COMPUTER ENGINEERING DEPARTMENT

CMPE 202

Software Systems Engineering

Spring 2018
Midterm Exam Instructions

PART 1

Multiple Choice Questions

50 Points Total
30 Minutes Duration

When Announced by Instructor or Proctor,
you may begin testing on Canvas Quiz
using LockDown Browser.

Exam Instructions (Read Carefully)

- You have 30 minutes on this part of the exam
- This is a closed book, closed notes exam.
- Close all documents, programs and windows on your computer
- The only allowed window should be the Canvas window using Lockdown Browser
- Questions will be presented one-at-a-time and your answers will be locked once submitted
- You may not return to your answers to change them, so make sure allow for enough time on each question
- You may use blank space on this exam as scratch area
- No talking or leaving exam room during Exam
- If you suspect any errors on the exam, please write a note of this on the exam paper and raise it to instructor's attention at the end of the exam
- Any violations of academic integrity will result in an immediate ZERO score on this exam

PART 2

Part 2 Exam Instructions :

UML Diagrams
75 Points Total
90 Minutes Duration

- *Exam is Closed Book, Closed Notes*
- *Please turn off all electronics*
- *Please make sure to*
  - *write in your name*
  - *sign the exam (on next page)*
- *Any talking/violations may result in deductions or excuse from exam at discretion of proctor.*
- *Exams not turned-in by exam end time will result in -5 points per minute late.*

In this part of the Exam, you will be provided with sample Java Source Code. Please review this Source Code and Draw UML Diagrams on the Answer Sheets with the following requirements.

UML State Diagram Requirements:

- Draw a State Machine Diagram for the State Machine implemented in the **Class PEM**.

UML Class Diagram Requirements:

Draw a Class Diagram for the Source Code/Files provided in the following pages using the following rules.

- **Classifiers**
  - The following Class or Interface must be shown as "Classifier Boxes" on the diagram
  - These include: **AC, MC, ICMD, IMENU, P, ISCR, SA, SB, SC, IPSM, ITEH, KP, CKA, ITA, KPH, PC,** and **PCH**
  - Make sure to clearly mark **Interfaces** with the "<<interface>>" stereotype

- **Attributes Compartments**
  - Attributes compartments can be excluded from Class Diagram

- **Operations Compartments**
  - include **Public** methods
  - Include name of **method arguments and type**
  - Include method return type except "**void**"
  - Include class/**Static** methods using underline
  - Do not include constructors
  - Do not include **Exceptions** thrown
  - Do not include **Private** methods

- **Relationships**
  - Show only **associations**, **implements** and **inheritance** relationships
  - Relationships should include **navigability** (or not navigatible) where appropriate
  - Show navigability with associations (including directionality and "no navigability" cross)
  - Instance variable names at association ends must include **visibility**
  - For **Static** instance variables, show using underline
  - Do not show any dependencies or "uses" relationships

UML Sequence Diagram Requirements:

Draw a Sequence Diagram for the executionof the method using the following rules.

- **Initial Method Call for Sequence Diagram**
  - **Start with call to app.touch(1, 5) in AT.test1() method. Note: assume all objects contructors and object references have been set up in call to "new A()". I.E. Do no show the messages before "app.touch(1,5)" call.**

```
public static void test1() {
     A app = new A();
     app.touch(1, 5);
 }
```

- **Participants**
  - Show Instance **Variable Names** and **Object Type**
  - If Instance Variable is Anonymous, just show Object Type

- **Messages**
  - **Method Calls** – Show **Method** names with **Argument values**
  - **Constructors** – You may **exclude all Constructor Calls** in Sequence Diagrams
  - **Return Messages** – No need to show these
  - For all **Console Output**, show the output as **UML comments linked to the messages**

- **Timelines**
  - **Activation Bars** – Must show activation bars for each method call timeline / scope.
  - **Self-Calls** – Show these as **Nested Calls** where appropriate.
  - **Call Stacks** – Must show these as **Nested Activation Bars**.