

## Sample Exam Questions for SE

Score	Evaluator	Section I: <b>Completion by Matching.</b> Identify the letter of the choice that best complete each statement.

- |   |  |
|---|--|
| <ul style="list-style-type: none"> <li>A. Availability</li> <li>B. Activity</li> <li>C. Client-server</li> <li>D. Context</li> <li>E. Development</li> <li>F. Ethnography</li> <li>G. Equivalence partition</li> <li>H. Functional requirements</li> <li>I. Heterogeneity</li> <li>J. Interaction</li> <li>K. Open-source</li> <li>L. Pair programming</li> <li>M. Pipe and filter</li> </ul> | <ul style="list-style-type: none"> <li>N. Reuse</li> <li>O. Scenario testing</li> <li>P. Sociotechnical systems</li> <li>Q. Specification</li> <li>R. System requirements</li> <li>S. Technical systems</li> <li>T. Test-driven</li> <li>U. Use-case</li> <li>V. User interfaces</li> <li>W. User requirements</li> <li>X. Validation</li> <li>Y. Verification</li> <li>Z. Workflow</li> </ul> |
|---|--|

- \_O\_ 5. \_\_\_\_\_ is an approach to release testing where you write a story describing how a system may be used and design tests based on the sequence of events in the scenario.
- \_P\_ 6. \_\_\_\_\_ is/are self-aware and include defined operational processes and procedures.
- \_A\_ 7. \_\_\_\_\_ is the ability of a system to deliver services when requested.
- \_L\_ 19. Requirements expressed as scenarios, \_\_\_\_\_, and test-first development are three important characteristics of extreme programming.
- \_W\_ 20. \_\_\_\_\_ are statements in a language that is understandable to a user of what services the system should provide and the constraints under which it operates.

Score	Evaluator	Section II: <b>Multiple Choices.</b> Identify the choice that best completes the statement or answers the question.

- \_D\_ 21. \_\_\_\_ is the phenomenon where a module in a system implements or partially implements a number of different system requirements.
- |               |             |
|---------------|-------------|
| a. Concerning | c. Floating |
| b. Scattering | d. Tangling |
- \_D\_ 22. Which is **NOT** one of the 4 levels at which software reuse is possible?
- |                          |                            |
|--------------------------|----------------------------|
| a. The abstraction level | d. The specification level |
| b. The object level      | e. The component level     |

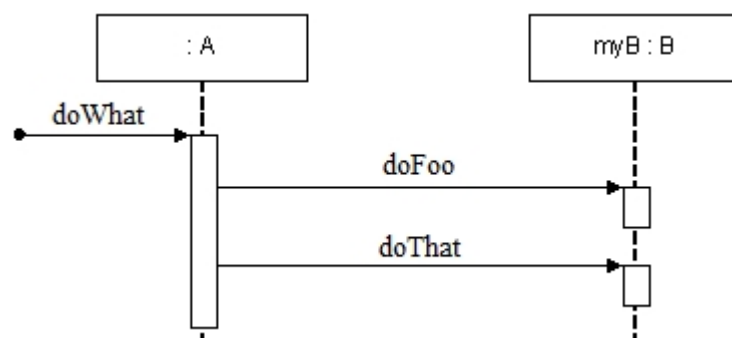
- c. The system level
- \_E\_ 23. Which is **NOT** one of the principal system re-engineering activities?
- a. Source code translation d. Reverse engineering
- b. Program structure improvement e. Program Debugging
- c. Program modularisation f. Data re-engineering
- \_A\_ 24. Which is **NOT** one of the four principal dependability properties?
- a. Understandability d. Reliability
- b. Availability e. Safety
- c. Security
- \_B\_ 25. Which is **NOT** one of the 4 sectors in each loop in Boehm's spiral model?
- a. Objective setting d. Risk assessment and reduction
- b. Feasibility study e. Development and validation
- c. Planning

Score	Evaluator	Section III: Short Answer. Please Select (mark a ✓ before the No. of the question chosen) and briefly answer 2 from the following 3 questions. (Question 31~33, 9 pts for each short answer, totally 18 pts)

- \_\_\_\_\_ 31. Explain, using examples as appropriate, what is meant by the following architectural style (choose only one from the following four styles):
- a. Pipes and filters architecture
- b. Layered architecture
- c. Client-server architecture
- d. Object-oriented architecture

Score	Evaluator	Section IV: Design Problems.

32. Consider the following Sequence Diagram in design stage, draw a equivalent communication diagram and write down the corresponding code segments in any object-oriented programming language for class A and class B.



Score	Evaluator	Section V: <b>Essay.</b> Answer the following question.

35. (? related to the course projects)