CMSE-201 problem session 29.05.2025

Subjects covered: 1) [Ch. 6 COCOMO](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE201/Spring%202022/Ch6-Estimation%20techniques-COCOMO.pptx), 2) [Ch. 7 Architectural design](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE201/Spring%202023/Ch7%20Architectural%20design.pptx); 3) [Ch. 8 System modeling](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE201/Spring%202022/Ch8%20System%20modeling.pptx); 4) [Ch. 9 Testing](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE201/Spring%202022/Ch9%20Testing.pptx) (slides 1-44); 5) [Ch. 10 Quality management](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE201/Spring%202022/Ch10%20Quality%20management.pptx) (slides 1-39); 6\_ [Ch. 11 Configuration management](https://staff.emu.edu.tr/alexanderchefranov/Documents/CMSE201/Spring%202022/Ch11%20Configuration%20management.pptx) (Slides 1-44)

Ch 6 COCOMO

1. What are the three modes of the Basic COCOMO model? What is used as an input to the model? What are its outputs?
2. What is the difference between Basic and Intermediate COCOMO models?
3. How many cost drivers are used in Intermediate COCOMO model? What for they are used? What four groups comprise the drivers?
4. Can COCOMO model be used even if size in KLOC is not available at beginning of the project development?
5. What function point (FP) count is? What are the five inputs for FP calculation? How it can be adjusted?
6. How many technical complexity factors are used for FP adjusting? What scale is used for them?
7. How FP can be converted to KLOC?
8. What are the five factors affecting software productivity?

Ch 7 Architectural design

1. Why reuse of archıtecture ıs possıble?
2. Give an example of a relation between architecture and system characteristics
3. What are the five constituents of the 4+1 view architecture model?
4. What is characteristic for layered architecture pattern? What are its benefits? Disadvantages?
5. What is client-server architecture pattern? How one server can serve many clients concurrently? What are its benefits? Deficiencies?
6. What is Model-View-Controller architecture pattern? What are its advantages? Deficiencies?
7. What is pipe-and-filter architecture pattern? What are its advantages? Deficiencies?
8. What is repository architecture pattern? What are its advantages? Deficiencies?
9. What are the four types of application architectures?

Ch. 8 System modeling

1. What are the two reasons to use of graphical system models?
2. What is the aim of the context model?
3. What is the difference between the context models and use-case diagram?
4. What is the aim of the process model? What are its elements? How synchronization is specified? How decisions are specified?
5. What is the aim of the sequence diagram? What are its elements? What is the message? How the message is denoted? What are the message parameters? How the message parameters are specified? How the activity period of an actor is specified? How the time-line of an actor is specified? How ALT and LOOP elements are specified?
6. What is the communication diagram? What is the difference between the communication and sequence diagrams?
7. What is the class diagram? What the class is? What is the relation between a class and an object? What is the state variable? What is the method? What is the interface of the method? What is the formal parameter? What is the actual parameter? What is the relation between the forma and actual parameters? What is the association diagram? Inheritance diagram? Composition diagram? Aggregation diagram? Dependency diagram? Navigate diagram? How constraints on the associations are specified? What is the meaning of 1..\*, 1, 0..1?
8. What four metrics can be derived from an association diagram?
9. What is the package diagram? How the package and association diagrams are related?
10. What is the data-flow diagram? What are its elements? How hierarchical data-flow diagrams can be constructed?
11. What is the state-machine diagram? What are its elements? How hierarchical state-machine diagrams can be constructed? What actions can be associated with a state? When the transition happens?

Ch. 9 Testing

1. Validation and defect testing
2. Verification and validation
3. Software inspection and testing
4. Guideline-based testing
5. Equivalence partitioning
6. Stubs and drivers, their use in the bottom-top and top-bottom development
7. Interface testing
8. Breadth-first and bottom-first strategies in testing

Ch. 10 Quality management

1. What are the aims of quality management on an organization level? Project level?
2. What is the aim of the quality management process?
3. How shall be composed quality team?
4. What a quality plan is? What is its structure?
5. What is “software fitness for purpose”?
6. How to resolve quality conflicts?
7. How product and process quality are related?
8. What is process-based quality?
9. Why standards are important? What are the five types of standards?
10. Software quality and ISO9001
11. Reviews and inspections
12. Quality reviews
13. Three phases of the review process
14. Program inspections
15. Inspection checklists on data faults
16. Inspection checklists on control faults
17. Inspection checklists on input/output faults
18. Inspection checklists on interface faults
19. Inspection checklists on storage management faults
20. Inspection checklists on exception management faults

Ch. 11 Configuration management

1. What the configuration is? What is configuration item? What is check-in and check-out?
2. What are the four configuration management activities?
3. What is the version? What is the baseline? What is the codeline? What is the mainline?
4. What is branching and merging? How versions are identified?
5. Version control systems features
6. Repository, check-in, check-out
7. Distributed version control
8. Storage management for versions
9. System building, build platforms, build system functionality
10. Agile building, continuous integration, daily building
11. Minimizing recompilation