

LAB 6.1 Functions with No Parameters

Copy and paste the following program into Visual Studio IDE.

```
// This program prints the proverb:
// "Now is the time for all good men to come to the aid of their party"
// in a function (procedure) called writeProverb that is called by the main function

#include <iostream>
using namespace std;

void writeProverb();    // This is the prototype for the writeProverb function

int main()
{
    // Fill in the code to call the writeProverb function

    return 0;
}

// *****
//          writeProverb
//
// task:      This function prints a proverb
// data in:    none
// data out:   no actual parameter altered
//
// *****

// Fill in the function heading and the body of the function that will print
// to the screen the proverb listed in the comments at the beginning of the program
```

Exercise 1: Fill in the code (places in bold) so that the program will print out the proverb listed in the comments at the beginning of the program. The proverb will be printed by the function which is called by the `main` function.

Print out the program and the running result for Exercise 1, and hand them in with the rest of the lab.

LAB 6.2 Introduction to Pass by Value

Copy and paste the following program into Visual Studio IDE.

```
// This program will allow the user to input from the keyboard whether the
// last word to the following proverb should be party or country:
// "Now is the time for all good men to come to the aid of their _____"
// Inputting a 1 will use the word party. Any other number will use the word country.

#include <iostream>
#include <string>
using namespace std;

// Fill in the prototype of the function writeProverb.

int main ()
{
    int wordCode;

    cout << "Given the phrase:" << endl;
    cout << "Now is the time for all good men to come to the aid of their ____"
        << endl;
    cout << "Input a 1 if you want the sentence to be finished with party" << endl;
    cout << "Input any other number for the word country" << endl;
    cout << "Please input your choice now" << endl;
    cin >> wordCode;
    cout << endl;

    writeProverb(wordCode);

    return 0;
}

// *****
// writeProverb
//
// task:      This function prints a proverb. The function takes a number
//            from the call. If that number is a 1 it prints "Now is the time
//            for all good men to come to the aid of their party."
//            Otherwise, it prints "Now is the time for all good men
//            to come to the aid of their country."
// data in:   code for ending word of proverb (integer)
// data out:  no actual parameter altered
// *****

void writeProverb (int number)
{
    // Fill in the body of the function to accomplish what is described above
}
```

Exercise 1: Some people know this proverb as “Now is the time for all good men to come to the aid of their country” while others heard it as “Now is the time for all good men to come to the aid of their party.” This program will allow the user to choose which way they want it printed. Fill in the blanks of the program to accomplish what is described in the program comments. What happens if you inadvertently enter a float such as -3.97?

Exercise 2: Change the program so that the main program will check the validity of the input. If the user enters “1”, the main program will print “party” at the end. If the user enters “2”, it will print “country”. If the user enters any other number, the main program will print an error message and repeatedly prompt the user to enter a new choice until it is valid.

Sample Run:

Given the phrase:

Now is the time for all good men to come to the aid of their _____

Input a 1 if you want the sentence to be finished with party

Input a 2 if you want the sentence to be finished with country

Please input your choice now

4

I'm sorry but that is an incorrect choice; Please input a 1 or 2

2

Now is the time for all good men to come to the aid of their country

Print out the program and the running result for Exercise 2. Hand them in with the rest of the lab.

Note: the running result should include at least one invalid input and one valid input.

Lab 6.3 Introduction to Pass by Reference

Copy and paste the following program into Visual Studio IDE.

```
// This program takes two numbers (payRate & hours) and multiplies them to
// get grosspay.
// It then calculates net pay by subtracting 15%

#include <iostream>
#include <iomanip>
using namespace std;

//Function prototypes
void printDescription();
void computePaycheck(float, int, float&, float&);

int main()
{
    float payRate;
    float grossPay;
    float netPay;
    int hours;

    cout << setprecision(2) << fixed;
    cout << "Welcome to the Pay Roll Program" << endl;
    printDescription();    // Call to Description function
    cout << "Please input the pay per hour" << endl;
    cin >> payRate;
    cout << endl << "Please input the number of hours worked" << endl;
    cin >> hours;
    cout << endl << endl;
    computePaycheck(payRate, hours, grossPay, netPay);

    // Fill in the code to output grossPay

    cout << "The net pay is $" << netPay << endl;
    cout << "We hope you enjoyed this program" << endl;

    return 0;
}
```

```

// *****
// printDescription
//
// task:      This function prints a program description
// data in:   none
// data out:  no actual parameter altered
//
// *****

void printDescription() // The function heading
{
    cout << "*****" << endl << endl;
    cout << "This program takes two numbers (payRate & hours)" << endl;
    cout << "and multiplies them to get gross pay " << endl;
    cout << "it then calculates net pay by subtracting 15%" << endl;
    cout << "*****" << endl << endl;
}

// *****
// computePaycheck
//
// task: This function takes rate and time and multiplies them to
// get gross pay and then finds net pay by subtracting 15%.
// data in: pay rate and time in hours worked
// data out: the gross and net pay
//
// *****

void computePaycheck(float rate, int time, float& gross, float& net)
{
    // Fill in the code to find gross pay and net pay
}

```

Exercise 1: Fill in the code (places in bold) and note that the function `computePaycheck` determines the net pay by subtracting 15% from the gross pay. Both `gross` and `net` are returned to the `main()` function where those values are printed.

Exercise 2: Compile and run your program with the following data and make sure you get the output shown.

```

Please input the pay per hour
9.50
Please input the number of hours worked
40
The gross pay is $380
The net pay is $323
We hoped you enjoyed this program

```

Print out the program and the running result for Exercise 2, and hand them in with the rest of the lab.

Exercise 3: Are the parameters `gross` and `net`, in the modified `computePaycheck` function you created in Exercise 1 above, pass by value or pass by reference?

Exercise 4: What will happen if we remove the two “&” symbols from the `computePaycheck` function prototype and the function header?

Answer questions in *Exercise 3 and 4* on a separate piece of paper and hand it in with the rest of the lab.