Langara College
Department of Computing Science & Information Systems
CPSC1160 – Algorithms and Data Structures I
Lab_10: Inheritance, Polymorphism and exception handling
Dr. Bita Shadgar

1. Instructions

a. Writing your program with nice style is part of your evaluation. Indentation, documentation, modularization and selecting good names for your variables and constants are important.
b. In order to **not to lose mark about indentation**, make sure that find and replace all the tabs in your final program with 4 spaces, then save the code and submit it. This is due to different size of tab in different word processors (editors). It happens sometimes that your program lose right indentation when it is open in a new editor.
c. Read whole assignment first and make sure that you understand different parts of assignment and due dates. If you have any doubt or you are not clear about the assignment, you should ask in lab sessions or office hours. There is no grantee to get answer for your questions in this regard via email out of those times.
d. Create a folder named **Lab10**.
e. Inside folder **Lab10,** create a file for each problem.

Problem 1: [10 marks] Online quiz (Filename: **Quiz.pdf**)

Answer Multiple Choice Quiz at www.cs.armstrong.edu/liang/cpp3e/quiz.html for chapters 15 and 16. Then save your results as an image and paste them into a text file. Finally convert text file to pdf version named Quiz.pdf.

Problem 2: [15 marks] Base class and derived class (Use a header file including header and implementation for each class)

Design a class named **Person** and its two derived classes named **Student** and **Employee**. Make **Faculty** and **Staff** derived class of **Employee**. A person has a name, address, phone number, and e-mail address. A student has class status (freshman, sophomore, junior, or senior). An employee has an office, salary, and date-hired. Define a class named **MyDate** that contains the fields year, month, and day. A faculty member has office hour and a rank. A staff member has a title.

Define a constant virtual **toString** function in the **Person** class and override it in each class to display the class name and the person's name.

Problem 3: [5 marks] Test program for classes (Test.cpp)

Write a test program that creates a Person, Student, Employee, Faculty and Staff and invoke their toString() function.

Problem 4: [10 marks] Exception Handling

The organization has an unwritten rule that for every 10-12 students a person must be employed.
Rewrite your classes and its method to throw a **runtime_error** if the number of employees doesn't satisfy the rule. The exception should be clear whether it is time to **hire** or **fire** employees.

Problem 5: [10 marks] Test program for exception handling (Test.cpp)

Write a test program that creates 1000 random person and checks the unwritten rule. The ratio of generating student, faculty and staff are 0.91, 0.02 and 0.07 respectively.

Due date

- By the end of the lab time, demonstrate **Problem 2**.
- By 11:59pm on Monday 10, April 2017, submit a zip file named Lab9.zip which includes 7 files named **Person.h, Student.h, Employee.h, Faculty.h, Staff.h, Test.cpp** and **Quiz.pdf** to D2L.