

Polytechnic University of Puerto Rico
Electrical and Computer Engineering & Computer Science Department
CECS 2203 – Computer Programming I Lab

Lab 3

Name: CORAL S. SCHMIDT MONTILLA ID#: 148830

1. Copy the source code developed for Lab 3 and paste it as **text** below. (15 points)

```
/*
* CECS 2203, Computer Programming I Laboratory
* Spring 2023, Sec. 06
* Date: March 30, 2023
* Topic: Lab 3 – Void and value returning methods
* File name: lab03.cpp
*
* Instructions and problem statement:
* The Fibonacci Sequence is the series of Fibonacci numbers. Each Fibonacci number
* is computed by adding the values of the previous 2 Fibonacci numbers. For this
* exercise we will define the first 2 Fibonacci numbers as:
* the Fibonacci number for 0 is 0, and the Fibonacci number for 1 is 1.
* The Fibonacci numbers are only defined for positive integers, which are the values
* greater than or equal to 0. Your solution prompts the user for the number of terms
* in the sequence, computes each Fibonacci number, and prints the corresponding
sequence.
*
* Complete the program by writing the correct C++ statements. Execute the program
and
* save a screenshot for your report.
*
* Name: CORAL SCHMIDT MONTILLA, YOUR ID# 148830
*/
// write the appropriate preprocessor directive
#include <iostream>

// write the appropriate using directive
using namespace std;

// declare a method named fiboNumber that receives 1 integer as parameter
// and returns an integer value
int fiboNumber(int);

// declare a method named fiboSequence that receives 1 integer as parameter
// and has no return value
void fiboSequence(int);

// declare a method named personalInfo which has no parameters or return value
void personalInfo();

int main() {
    // Write the statement that declares the integer variable terms
    // and initializes it to 0
    int terms = 0;
```

Polytechnic University of Puerto Rico
Electrical and Computer Engineering & Computer Science Department
CECS 2203 – Computer Programming I Lab

```
    // Develop a do/while iteration control structure which uses the value of
terms    // as the sentinel. Prompt the user for a positive integer, using the phrase
    // "Enter the numbers of terms to be printed: " and store the value in terms.
    // Make sure that the value entered by the user is greater than 0 using a
do/while  // iteration control structure. If the value is not acceptable, print the
phrase    // "\n\tERROR! Value must be a greater than 0!\n\n". Once a correct value is
received, // call the fiboSequence method using the value of terms as parameter. After
the       // sequence has printed, ask the user if another sequence is wanted using the
phrase    // "Print another sequence? Enter 0 to exit or any other number to continue:
", and save
    // the value in terms. The do/while cycle will continue if the value of terms
is not 0.
    // Test your code by entering at least 1 incorrect value and 3 correct
values.

do {
    // start the nested do/while structure here

    do {

        // Prompt the user for the number of terms
        cout << "Enter the numbers of terms to be printed: ";

        // store the value in terms
        cin >> terms;

        // implement an if selection control structure to print the
error message    if (terms <= 0) {
                    // print the error message
                    cout << "\n\tERROR! Value must be a greater than 0!\n\n";
                }

        } // the condition tests if the value of terms is acceptable
while (terms < 0);

        cout << endl;
        // call the fiboSequence method using the value of terms as parameter
        fiboSequence(terms);
        cout << endl;

        // ask the user if another sequence is wanted
        cout << "Print another sequence? Enter 0 to exit or any other number to
continue: ";

        // store the value in terms
        cin >> terms;
        cout << endl;
```

Polytechnic University of Puerto Rico
Electrical and Computer Engineering & Computer Science Department
CECS 2203 – Computer Programming I Lab

```
} // The do/while cycle will continue if the value of terms is not 0.
while (terms != 0);

// call the personalInfo method
personalInfo();

cout << endl;
system("pause"); // for Visual Studio only
return 0;
}

// The fiboNumber method receives 1 integer as parameter and returns an integer
value.
// This function computes the Fibonacci number adding the previous 2 Fibonacci
numbers.
// Recall that the Fibonacci number for 0 is 0, and the Fibonacci number for 1 is 1.
// Implement and if/else if/else selection control structure to return the correct
value.
// Use the label number for the parameter. The first two conditions test if the
value of
// the parameter is 0 or 1 and the block of code associated with the else computes
and returns
// the Fibonacci number using a for iteration control structure. In the else block,
declare the
// integer variable fn2 initialized to 0, fn1 initialized to 1, and fn initialized
to 0. Develop
// a for cycle to compute the Fibonacci number. Recall that the formula used to
compute a
// Fibonacci number is  $fn = fn1 + fn2$ , and after each iteration fn2 is assigned the
value of fn1
// and fn1 is assigned the value of fn. When the for cycle stops, return the value
stored
// in the fn variable.

int fiboNumber(int number){
    if (number == 0) // test if the value of the parameter is 0
        // if true, return 0
        return 0;

    else if (number == 1) // test if the value of the parameter is 1
        // if true, return 1
        return 1;

    else { // compute the Fibonacci number
        // declare the integer variables f2 with initial value of 0,
        // f1 with initial value of 1, and fn with initial value of 0
        int f2 = 0;
        int f1 = 1;
        int fn = 0;

        // implement a for control structure using i as the counter with
        // initial value of 0, and the condition is that i must be less than
```

Polytechnic University of Puerto Rico
Electrical and Computer Engineering & Computer Science Department
CECS 2203 – Computer Programming I Lab

```
        // the value of the parameter.
        for (int i = 0; i < number; i++) {
            // save in fn the sum of f1 and f2
            fn = f1 + f2;

            // assign the value of f1 to f2
            f2 = f1;

            // assign the value of fn to f1
            f1 = fn;
        }
        // return the value stored in fn
        return fn;
    }
}

// The fiboSequence method receives 1 integer as parameter and has no return value.
// It prints the phrase "The [numbers] term Fibonacci sequence is: ", and then
// implements a for
// iteration control structure which calls the fiboNumber method to get the
// corresponding
// value to be printed. Develop the logic that prints ", " after each term of the
// sequence
// and "." after the last term. Label the parameter as numbers

void fiboSequence(int numbers){

    // print the initial phrase; do not include an endl after the phrase!
    cout << "The " << numbers << " term Fibonacci sequence is : ";

    // print the sequence, use i as the counter, initialize to 0
    // the condition is that the value of the counter is less than the value
    // of the parameter
    for (int i = 0; i < numbers; i++) {
        // print the return value of fiboNumber using with the value of the
        // counter as parameter
        cout << fiboNumber(i);

        // implement an if/else structure to decide which punctutation is to be
        // printed
        if (i == numbers - 1)
            cout << ".";

        else
            cout << ", ";
    }
}

// The personalInfo method has no parameters or return value. It prints the phrase:
// "Program developed by [YOUR NAME], ID#[YOUR ID NUMBER]", where the square
// brackets
// and the text within is substituted with your personal information. Make sure to
// add
```

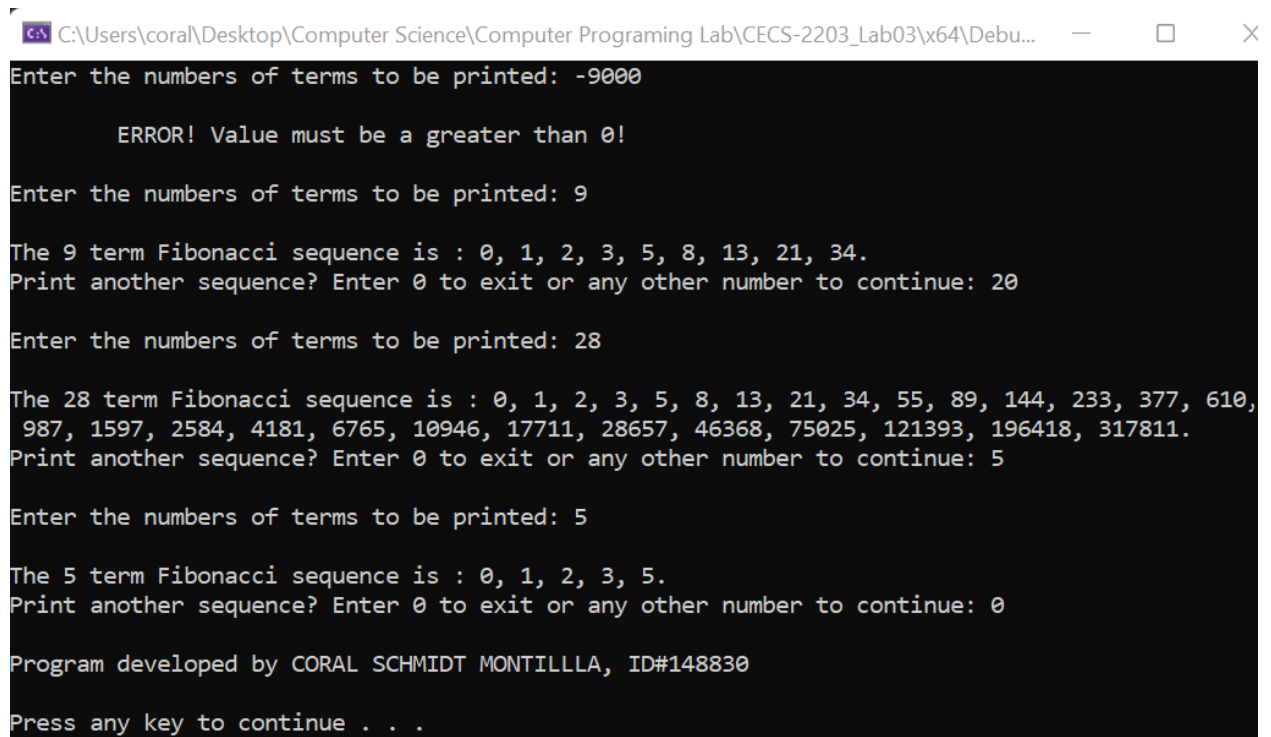
Polytechnic University of Puerto Rico
Electrical and Computer Engineering & Computer Science Department
CECS 2203 – Computer Programming I Lab

```
// an empty line after the phrase.
void personalInfo() {

    cout << "Program developed by CORAL SCHMIDT MONTILLLA, ID#148830";
    cout << endl;

}
```

2. Paste the screenshots of the program's execution below. (5 points)



```
C:\Users\coral\Desktop\Computer Science\Computer Programing Lab\CECS-2203_Lab03\x64\Debu...
Enter the numbers of terms to be printed: -9000

        ERROR! Value must be a greater than 0!

Enter the numbers of terms to be printed: 9

The 9 term Fibonacci sequence is : 0, 1, 2, 3, 5, 8, 13, 21, 34.
Print another sequence? Enter 0 to exit or any other number to continue: 20

Enter the numbers of terms to be printed: 28

The 28 term Fibonacci sequence is : 0, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610,
987, 1597, 2584, 4181, 6765, 10946, 17711, 28657, 46368, 75025, 121393, 196418, 317811.
Print another sequence? Enter 0 to exit or any other number to continue: 5

Enter the numbers of terms to be printed: 5

The 5 term Fibonacci sequence is : 0, 1, 2, 3, 5.
Print another sequence? Enter 0 to exit or any other number to continue: 0

Program developed by CORAL SCHMIDT MONTILLLA, ID#148830

Press any key to continue . . .
```

3. Comment on any warnings or errors revealed by Visual Studio. If any error messages were present, list the error and describe how you corrected it. If no errors or warnings were revealed, comment on the most important aspect of developing the solution. (5 points)

There were no warnings or errors revealed by Visual Studio during the making of this program. The program wasn't that hard to develop. The instructions were pretty clear, and easy to understand. It still was a good way to challenge my brain.