to reverse all words that have odd lengths in a string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
**Test Data:**  
("Exercises Practice Solution" ) -> "sesicrexE Practice Solution"  
("The quick brown fox jumps over the lazy dog") -> "ehT kciuq nworb xof spmuj over eht lazy dog."  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-35.php)

**36.** Write a C++ program to check whether there are two consecutive (following each other continuously.), identical letters in a given string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
**Test Data:**  
("Exercises") -> 0  
("Yellow") -> 1  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-36.php)

**37.** Write a C++ program that counts the number of instances of a certain character in a given string. [.](https://www.w3resource.com/cpp-exercises/string/index.php#EDITOR)  
**Test Data:**  
("Exercises", "e") -> 2  
("Compilation Time", "i") -> 3  
[.](https://www.w3resource.com/cpp-exercises/string/cpp-string-exercise-37.php)

**38.** Write a C++ program that removes a specific character from a given string. Return the new string.    
**Test Data:**  
("Filename", "e") -> "Filnam"  
("Compilation Time", "i") -> "Complaton Tme"

**39.** Write a C++ program that checks whether a given string contains unique characters or not. Return true if the string contains unique characters otherwise false.   
**Test Data:**  
("Filename") -> 0  
("abc") -> 1

**40.** For two given strings, str1 and str2, write a C++ program to select only the characters that are lowercase in the other string at the same position. Return the said character(s) as a single string.   
**Test Data:**  
("Java", "jscript") -> "scr"  
("jScript", "Java") -> "Jva"  
("cpp", "c++") -> "c++"

**41.** Write a C++ program that finds the position of the second occurrence of a string in another string. If the Substring does not appear at least twice return -**Test Data:**  
("the qu qu", "qu") -> 7  
("theququ", "qu") -> 5  
("thequ", "qu") -> -1

**42.** Write a C++ program that alternates the case of each letter in a given string of letters.    
Pattern: First lowercase letter then uppercase letter and so on.  
**Test Data:**  
("JavaScript") -> "jAvAsCrIpT"  
("Python") -> "pYtHoN"  
("C++") -> "c++"