PIC-10A: Homework 3

Due 10/22/21 11:59pm via CCLE

Problem 1:

Write a program that reads four integers and prints "two pairs" if the input consists of two matching pairs (in some order), print "one pair" if the input consists of one matching pair (in some order), and "There is no pair" otherwise.

Here is a sample dialog, the user input is in blue:

Please enter 4 integers (separated by spaces): 1 2 2 1
Two pairs.

Please enter 4 integers (separated by spaces): 1 2 2 3
One pair.

Please enter 4 integers (separated by spaces): 2 2 2 2
Two pairs.

Please enter 4 integers (separated by spaces): 1 2 2 2
One pair.

Please enter 4 integers (separated by spaces): 1 2 3 4
There is no pair.

You can assume the user's input is always valid. **Do not use `for` loop**

Submit your file as *hw3_1.cpp*

Problem 2:

Write a program that prompts the user to provide a single character from the alphabet. Print `Vowel` or `Consonant`, depending on the user input. If the user input is not a letter (between `a` and `z`, or `A` and `Z`), or is a string of length > 1, print an error message.

Hint: use string comparison and ASCII table. **Do not use `array` or `for` loop**

Submit your file as *hw3_2.cpp*

Problem 3

Write a program that prompts the user to enter a strictly positive integer. Then print out its binary representation.

Here is a sample dialog

Please enter a strictly positive integer: 123
The binary representation is: 1111011

In this program, we can assume the user always enters a strictly positive integer.

Hint:

- While loop(s) is enough to solve this problem.
- You must use integer (either of type int or long long) to store the input
- You are allowed to use the function `to_string(int)` to convert a number to a string. `to_string(int)` is a function of the `string` library.

This is a procedure to convert a number from decimal to binary: The example is number 49:

- Divide 49 by 2, we obtain quotient = 24, remainder = **1**
- Divide 24 by 2, we obtain quotient = 12, remainder = **0**
- Divide 12 by 2, we obtain quotient = 6, remainder = **0**
- Divide 6   by 2, we obtain quotient = 3, remainder = **0**
- Divide 3   by 2, we obtain quotient = 1, remainder = **1**
- Divide 1   by 2, we obtain quotient = 0, remainder = **1**

Collecting all the remainders of this procedure from the bottom up, we get the binary representation of 49: **110001**

Submit your file as *hw3_3.cpp*