Module 5 - Exercises

1. Dynamic memory allocation for objects

Write a C++ program which defines a class named **Book** with two private attributes ***name (string)*** and ***price* (integer)**. Provide constructor to initialize those attributes.

a) Create three **Book** objects using dynamic memory allocation and initialize values for them through the constructor. *You should check for NULL pointer to detect whether the allocations are succeeded or not.*

Use range-based for loop to print out all information of the objects, and free up memory space after that.

b) Ask the user to enter a number ***n***. Create an array of ***n* Book** objects using dynamic memory allocation and ask the user to input value for them.

Print out the most expensive book. Free up memory space after that.

2. Dynamic memory allocation for class attribute

Duplicate and modify the program of question 1 so that **name** attribute is a pointer, which is *dynamically allocated* in heap memory via the class **constructor**. Write a destructor to free up memory allocated for **it** (when a Book object is destroyed).

3. Try-Catch statement. Write a program to

a. Ask the user to enter two double values **a** and **b** for division (a/b). Throw an exception if b is 0, and ask the user to re-enter value for b.

b. Write a loop that continuously request 1MB from the heap (as an array of char). Catch the exception to stop the loop if allocation fails and print out how much memory has been allocated.

Check with physical size of RAM and Page file memories of your computer (using dxdiag command). Does it makes sense ?