Computer Science 2510 - Lab 7

Readings

- Class Notes
- Textbook: Chapter 18

Objectives

- To become familiar with constructors, destructors, arrays and vectors.

Notes

- Most of the exercises in this lab were taken from the "Try This" and "Drill" section of Chapter 18 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. **Chapter 18, Section 18.4.2 - Try This**

   Constructors and Destructors Example Code (click/right-click to download)

   Run this example and make sure you understand the result. If you do, you'll understand most of what there is to know about construction and destruction of objects.

2. **Chapter 18 Drills, Array drill**

   2.1. Define a global `int` array `ga` of ten `int`s initialized to 1, 2, 4, 8, 16, etc.

   2.2. Define a function `f()` taking an `int` array argument and an `int` argument indicating the number of elements in the array.

   2.3. In `f()`:

   - Define a local `int` array `la` of ten `int`s.
   - Copy the values from `ga` into `la`.
   - Print out the elements of `la`.
   - Define a pointer `p` to `int` and initialize it with an array allocated on the free store with the same number of elements as the argument array.
   - Copy the values from the argument array into the free-store array.
   - Print out the elements of the free-store array.
   - Deallocate the free-store array.

2.4. In `main()`:

- Call `f()` with `ga` as its argument.
- Define an array `aa` with ten elements, and initialize it with the first ten factorial values (1, 2*1, 3*2*1, 4*3*2*1, etc.).
- Call `f()` with `aa` as its argument.

3. Chapter 18 Drills, Standard library `vector` drill

3.1. Define a global `vector<int> gv`; initialize it with ten `int`s, 1, 2, 4, 8, 16, etc.

3.2. Define a function `f()` taking a `vector<int>` argument.

3.3. In `f()`:

- Define a local `vector<int> lv` with the same number of elements as the argument `vector`.
- Copy the values from `gv` into `lv`.
- Print out the elements of `lv`.
- Define a local `vector<int> lv2`; initialize it to be a copy of the argument `vector`.
- Print out the elements of `lv2`.

3.4. In `main()`:

- Call `f()` with `gv` as its argument.
- Define a `vector<int> vv`, and initialize it with the first ten factorial values (1, 2*1, 3*2*1, 4*3*2*1, etc.).
- Call `f()` with `vv` as its argument.

Author: Department of Computer Science, MUN (BE220531)