Week 7 Lab Worksheet

Matthew Ung

1. Write a power function that takes a number to a specific power. The function should have two parameters, an int and a double (to represent the number). The function should return the computed value.
2. A year with 366 days is called a leap year. A year is a leap year if it is divisible by 4 (for example, 1980), except that it is not a leap year if it is divisible by 100 (for example, 1900). However, a leap year is divisible by 400, (for example, 2000). Write a function that determines if a year is a leap year. Invoke the function created, and allow the user to enter as many years as they wish.
3. Write a function which finds the factorial of a number entered by the user. Check for all types of conditions.
4. Implement the swap function. The swap function does not return any value, and uses two doubles as input.
5. Create a simple Palindrome checker program. The program should allow the user to enter a string and check whether the given string is a palindrome or not. Only digits and alphabets should be considered while checking for palindromes -- any other characters are to be ignored. The palindrome check should be created as a function.
6. Using number 4, add a static variable that counts how many times the swap function has been used. Output the total times the variable has been used each time it is called.
7. Write a program that tells what coins to give out for any amount of change from 1 cent to 99 cents. For example, if the amount is 86 cents, the output would be something like the following:

86 cents can be given as

3 quarter(s) 1 dime(s) and 1 penny(pennies)

Use coin denominations of 25 cents (quarters), 10 cents (dimes), and 1 cent (pennies). Do not use nickel and half-dollar coins. Your program will use the following function (among others):

void compute\_coins(int coin\_value, int& num, int& amount\_left);

// Precondition: 0 < coin\_value < 100; 0 <= amount\_left < 100.

// Postcondition: num has been set equal to the maximum number of coins of

// denomination coin\_value cents that can be obtained from amount\_left. Additionally,

// amount\_left has been decreased by the value of the coins, that is, decreased by

// num \* coin\_value

For example, suppose the value of the variable amount\_left is 86. Then, after the following call, the value of number will be 3 and the value of amount\_left will be 11 (because if you take 3 quarters from 86 cents, that leaves 11 cents):

compute\_coins(25, number, amount\_left);

Include a loop that lets the user repeat this computation for new input values until the user says he or she wants to end the program.