Practice 4

Problem 1: Pass by Value vs. Pass by Reference

- Type in the following two user-defined functions and call them in the main function as described below.

```
void exchange1(int num1, int num2)
{
    int hold = num1;
    num1 = num2;
    num2 = hold;
}
void exchange2(int& num1, int& num2)
{
    int hold = num1;
    num1 = num2;
    num2 = hold;
}
```

```
int main() {
    int num1 = 1;
    int num2 = 2;

    cout << "num1 and num2 before exchange1: " << num1 << "," << num2 << endl;
    exchange1(num1, num2);
    cout << "num1 and num2 after exchange1: " << num1 << "," << num2 << endl;
    cout << "num1 and num2 before exchange2: " << num1 << "," << num2 << endl;
    exchange2(num1, num2);
    cout << "num1 and num2 after exchange2: " << num1 << "," << num2 << endl;
    cout << "[your answer_____]" << endl;
    return 0;
}
```

- Both exchange1 and exchange2 functions try to exchange the values of num1 and num2, but only one works correctly.
- Fill the blank in the main function with your answer why only one function works but the other does not.
- Hint: refer to Figure 4-15 and 16 on page 142-143

Problem 2: Pass by Reference

- Recall the problem 3 in practice 3 where you already wrote a program that calculates the mean of three numbers.
- The prototype of the original "mean" function was

```
float mean(int number1, int number2, int number3),
```

  - where the function returned the average of the three integer parameters.
- Modify the original program so that now the function takes one more parameter in which the average is saved. Use "pass by reference" technique.
- The prototype of the modifies function would be

```
void mean(int number1, int number2, int number3, float& answer),
```

  - Note that now the return type of the function is void.
- In the main function, define a float-type variable and pass it to the function for storing the result, call the function, and then print the value of it.

Problem 3: Default Parameter

- Type in the following program code.

```
#include <iostream>
using namespace std;
long fun(long x = 1, long y = 1);
int main() {
    int a = 2, b = 3, c = 4, d = 5;
    long r = fun(a,b);
    long s = fun(fun(a,b),fun(c));
    long t = fun(c+3,fun(d+7,b));
    long u = fun(fun(fun(fun())));
    long v = fun(fun(b+c,fun(c+3)),fun(a+d,fun()));
    cout << r << ' ' << s << ' ' << t << ' ' << u << ' ' << v << endl;
    return 0;
}
long fun(long x, long y) {
    long t = x + y;
    long z = x * y;
    return (t + z);
}
```

- The "fun" function takes two long-type parameters and return a long-type integer after some basic calculation.
- Both the parameters have default values.
- Execute the program and inspect the results. Then, try to figure out how the default parameters are selected.
- For each line in the main function, add your comment how the result is obtained.
  Ex) r = fun(fun(a,b),fun(c)); //fun(fun(2,3),fun(4,1))

Problem 4: Abs, Floor, and Ceil Functions

- Write a program that first gets one float-type number input from a user and then prints the absolute, ceiling, floor, and square root value of the number.
- Use the standard library functions introduced in Section 4-5.
- The result should be

```
Enter a floating number: 3.1415
Absolute value: 3.1415
Ceiling value: 4
Floor value: 3
Square root value: 1.77243
계속하려면 아무 키나 누르십시오 . . .
```

Problem 5: Random Number Generation 1

- Write a program that conducts the following operations
  - Prompt a user to enter an integer number.
  - Generate a random number ranged from 3 to 37.
  - Calculate the multiplication of the integer number and the random number.
  - Calculate random number to the power of the integer number ($randomNumber^{intNumber}$). Use the "pow" function on the page 149.
  - Print the results of the calculation as follows.

```
Enter an integer number: 3
Random number: 21
Multiplication: 63
Random number to the power of 3: 9261
```

Problem 6: Random Number Generation 2

- Generate three random numbers ranged from 1 to 100.
- Calculate the average of the three numbers using the function you made in Problem 2.
- Calculate the variance and the standard deviation of the three numbers. Use your code for the assignment 2.
- Generate another three random numbers (1~100) and calculate the average, variance, and standard deviation.
- Calculate the difference of the two averages, variances, and standard deviations. Print the differences.

Problem 7: Fruit Shop – Pass by Reference

- Recall the fruit shop program from the previous practice.
- Modify your unitTotal function so as to use pass by reference technique.
- Prototype of the modified unitTotal function is

  void unitTotal(int unitPrice, int numberPurchase, int& total)
  - For example, calling unitTotal(5000,4,total) saves 20000 in the variable total.
  - You may need to call this function five times to calculate the sub-total price of each fruit.
  - Print the same report as shown here →

```
How many apples did you buy: 2
How many bananas did you buy: 2
How many peaches did you buy: 3
How many grapes did you buy: 1
How many melons did you buy: 4
Price for apple: 2 * 5000 = 10000
Price for banana: 2 * 1500 = 3000
Price for peach: 3 * 3500 = 10500
Price for grape: 1 * 6000 = 6000
Price for melon: 4 * 1200 = 4800
Total price of your purchase: 34300
```