

**Instructions:**

The exam is online and open (book/notes-printed/notes-written/electronics). State your assumptions (if any) clearly in a file named *assumptions.txt*.

There are four questions in this exam. You need to answer each of the following questions. Record a maximum of 2-minute video response to questions 1 to 3. Write code for the 4<sup>th</sup> question. Upload your responses on your GitHub repository (; for submitting the exam, follow the same procedure that you have been following for PA submission). The link that lets you create a repository is emailed to you. The repository contains all the material that you need to start answering the exam.

**(Summary:** For the first 3 questions, your response should have 3 video files named A1, A2, A3 resp. (with appropriate file extensions) each of a maximum duration of 2 mins. For the 4<sup>th</sup> question, name the file as A4.cpp or A4.java. All your responses will remain confidential.)

Each instance of incorrect naming of a response file carries a **penalty of 0.5 points**. Incorrect submission (including Git commit and tagging), carries a **penalty of 2 points**.

**1. SDLC Process-** while working on the software development for PA1 in a team:

- a. Identify the software process or methodology that you followed. What are the distinguishing characteristic features of this process? **0.5 points**
- b. Identify at least one activity that you thought went well when you followed this process. **0.5 points**
- c. Identify at least one activity that you thought did not go well. **0.5 points**  
Your comments should focus on the process, planning and coordination, execution only.

**2. Software Architecture**

- a. Identify at least two architectural styles that you have used in courses studied (/studying) in your curriculum (excluding CS305) or elsewhere **0.5 points**
- b. Name the courses where you have used them. **0.5 points**
- c. How did you use them? **0.5 points**

**3. Detailed Design**

- a. Identify at least one design pattern that you have used in courses studied (/studying) in your curriculum (excluding CS305) or elsewhere **0.5 points**
- b. Name the course(s) where you have used it. **0.5 points**
- c. How did you use it? **0.5 points**

**4. Design Principle**

- a. You are given two implementations of the FactoryMethod design pattern using C++ and Java. *The implementation in `vehicLefactory.cpp` is less cohesive compared to that in `FactoryDemo.java`. Re-implement `vehicLefactory.cpp` to improve the cohesiveness.* You may use either C++ or Java to reimplement. If you are writing a Java program, imagine that you were using a Java version that did not support constructors, methods, and attributes in enums. Write code in a file called `A4.cpp` or `A4.java`. Your code should not have any compiler/linker errors. **3.5 points**