

CS 425 Final Exam Review

Scheduled on: Tuesday December 17, 2024 from 10:15 am

Weight: 21% of course grade

Exam type: open-book ("take home"), timed, individual, online (via Canvas),
synchronous (everyone starts at the same time)

CS 425 Final Exam, Tuesday, December 17, 2024

- You have **100 minutes** to take the exam. This exam is **open book** and should be taken **online, electronically** through web campus (Canvas). Please note the following:
 - The exam **will start for all exactly at 10:15 am on Tuesday December 17, 2024**
 - You may start few minutes late (like max 10 minutes without losing exam time) **but the exam will be closed automatically on Canvas at 12:15 pm or after 100 minutes from your start, whichever comes first.**
 - During the test there will be a **Zoom link available to you to report only special, unexpected situations, but not to ask questions.** This link will be provided a day or two before the final exam. However, you do not have to log on Zoom for the exam. In fact, **try to not use Zoom during the test** – instead, **login in on Canvas to take the midterm exam there.**
 - **DRC-approved students** may have a longer time to complete the exam, but they will also **start it at 10:15 am on December 17, 2024.**

CS 425 Final Exam, Tuesday, December 17th, 2024

[continued]

- We will make this final exam as clear and straightforward as possible. [We would not answer any questions about it during the exam](#) as this would be unfair or disruptive to other students. If strictly needed, make a [reasonable assumption](#) about the question you have doubts about and state it in your answer/comments to submission (in one of the essay questions).
- Please note that even though this exam is open book, if you do not study ahead it might be difficult to answer all the questions within the 100-minute time frame. So, [please study for the test before the exam](#). During the test make sure [you pace yourself well](#), to allow you to answer all the questions.

Other Information about the Final Exam

- You will only be able to **answer 1 question at a time** and **you will not be able to go back to previous questions**
- Questions will be shuffled so the exam questions will have similar but somewhat different contents
- You need **50% or higher in exams** (midterm plus final) to pass the class
- Midterm was 13%, the final exam is 21% of the course grade
- While the final exam is open book, you cannot consult with anyone else while taking the exam. The exam is **strictly individual** and it **would be cheating if you communicate with anyone during the exam.**
- When the teaching team will grade the test, the essay answers will be cross-checked across the class. **Please do not cheat in any way.** Take pride on getting a grade based solely on your own work and effort!

Final Exam Questions

- Two types
 - Multiple Choice Questions (MCQs)
 - 24 to 36 questions
 - For each MCQ there will be 4 possible answers, with only one correct. You will be able/allowed to choose a single answer
 - Essay or Short Answer Questions (EQs)
 - 3 or 4 Essay Questions
 - You will need to elaborate your written answers on each of them

Study Required for the Final Exam (parts I and II)

I. FROM the TEXTBOOK

- **Chapters required** from the textbook Ian Sommerville, Software Engineering, 10th Edition, 2015.
 - Chapter 3 AGILE SOFTWARE DEVELOPMENT
 - Chapter 4 REQUIREMENTS ENGINEERING
 - Chapter 5 SYSTEM MODELING
 - Chapter 6 ARCHITECTURAL DESIGN
 - Chapter 8 SOFTWARE TESTING
 - Chapter 9 SOFTWARE EVOLUTION
 - Chapter 10 DEPENDABLE SYSTEMS
 - Chapter 22 PROJECT MANAGEMENT
 - Chapter 23 PROJECT PLANNING
- **Sections and subsections not required:**
 - Subsection 3.4.3 Agile Methods for Large Systems
 - Subsection 3.4.4 Agile Methods across Organizations
 - Section 5.5 Model-Driven Architecture
 - Section 6.4 Application Architectures
 - Subsection 8.3.2 Scenario Testing
 - Subsection 9.3.1 Maintenance Prediction
 - Section 10.3 Redundancy and Diversity
 - Section 10.4 Dependable Processes
 - Section 23.5 Estimation Techniques
 - Section 23.6 COCOMO Cost Modeling

Study Required for the Final Exam (continued)

II. From the PROJECT

In addition, you should know **your project** (Parts 1, 2, 3, and 4) as related essay questions (EQs) will most likely be included in the final exam.

Sample MCQs

1. What kind of software is tested during *release testing*?

- (a) Program components that need to be integrated by the development team
- (b) Program units such as methods or classes
- (c) A software system intended for use by customers and users
- (d) None of the above

2. Which of the following is a type of *software maintenance*?

- (a) Functionality addition to the software
- (b) Environmental adaptation of the software
- (c) Fault repairs to fix bugs and vulnerabilities
- (d) All of the above

Sample MCQs

3. Which of the following is not an activity of *reengineering*?

- (a) Source code translation
- (b) Dependency verification
- (c) Reverse engineering
- (d) Program structure improvement

4. Three key principles of *agile methods* are:

- (a) Customer involvement, embrace change, people not process
- (b) Risk analysis, maintain simplicity, incremental delivery
- (c) Incremental delivery, people not process, formal specifications
- (d) None of the above (that is, none of the above answers contains three valid key principles of agile methods)

Sample EQs

1. Briefly describe your team's *project topic* in CS 425 (what is your project about), describe its main goals, and indicate what would be several of its new or innovative characteristics. Then, consider the XP practices of *collective ownership* and *refactoring*. Briefly describe these two practices and discuss how they could be applied to your project's development.
2. Regarding *refactoring*, explain what *bad smells* are and give 3 concrete examples of bad smells and how to remedy them.
3. Consider the *layered architecture* pattern/style. Describe it, indicate in what situations it should be used, and discuss its advantages and disadvantages. You do not need to draw a related graphical diagram.
4. Assume you are a software project manager. Indicate *four way (means)* you would use *to motivate* the people working under your supervision.