EECE476 OOP and Data Structures for Computer Engineering

Project 02

Objective:

➢ Understand how to define classes and create objects.
➢ Understand inheritance and run-time polymorphism through virtual functions.
➢ Understand overloaded operators.

Description:

For this project, you will implement data structure using objected-oriented programming. You will create four classes: person, student, teacher and course.

1. Person class has two attributes: they are "name" and "age". Person class has two methods that return the name and age attributes. Person class has one constructor and one destructor. Person class also has one virtual function called "printname" that print out the names of the person in the message: "This is ###". ### is the name. (implement person.h and person.cpp file)
2. Student class is inherited from person class.
   1) Student class have two member variables (attributes): student id number "student_num" and a list of course (arrays as class member) that he/she is taking.
   2) Student class has four member methods: 1) get id student number; 2) print out the list of courses that he/she is taking; 3) add courses to the list; 4) delete courses in the list.
   3) Student class also has a friend function, which overloads operator ==, to compare two students' list of courses are equal or not.
   4) Student class overrides the virtual function "printname" from Person class. It prints out the message "This is student ###". ### is the name.
   5) Student class has one constructor and one destructor. (implement student.h and student.cpp file).
3. Teacher class is inherited from person class.
   1) Teacher class has two more attributes: salary and a list of courses that he/she is teaching. Teacher class has two additional methods. One is to get the teacher's salary, the other is to print out the course that he/she is teaching.
   2) Teacher class overrides the virtual function "printname" from Person class. It prints out the message "This is teacher ### ". ### is the name.
   3) Teacher class has one constructor, one destructor. (implement teacher.h and teacher.cpp file)
4. Course class has three attributes. They are course number, course name and course description. Course class has three methods to get its three attributes. Course class has one constructor and one destructor (implement course.h and course.cpp file)

You also need to write a C++ test driver program (called test.cpp) to enter the data and finish the following tasks:

1) Initialize five courses objects and three person objects, using constructors.
2) Initialize three student objects and three teacher objects using constructors.
3) Enter the courses that students are taking, and teachers are teaching now.
4) Student add courses and delete courses.
5) Compare two students' list of courses are equal to test overloaded operator "==".
6) Demonstrate run-time polymorphism through "printname" virtual functions.
7) Print out all information about the data structure that you created. They are students' information, teacher's information, the courses they are taking and teaching.

Requirement:

For a list of courses (the class member of student class and teacher class), you can use array or existing data structure (list, map, vector, etc) in C++ Standard Template Libraries (STL) to implement it. STL are a set of C++ template classes to provide common programming data structures and functions such as doubly linked lists (list), paired arrays (map), expandable arrays (vector), large string storage and manipulation (rope).

Submission:

1) Working code - source files (*.cpp, *.h files) (70pts)
1) A lab report is required. The content should include: a. Introduction; b. Screenshot of outputs and explanations; c. Conclusion. The report should follow the report guideline. (20pts)
2) Comments in your code (10pts).