CE/CS/SE 3354.004, Software Engineering, Fall 2017
Homework Assignment #1

Due: 11:59pmm, September 29
Submission: *.doc (or *.pdf) via elearning
Maximum points: 100
The submitted file should contain only solutions, and should be named by the assignment, your id, and your first-last name, e.g., hw01-xxx123456-jon-bell.doc

I Software Process Models

How does the waterfall model differ from the iterative model? What are the main advantages of the latter over the former? (10 Points)

II Requirements Specification

Suppose that you are going to develop a network-based course registration system for the university. For the ease of understanding a partial natural-language based requirements has been shown as follows:

1. Students
    1.1. The students should be able to log into the system
    1.2. The students should be able to add their course registrations
    1.3. The students should be able to remove their course registrations
    1.4. The students should be able to change their course registrations
    1.5. The students should be able to query all courses
    1.6. The students should be able to list all their registered courses
2. Instructors
    2.1. The instructors should be able to log into the system
    2.2. The instructors should be able to query student list for all classes
    2.3. The instructors should be able to edit course descriptions
    2.4. The instructors should be able to print the course information
3. System administrators
    3.1. The system administrators should be able to log into the system
    3.2. The system administrators should be able to manage the student information
    3.3. The system administrators should be able to manage the course information, e.g., add, change, and remove courses
    3.4. The system administrators should be able to add, switch, and remove students for each course

Please use form-based specification to describe how the students can **add**, **remove**, and **change** their course registrations. (20 Points)
Instruction: each form-based specification should include the following items: Function, Description, Inputs, Source, Outputs, Destination, Action, Requirement, Pre-condition, Post-condition, and Side effects. Use the insulin pump example in the lecture slides as a template.

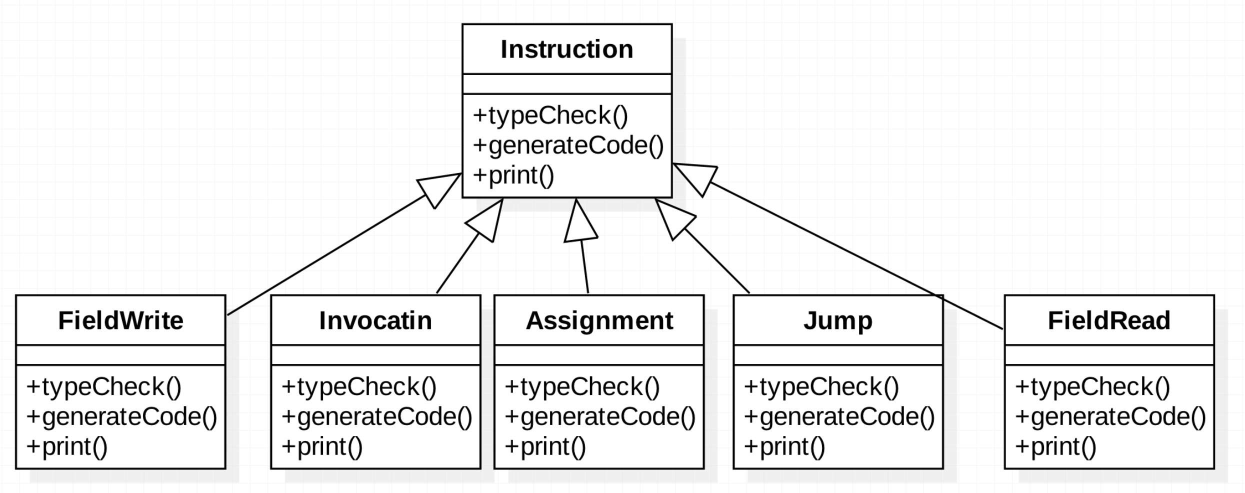
III UML Class and Sequence Diagrams

1) Please draw the class diagram to design the course registration system. (25 Points)
Instruction: utilize various class relations: generalization, aggregation, composition, and association; the class diagram should **include** but not be limited to the following class names: User, Student, Instructor, Admin, Course, LogonPage, AddPage, DropPage, ChangePage, PrintPage, QueryPage, CourseInfoDB, StudentInfoDB, InstructorInfoDB, AdminInfoDB, and EntireDB.

2) Please draw a sequence diagram for each of the following two scenarios:
a) An instructor logs into the system, and then prints course information and finally logs out. (15 points)
b) A student logs into the system, and then can query, add, and remove courses until he/she finalizes the course selection. Then he/she logs out. (15 Points)

IV Design Patterns

Given the following original class diagram for a compiler project under construction, please use the visitor design pattern to redesign the project, and draw the new class diagram. (15 Points)
Instruction: use visitor pattern for all operations from each class.


Instruction
+typeCheck()
+generateCode()
+print()
FieldWrite
+typeCheck()
+generateCode()
+print()
Invocatin
+typeCheck()
+generateCode()
+print()
Assignment
+typeCheck()
+generateCode()
+print()
Jump
+typeCheck()
+generateCode()
+print()
FieldRead
+typeCheck()
+generateCode()
+print()