



## University of Colombo, Sri Lanka



**UCSC**

*University of Colombo School of Computing*



### **DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY ( EXTERNAL )**

Academic Year 2022 — 1<sup>st</sup> Year Examination — Semester 2

#### **IT2206 — Fundamentals of Software Engineering**

*Multiple Choice Question Paper*  
(2 Hours)

#### **Important Instructions**

- The duration of the paper is **2 Hours**.
- The medium of instructions and questions is English.
- This paper has **40 questions** on **12 pages**. Answer **all** questions.
- All questions are of the **MCQ** (Multiple Choice Questions) type.
- Each question will have **5 (five)** choices with **one or more** correct answers.
- This paper consists of 100 marks and all the questions will carry equal marks.
- There will be a penalty for incorrect responses to discourage guessing.
- The mark given for a question will vary from -1 (All the incorrect choices are marked & no correct choices are marked) to +1 (All the correct choices are marked & no incorrect choices are marked). However, **the minimum mark per question would be zero**.
- Answers should be marked on the **special answer sheet** provided.
- Note that questions appear on both sides of the paper. If a page is not printed, please inform the supervisor/invigilator immediately.
- Mark the correct choices on the question paper first and then transfer them to the given answer sheet which will be machine marked. **Please completely read and follow the instructions given on the other side of the answer sheet before you shade your correct choices.**
- Calculators are **not** allowed.
- *All Rights Reserved.* This question paper CANNOT be used without proper permission from the University of Colombo School of Computing.

**1).** Which of the following statement is/are best describes the term “Software Engineering”?

- a). The process of creating computer software programs.
- b). A systematic approach to software production that considers practical cost, schedule and dependability issues.
- c). The methodological process of creating software without a single error.
- d). The process of studying the application of computer algorithms to develop better software.
- e). An engineering discipline that is concerned with theories, methods, and tools for professional software development.

**2).** Which of the below statements is/are considered ethical principles that software engineers should adhere to in software development?

- a). Consider the broader impact of their work on society and strive to create software solutions that promote the public good.
- b). Shall not obtain proper consent or legal authorization from the users to collect or access personal data as they are involved in the design process.
- c). Accept all the work assigned to them without concentrating on their competency level to expedite software development.
- d). Respect the client’s confidentiality only if a formal confidentiality agreement has been signed.
- e). Be aware of the local laws in using patents and copyrights to ensure that the intellectual properties of both the employees and clients are protected.

**3).** Consider the following statements.

- A. In software projects, the environment in which software operates may change regularly compared to other projects making it harder to meet the user requirements.
- B. Software projects are less complex compared to other projects as these can be done without specialized knowledge or skills.
- C. Software projects can be tested exhaustively, and hence ensuring the quality, reliability, and functionality of software is easy compared to other projects.

Identify the correct statement(s) from the above that differentiate Software projects from other projects.

- a). A only.
- b). B only.
- c). A and C only.
- d). A and B only.
- e). All A, B and C.

**4).** The failure curve for software represents,

- a). The complexity of the software development process.
- b). The efficiency of software over time.
- c). The frequency of software failures throughout the software life-cycle.
- d). The time required to fix software defects.
- e). The cost of software maintenance over time due to software failures.

**5).** A software process model

- a). is an abstract representation of a process.
- b). guides the software development team to build the product.
- c). provides a more structured approach to organize the activities in the software process.
- d). illustrates how different people interact and involve with various activities in the model.
- e). always follows a plan-driven approach.

**6).** Consider the following statements.

- A. Each increment must deliver a fully functional software product.
- B. Each increment builds upon the previous increment, adding new features or functionality.
- C. Each increment requires producing extensive documentation illustrating the implementation.

Which of the above statements is /are correct regarding incremental development?

- a). A only.
- b). B only.
- c). A and C only.
- d). A and B only.
- e). B and C Only.

**7).** Which of the following is/are correct regarding software prototyping?

- a). A software prototype can be used in the systems design process to explore different software solutions and UI design.
- b). Sometimes software prototypes may result in creating sub-optimal solutions due to expediting building prototypes.
- c). Developers should always produce documentation when producing software prototypes.
- d). Software prototypes can only be used in plan-driven development.
- e). Software prototyping is an activity that must be carried out in a software process.

**8).** Consider the following scenarios.

- A. The main objective is to reduce the development time and effort by using well-established components.
- B. The technology stack and development tools are well-known and stable.
- C. The requirements are not fully defined or may evolve over time.

Identify the best software process model that can be used in the above scenarios.

- a). (A)- Waterfall; (B)-Incremental; (C)- Reused-based
- b). (A)-Agile; (B)- Waterfall ; (C)-Reused-based
- c). (A)-Waterfall; (B)-Waterfall ; (C)-Reused-based
- d). (A)- Reused-based; (B)- Waterfall; (C)-Incremental
- e). (A)-Incremental; (B)-Agile ; (C)-Plan driven

**9).** Consider the following statements.

- A. Process improvement is the process of improving the quality of user requirements and software design.
- B. The process maturity and agile approaches are two approaches to process improvement and change.
- C. Process analysis assesses the current process to identify weaknesses and bottlenecks.

Identify the correct statement(s) from the above regarding the process improvement in software engineering.

- a). A only.
- b). B only.
- c). A and C only.
- d). A and B only.
- e). B and C Only.

**10).** Which of the following is/are considered core principle(s) of agile software development?

- a). Emphasizing on the comprehensive documentation throughout the process.
- b). Following a strict plan without room for changes or adaptations.
- c). Encouraging customer collaboration and frequent feedback.
- d). Prioritizing process adherence over-delivering working software.
- e). Minimizing interactions between developers and stakeholders.

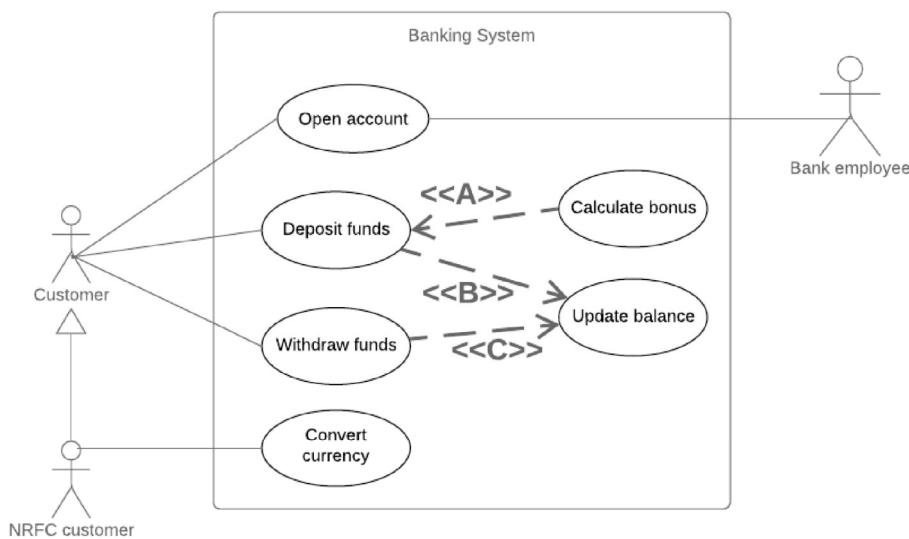
**11).** Which of the following is/are correct regarding SCRUM roles, events, and artifacts?

- a). Scrum retrospective is an artifact in SCRUM.
- b). The SCRUM team modifies the Sprint Backlog throughout the Sprint by adding or removing tasks or user stories appropriately.
- c). A sprint is a timeboxed development iteration that is 8 to 16 weeks long.
- d). At the initial phase of SCRUM, it completes the required documentation and assesses the lessons learned from the project to start the project.
- e). The product owner is the person who is responsible for requirements and maximizing the Return on Investment.

**12).** Extreme Programming,

- a). is a project management framework.
- b). discourages the use of automated testing and refactoring.
- c). practices heavy process documentation.
- d). emphasizes continuous integration and frequent releases.
- e). may release new versions several times per day.

**13).** Consider the following use-case diagram drawn in UML notations.



Which of the following is/are correct regarding the above diagram?

- a). The rectangle denotes the alternative associations and their interaction.
- b). “Customer” is the descendant (child) and the “NRFC Customer” is the ancestor (Parent).
- c). There are six (6) use-cases in this diagram.
- d). The “NRFC Customer” associates with “Open account,” “Deposit funds,” and “Withdraw funds” use-cases.
- e). A denotes an Include relationship while B and C denotes the Extend relationship.

**14).** Decide whether the below statements regarding user stories and scenarios are “True” or “False.”

- A. User stories are comprehensive, detailed, and formal descriptions of a feature or functionality from an end user’s perspective.
- B. Scenarios and user stories are formulated based on real-life situations to capture and communicate requirements and functionality.
- C. Scenarios and user stories are documented in the form of narrative text to capture and convey requirements and functionality.

Identify the correct combination of answers.

- a). (A)-False, (B)-True, (C)-False
- b). (A)-True, (B)-False, (C)-True
- c). (A)-False, (B)-False, (C)-False
- d). (A)-False, (B)-True, (C)-True
- e). (A)-True, (B)-True, (C)-True

**15).** Identify the non-functional requirement(s) from the below list of requirements.

- a). The system should support multiple user roles with varying access privileges.
- b). The system shall generate monthly invoices for subscribed customers.
- c). The system shall be scalable to handle a minimum of 1,000 concurrent users.
- d). The system shall comply with industry-specific security standards, such as ISO 27001.
- e). The system should be compatible with Windows, macOS, and Linux operating systems.

**16).** Which of the following persons is/are considered project stakeholders?

- |                          |                              |                              |
|--------------------------|------------------------------|------------------------------|
| a). The Project Manager  | b). The Project Sponsor      | c). The Business Competitors |
| d). The Business Analyst | e). Unrelated Business Units |                              |

**17).** The Use-case diagrams in Software Engineering are used to depict the,

- a). dependencies of the internal structure and organization of a software system.
- b). dynamic behavior of a software system.
- c). sequence of steps in a specific use case.
- d). dependencies among the functional and non-functional requirements.
- e). roles that interact with the system and the functionality provided by the system.

**18).** The Software Requirements Specification Document,

- a). describes the design and implementation details of a software system.
- b). includes both user and system requirements.
- c). is not useful for system test engineers as they cannot use the document to derive requirements for validation testing.
- d). outlines the user interaction design of a software system and is used by software developers to write code for a software system.
- e). should be detailed and precise when a separate company is developing the software through outsourcing.

**19).** Consider the following requirements validation checks and their proposed definitions.

- A. Consistency - Do any conflicts exist among the requirements?
- B. Realism – Have all the functions required by the customer been included?
- C. Verifiability - Is it feasible to implement the requirements considering the available budget?

Which of the above requirements validation check(s) are defined correctly?

- a). A only.
- b). B only.
- c). A and B only.
- d). B and C only.
- e). All A, B and C

**20).** Which of the following statements is/are correct regarding system modeling in software engineering?

- a). The main purpose of system modeling is to document user stories and scenarios.
- b). Elimination of the need for software documentation is a benefit in system modeling.
- c). The process in designing the system models enhances the collaboration among project stakeholders.
- d). Software verification techniques cannot be applied in the stage of System Modeling.
- e). System modeling is the process of creating diagrams to represent the structure and behavior of a software system.

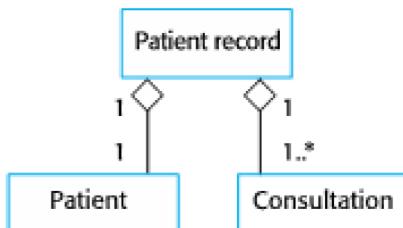
**21).** Which of the following UML diagram(s) is/are best suited to represent the dynamic interactions between actors and the system and between system components?

- a). Activity diagram
- b). Use case diagram
- c). Class diagram
- d). Sequence diagram
- e). State diagram

**22).** Which of the following statements is/are correct regarding different types of models used in system modeling?

- a). The structural models can be static or dynamic and illustrate the system design structure or the system when executed.
- b). Context models depict the operational context of a system, including the boundaries, the external entities, and the systems that it interacts.
- c). Event-driven modeling shows how a system responds to external and internal events.
- d). Use case and state transition diagrams are examples of interaction models.
- e). Behavioral models represent a system's static behavior and show what happens or what is supposed to happen when a system responds to a stimulus from its environment.

**23).** Consider the following class diagram.



The relationship between the “Patient record” and “Consultation” is/are (a)

- a). Generalization
- b). Inheritance
- c). Extent
- d). Include
- e). Aggregation

**24).** Consider the following statements.

- A. The programs that execute on a hardware/software platforms are generated automatically from the models.
  - B. An approach to software development that uses models as artifacts rather than programs.
  - C. A software development approach that does not involve any modeling activities.
- Which of the above statements is/are correct regarding Model Driven Engineering (MDE)?

- a). A only.
- b). B only.
- c). A and B only.
- d). B and C only.
- e). A and C Only.

**25).** Which of the following non-functional requirements should be considered in choosing an appropriate architectural style and the structure?

- |                  |                     |                            |
|------------------|---------------------|----------------------------|
| a). Security     | b). Scalability     | c). User Experience Design |
| d). Availability | e). Maintainability |                            |

**26).** The 4+1 View Model of Software Architecture,

- |  |
|--|
| a). includes four views (logical, process, development, and physical) and an additional view for testing.                  |
| b). provides multiple perspectives to analyze the software architecture.   |
| c). separates the architecture into four distinct layers: user interface, business logic, data access, and infrastructure. |
| d). emphasizes the separation of concerns and modularization of the software system.                                       |
| e). is a software development methodology based on incremental development.  |

**27).** The architectural patterns in Software Engineering,

- |   |
|---|
| a). solve specific design problems at the code level.                               |
| b). enforce coding standards to improve the code maintainability.                   |
| c). provide reusable solutions to common design problems at the structural level.   |
| d). defines the major components, relationships, and the interactions between them. |
| e). define an abstract description of good practice that has been tested earlier.   |

**28).** Which of the following is/are considered as an Architectural Pattern?

- |                   |                      |                             |
|-------------------|----------------------|-----------------------------|
| a). Observer      | b). Pipe and Filter  | c). Multi Purpose Publisher |
| d). Client Server | e). Abstract-Factory |                             |

**29).** Which of the following statements is/are correct regarding the Repository Architectural Pattern?

- |  |
|--|
| a). Focuses solely on data retrieval and does not support data modification or persistence.                          |
| b). Allows direct access to the database from the domain components rather than having a separate data access layer. |
| c). Changes done by one component can be easily propagated to all components.  |
| d). Defines a central repository that mediates between the components and the data storage.                          |
| e). Organizes the system into Layers with related functionality associated with each layer.                          |

**30).** Which of the following is/are used at the System level of software reuse?

- a). Language Libraries
- b). Architectural Patterns
- c). COTS
- d). Component Frameworks
- e). Design Patterns

**31).** Which of the following statements is/are correct regarding Open-Source Software / Development?

- a). Open-source development encourages a community-driven approach, where developers worldwide contribute to the software project.
- b). Open-source software is distributed under licenses that grant users the freedom to use, modify, and distribute the software.
- c). Open-source development communities maintain exclusive ownership of the software by a single entity.
- d). Open-source development allows only limited customization options for Business communities.
- e). Open-source software licensed under the GPL license can be sold since the code can be included in proprietary systems.

**32).** Identify the activity(ies) under the configuration management in Software Engineering.

- a). System Integration
- b). Version Management
- c). Host-target Development
- d). Problem Tracking
- e). Reverse Engineering

**33).** Consider the following statements.

- A. Software verification focuses on assessing the system's compliance with specified requirements, while software validation focuses on evaluating the system's fitness for its intended purpose.
- B. Software verification is concerned with testing the individual components of a software system, while software validation is concerned with testing the system as a whole.
- C. Software verification and validation are synonymous terms used interchangeably to refer to the same process of assessing the software's functionality and quality.

Which of the above statement(s) best describes the relationship between software verification and validation?

- a). A only.
- b). B only.
- c). C only.
- d). B and C only.
- e). All A, B and C.

**34).** Dynamic software testing techniques can be used to identify defects in

- a). program code.
- b). user stories.
- c). design diagrams.
- d). executable prototypes.
- e). paper prototypes.

**35).** Which of the following statements is/are correct regarding Test-Driven Development(TDD)?

- a). The main purpose of TDD is to ensure that only the critical functionalities are tested thoroughly.
- b). It eliminates the need for writing manual tests and speeds up the coding process.
- c). The correct sequence of steps in TDD includes writing code, writing tests, and refactoring code.
- d). A regression test suite is developed incrementally in TDD as the program is developed.
- e). The TDD can only be applied in Agile software development methodologies, and this could not be incorporated in plan-driven methodologies.

**36).** Identify the correct statement(s) regarding White and Black Box testing techniques used in Software testing.

- a). Black box testing requires knowledge of the system's internal workings, such as the code, algorithms, and data structures.
- b). The system as a whole, ensuring that it meets the intended functionality and requirements is essential in White box testing.
- c). Black box testing is a testing technique that focuses on the external behavior and functionality of the software system.
- d). In White box testing, test cases are derived based on the system's internal logic to ensure that all code paths are exercised and tested.
- e). White and Black box testing techniques are mutually exclusive and cannot be used together to test a software.

**37).** Stress testing

- a). ensures that the changes done to the code have not ‘broken’ the previously working code.
- b). is used to determine the stability of a given system.
- c). is a type of non-functional testing.
- d). ensures the software does not crash in conditions of insufficient computational resources.
- e). is a form of performance testing where the system is deliberately overloaded to test its failure.

**38).** Which of the following statements is/are correct regarding Legacy Systems?

- a). Lack of documentation and understanding of the system's functionality is a challenge in maintaining Legacy Systems.
- b). An organization no longer uses a Legacy System as such systems are only used for historical purposes.
- c). Legacy systems are always outdated and inefficient.
- d). Legacy systems are broader socio-technical systems that include hardware, software, libraries and other supporting software and business processes.
- e). Legacy systems cannot be transformed through re-engineering to improve its maintainability.

**39).** Identify the aspects on which the complexity of software depends on.

- a). Control Structures
- b). Data Structures
- c). The number of users
- d). SE Methodology
- e). Module Size

**40).** Consider the following reengineering process activities and the definitions.

- A. Reverse engineering: Analyzing the program to increase the understandability.
- B. Program modularization: Reorganizing the program structure.
- C. Data reengineering: Cleaning up and restructuring system data.

Which of the above is/are correct regarding the Reengineering process activities?

- a). A only.
- b). B only.
- c). C only.
- d). A and B only.
- e). All A, B and C.

\*\*\*\*\*