INDIAN INSTITUTE OF TECHNOLOGY ROORKEE
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CSC-101 PROGRAMMING WITH C AND C++

Assignment 06 Autumn 2023

1. In the world of Bitonia, people communicate using binary strings. However, the Bitonians have a unique practice: they believe that communication should be "balanced". A balanced binary string for them is a string in which the number of 0's is equal to the number of 1's. But, Bitonians have an advanced notion of balance. They are interested in the largest balanced contiguous sub-string which is also a palindrome.

   You need to create a program that, given an input binary string, identifies the longest balanced palindromic sub-string. If multiple such sub-strings exist with the same length, the program should select the one that appears first. Furthermore, your program should also calculate the number of 1's in this sub-string.

   For instance, if the input string is "0010110100101", the longest balanced palindromic sub-string is "01011010" and it has 4 ones.

   Input Constraints:

   The length of the binary string: $1 \leq$ 'n' $\leq 10^5$.

   Example:

   Input:

   0010110100101

   Output:

   Longest Balanced Palindrome: 01011010

   Number of 1's: 4

   Hint: Use two pointers technique combined with a palindrome checker to efficiently solve this problem.

2. In the land of Matrixia, matrices are the foundation of their language. People use matrices to write everything from their names to their history. In this culture, a "spiral pattern" holds a significant value and is considered a symbol of enlightenment.

You are to write a C++ program that, given a matrix of integers, extracts the spiral pattern of the matrix starting from the top-left corner and moving clockwise. This extracted spiral should be presented as a single row matrix.

Additionally, for a twist, your program should calculate the product of every third element of the spiral.

For instance, if the input matrix is:

```
1  2  3  4
5  6  7  8
9 10 11 12
```

The spiral pattern will be: '1, 2, 3, 4, 8, 12, 11, 10, 9, 5, 6, 7'.

Thus, every third element is: '3, 12, 9, 7'. And their product is '2268'.

Input Constraints:

- The matrix will have a size of 'm x n', where $1 \leq$ 'm', 'n' ≤ 100.
- Each integer in the matrix: $1 \leq$ 'value' $\leq 10^4$.

Example:

Input:

```
1 2 3 4
5 6 7 8
9 10 11 12
```

Output:

Spiral Pattern: 1 2 3 4 8 12 11 10 9 5 6 7

Product of every third element: 2268

3. The citizens of Graphonia are fascinated by graphs, especially the ones that resemble mountains. They recently discovered a unique type of graph which they named "Peak Valley Graphs." A Peak Valley Graph is a bar graph where bars rise and fall alternately, forming peaks and valleys. A peak is formed when a bar is taller than its adjacent bars, and a valley is formed when a bar is shorter than its adjacent bars.

Write a C++ program that, given a sequence of integers representing the heights of bars in a graph, determines if the graph is a Peak Valley Graph. If it is, the program should also return the height of the highest peak.

Input Constraints:

- The sequence will contain 'n' integers, where $3 \leq$ 'n' ≤ 100.
- Each integer in the sequence: $1 \leq$ 'value' $\leq 10^5$.

Output Constraints:

If the graph is not a Peak Valley Graph, print "Not a Peak Valley Graph". If it is, print the height of the highest peak.

Examples:

Input:

5 7 4 6 3

Output:

Highest Peak: 7

Input:

2 5 5 8 6

Output:

Not a Peak Valley Graph

4. In the magical land of Listopia, people communicate using a special kind of language called "String Magic." In this language, the strength of a word is determined by the distinct consecutive characters in it. For instance, the word "aabb" has a strength of 2 because there are two distinct consecutive characters ("aa" and "bb"). However, there's a twist! If a word contains the same character consecutively more than twice, its strength gets reduced by the number of times the character repeats consecutively beyond the second occurrence.

   Write a C++ program to compute the strength of a given word in String Magic.

Input Constraints:

The word will contain 'n' lowercase English characters, where $1 \leq$ 'n' ≤ 100.

Output Constraints:

Print the strength of the word.

Examples:

Input:

aabbcc

Output:

Strength: 6

Input:

aaabbb

Output:

Strength: 0

5. In the land of Stringtopia, sentences are constructed in a unique manner. The citizens have a tradition where every sentence they speak or write ends with the same word it starts with. However, being a unique land, they also impose an additional condition: the last word should be a reverse of the first word.

   Your task is to write a C++ program that reads a sentence and determines whether the sentence is valid according to Stringtopian traditions. Additionally, if the sentence is valid, the program should return the number of vowels in the first (and last) word.

   Input Constraints:

   - The sentence will contain between 1 and 200 characters.
   - Words in the sentence are separated by single spaces.
   - Only lowercase letters and spaces are used.

   Output Constraints:

   If the sentence does not meet the Stringtopian tradition, print "Invalid Stringtopian Sentence". If it does, print "Valid with X vowels", where X is the number of vowels in the first and last word.

Examples:

Input:

apple elppa

Output:

Valid with 2 vowels

Input:

hello olleh how are you doing today

Output:

Invalid Stringtopian Sentence

6. Welcome to Vectorland, a peculiar dimension where vectors come to life. In Vectorland, there's a sport called "Vector Weaving." In this sport, two vectors are taken, and they are woven together to create an "Interleaved Vector." Your challenge, should you accept, is to write a C++ program that accomplishes the following:

   Specifications:

   1. Take two integer vectors as input, 'v1' and 'v2'. Their lengths are 'n' and 'm' respectively.

   2. Generate an "Interleaved Vector" where elements from 'v1' and 'v2' alternate. If one vector runs out of elements before the other, append the remaining elements of the longer vector to the result.

   3. Compute the sum of the squares of the difference between every consecutive pair of numbers in the "Interleaved Vector".

   4. Print the resulting "Interleaved Vector" and the computed sum.

   Constraints:

   - $1 \leq$ 'n', 'm' ≤ 100
   - The vectors only contain integers between 1 and 1000, inclusive.

Example:

Input:

v1: 1 3 5

v2: 2 4

Output:

Interleaved Vector: 1 2 3 4 5

Sum: 4

7. Deep within a mythical forest, there exists an enchanted path that is said to have been paved by a whimsical wizard. The path consists of large stones, each carved with a single unique word. Legend says that the wizard, fond of riddles, enchanted the path such that those who tread upon it can move ahead only if the current stone's word is an anagram of the previous stone's word.

   Your mission, should you choose to accept it, is to devise a C++ program that guides travelers through the path. Given a sequence of words, your program should determine if a traveler can successfully traverse from the start to the end of the path.

   Input:

   A list of 'n' words ($1 \leq n \leq 100$), each word contains between 1 and 20 lowercase letters.

   Output:

   - Print "Mystical Path Cleared" if one can traverse the path from the beginning to the end based on the wizard's enchantment.
   - Else, print "Path Blocked".

   Example:

   Input:

   listen silent enlist slinte tinsel

   Output:

   Mystical Path Cleared

Input:

hello olleh ollehh

Output:

Path Blocked

8. In the forgotten city of Eldoria, three tall towers stand. Each tower is decorated with ancient runes. Long ago, scholars decoded the significance of these runes: each rune represents a unique positive integer. Over time, a puzzling legend about these towers emerged.

   The legend states that there exists a secret vault beneath the city which can only be accessed when the runes on these towers are reordered to create three sequences of decreasing integers. Moreover, no tower should have a greater total rune value than the tallest tower's original total rune value.

   Your challenge is to write a C++ program to determine if the runes on these towers can be rearranged to unlock the secret vault.

   Input:

   Three sequences of positive integers representing the rune values on each tower. The sequences are separated by commas. Each sequence contains between 1 and 50 runes (integers). Each rune value is between 1 and 1000.

   Output:

   - "Vault Unlocked" if the runes can be reordered to fulfill the legend's requirements.
   - Otherwise, "Vault Sealed".

   Example:

   Input:

   3 2 1, 4 3 2, 6 5 4

   Output:

   Vault Unlocked

   Input:

   7 6, 8 1 2, 5 4 3

   Output:

   Vault Sealed

9. The tech wizard of Centropolis has hidden a message within a string by encoding it using a unique method. Every alphabetic character is replaced with its reverse in the alphabet (e.g., 'a' becomes 'z', 'b' becomes 'y', etc.), while digits are replaced by their complement to 9 (e.g., '1' becomes '8', '0' becomes '9', etc.). Punctuation and whitespace remain unchanged.

   Your objective is to construct a C++ program that unveils the concealed message.

   Guidelines:

   1. The application should prompt the user to input a coded string.
   2. The script must decipher the string and display the uncovered message.

10. In the mystical realm of Echoland, reflections aren't just limited to mirrors. In this world, numbers mirror each other. To be an Echolandic number, a number must look the same when its digits are reversed, just like palindromic numbers. But there's a twist: the number, when reversed, should be added to the original number to form a new number. If this new number is a palindrome, then the original number is said to be an "Echolandic number." Write a C++ program to determine if a given number is Echolandic. If it is, the program should also display the palindromic number obtained after addition.

    Input Constraints: The number will be a positive integer up to 10^6.

    Output Constraints:

    - If the number is Echolandic, print "Echolandic Number: X", where X is the palindromic number after addition.
    - If not, print "Not an Echolandic Number".

    Examples:

    Input:

    123

    Output:

    Echolandic Number: 444

    Input:

    58

    Output:

    Not an Echolandic Number

Submission Link