



# Computer Science 2510 - Lab 2

## Readings

- Class Notes
- Textbook: Chapters 5-6

## Objectives

- To become familiar with types of errors and exceptions, and simple functions.

## Notes

- Most of the exercises in this lab were taken from the "Exercises" and "Drill" sections of Chapters 5 and 6 of the textbook (Bjarne Stroustrup, *Programming - Principles and Practice Using C++*, Second edition, Addison-Wesley, 2014, ISBN 978-0-321-99278-9.)

## Lab Exercises

1. If you have not already done so, create a directory for the course, called `comp2510`, and subdirectories for each of the labs, called `lab01`, `lab02`, etc. You should save the programs that you create/modify for this lab in your `lab02` directory.
2. If you have not already done so, download the modified header file which accompanies the textbook, `std_lib_facilities.h` here: [std\\_lib\\_facilities.h](#).

Put this file in your `comp2510` directory. Any C++ files contained in one of your `lab` directories can then reference it by using: `#include "../std_lib_facilities.h"`

### 3. Chapter 5 Exercises

Remember, to run a C++ program called, for example, `CelsiusToKelvin.cpp`, you would use the command: `c++14 -o CelsiusToKelvin CelsiusToKelvin.cpp`

- 3.1. The following program takes in a temperature value in Celsius and converts it to Kelvin. This code has many errors in it. Find the errors, list them, and correct the code.

```
double ctok(double c) // converts Celsius to Kelvin
{
    int k = c + 273.15;
    return int
}
int main()
{
```

```

double c = 0; // declare input variable
cin >> d; // retrieve temperature to input variable
double k = ctok("c"); // convert temperature
Cout << k << '\n' ; // print out temperature
}

```

- 3.2. Absolute zero is the lowest temperature that can be reached; it is  $-273.15^{\circ}\text{C}$ , or 0K. The above program, even when corrected, will produce erroneous results when given a temperature below this. Place a check in the main program that will produce an error if a temperature is given below  $-273.15^{\circ}\text{C}$ .
- 3.3. Do the previous exercise again, but this time handle the error inside `ctok()`.
- 3.4. Add to the program so that it can also convert from Kelvin to Celsius.
- 3.5. Write a program that converts from Celsius to Fahrenheit and from Fahrenheit to Celsius (formula in Section 4.3.3,  $9.0/5*\text{celsius}+32$ ). Use estimation (see Section 5.8 of text for more details) to see if your results are plausible.
- 3.6. Write a program that reads and stores a series of integers and then computes the sum of the first  $N$  integers. First ask for  $N$ , then read the values into a `vector`, then calculate the sum of the first  $N$  values. For example:
 

```

Please enter the number of values you want to sum:
3
Please enter some integers (press '|' to stop):
12 23 13 24 15 |
The sum of the first 3 numbers ( 12 23 13 ) is 48.

```

Handle all inputs. For example, make sure to give an error message if the user asks for a sum of more numbers than there are in the vector.
- 3.7. Modify the program from exercise 3.6 above to use `double` instead of `int`. Also, make a `vector` of `doubles` containing the  $N-1$  differences between adjacent values and write out that `vector` of differences.
- 3.8. Write a program that writes out the first so many values of the **Fibonacci** series, that is, the series that starts with 1 1 2 3 5 8 13 21 34. The next number of the series is the sum of the two previous ones.

**Author: Department of Computer Science, MUN (BE220531)**