

COMP380 SPRING 2011 Midterm I

Student Name: _____ Team No.: _____

1. Software engineering is "the application of a _____, _____, _____ approach to the development, operation, and maintenance of software, that is, the application of engineering to software."
 - a. asystematic, ad-hoc, quantifiable.
 - b. asystematic, disciplined, quantifiable.
 - c. asystematic, ad-hoc, qualifiable.
 - d. systematic, disciplined, qualifiable.
 - e. systematic, disciplined, quantifiable.

2. SWEBOK stands for _____.
 - a. Software Engineering Bureau of Knowledge.
 - b. Software Engineering Book of Knowledge.
 - c. Software World Engineering Body of Knowledge.
 - d. Software World Engineering Book of Knowledge.
 - e. Software Engineering Body of Knowledge.

3. Which of the following SWEBOK Key Area (KA) is NOT **directly** related to software development?
 - a. Software requirements.
 - b. Software design.
 - c. Software testing.
 - d. Software construction.
 - e. Software users.

4. Which of the following statement is FALSE about software requirements?
 - a. Non-functional requirements may be more critical than functional requirements.
 - b. They are related to the product, organization, and external requirements.
 - c. The cost for fixing requirements errors is high so validation is very important.
 - d. Generally, functional user requirements should describe the system functionality in detail.
 - e. If non-functional requirements are not met, products are useless.

5. Which of the following UML diagram is NOT a behavioral diagram?
 - a. Use case diagram.
 - b. Activity diagram.
 - c. Statechart diagram.
 - d. Class diagram.
 - e. Sequence diagram.

6. Which of the following statement is FALSE about use cases?
 - a. Use case can be used for product testing.
 - b. Use cases provide the basis of communication between clients and developers in early phase.
 - c. A fully-dressed use case should include "whats" so that they are ready for "realization."
 - d. A use case is an interaction between a user and a system.
 - e. Use cases describe the structure of the system.

7. Which of the following statement is FALSE about use cases?
- Use cases capture functional requirements.
 - A use case may have multiple scenarios.
 - Use cases are generally initiated by an actor.
 - Good source for identifying use cases is external events.
 - Use cases must be developed after identifying classes.
8. Which statement is FALSE about an actor?
- An actor plays a role as user with respect to the system.
 - Generalization is applicable to actors.
 - An actor does not need to be human.
 - A subsystem or external system can be modeled as an actor.
 - An actor becomes a class in the design phase.
9. Which of the following statement is FALSE about UML?
- UML is a visual modeling language that enables software engineers to create blueprints that capture their visions in a standard, easy-to-understand way, and provides a mechanism to effectively share and communicate with others.
 - IEEE maintains UML.
 - Because UML is a language, it has rules for combining graphical elements.
 - The purpose of the UML diagrams is to present multiple views of a software product.
 - UML has become a de facto standard in software industry, and it continues to evolve.
10. Which of the following statement is FALSE about activity diagrams?
- Activity diagrams can be used to depict workflow for a particular business activity.
 - Activity diagrams can be used to explore/discover parallel activities.
 - Activity diagrams do not tell you which object does what and are difficult to trace back to objects.
 - Activity diagrams can be used to depict realize use cases.
 - Activity diagrams can be used to describe the structure of a system.
11. Which of the following statement is FALSE about use-case driven development?
- Requirements are primarily captured in use cases.
 - Use cases are the essential part of iterative planning by choosing some use case scenarios.
 - Use cases are key input to project sizing.
 - User manuals are normally organized based on use cases.
 - Use case diagrams include uses cases, classes, and actors.
12. One use case scenario is:
- One particular set of interactions between a specific use case and actors.
 - A generic description of the overall functionality of a product.
 - The overall interaction between the classes of the software product and the actors (users) of the product.
 - A description of the actions performed by or to each class or subclass.
 - A description of the classes, their attributes and behaviors.
13. Which of the following statement is FALSE about classes and class diagrams?
- Class diagrams are one of the most fundamental diagrams in UML.
 - Class diagrams are used to capture the static relationships of software.
 - A class represents a group of things that have common state and behavior.

- d. Each specific type of a class is called an object.
 - e. Class diagrams focus on the execution and flow of the behavior of a system.
14. Which of the following statement is FALSE about sequence diagrams?
- a. Sequence diagrams emphasize the communication between objects.
 - b. Sequence diagrams focus on the execution and flow of the behavior of a system.
 - c. A lifeline (a dashed line dropping down from a rectangle) shows how long the object is actually in existence.
 - d. Place an X at the bottom of the lifeline where the object ceases to exist.
 - e. Focus of control indicates that an object is busy for some period of time.
15. In object-oriented analysis, a CRC card contains:
- a. constraints, requirements and containers.
 - b. classes, responsibilities and collaborators.
 - c. conditions, relationships and code.
 - d. cohesion, review and consistency.
 - e. classes, relationships and collaborators.
16. In which of the following circumstances might the rapid prototyping model be an appropriate one to use?
- a. large-scale, in-house products
 - b. for products with complex user-interface where it is important to meet a particular client's needs
 - c. real time systems
 - d. object-oriented systems
 - e. short programs that will not need maintenance
17. In which of the following circumstances might the **waterfall** model be an appropriate one to use?
- a. large-scale, in-house products
 - b. for products utilizing an open architecture and complex user-interface
 - c. real time systems
 - d. object-oriented systems
 - e. large document-driven projects requiring a disciplined approach
18. Which feature of the **waterfall** model allows modifications to be made to the deliverables produced at an earlier stage?
- a. the cascade effect
 - b. re-selection of the best solution, of those developed in parallel, from the previous stage
 - c. risk analysis and risk resolution
 - d. no modifications can be made as each stage once signed off by the clients and developers becomes definitive
 - e. feedback loops
19. A **critical** point of the **waterfall** software life-cycle model is that:
- a. the cascade effect ensures that the product gains momentum through each phase in its development thus ensuring buy-in from the clients
 - b. re-selection of the best solution, of those developed in parallel, from the previous stage
 - c. risk analysis and risk resolution are explicitly defined for each stage
 - d. no modifications can be made as each stage once signed off by the clients and developers becomes definitive

- e. no phase is complete until the documentation for that phase has been completed and the deliverables for that phase have been approved by the SQA group
20. How is maintenance treated in the **spiral model**?
- a. Maintenance is not treated at all. This is one of the major limitations of the spiral model.
 - b. Maintenance is the last of the four quadrants in the model.
 - c. Maintenance is the first of the four quadrants in the model.
 - d. Maintenance is treated within each quadrant of the model.
 - e. Maintenance is treated as a further iteration of the model.
21. To a certain extent the use of the **rapid prototyping** model is to overcome one of the weaknesses of the waterfall model. To this extent the working model is primarily designed to be evaluated by the:
- a. developers
 - b. analysts
 - c. test team
 - d. managers
 - e. end-users
22. Which of the followings is NOT an advantage of the **incremental and iteration model**?
- a. Change and adaptation are natural to the model.
 - b. An operational quality product is delivered to the client at each stage.
 - c. As the design is flexible enough to support incremental builds, enhancements undertaken during the maintenance phase can be easily incorporated as they are additional builds.
 - d. It reduces the traumatic effect of introducing changes to the organization as each new increment builds upon the work delivered in the previous one.
 - e. No phase is complete until the documentation for that phase has been completed.
23. A simple way of looking at the **spiral** model is as a waterfall model with each phase proceeded by:
- a. code-and-fix
 - b. freezing
 - c. synchronization
 - d. testing
 - e. risk analysis
24. Which of the following is NOT a limitation of the **spiral model**?
- a. It is suited to internal development. For instance, termination of a contract with an external developer may lead to costly litigation.
 - b. For small projects the cost of conducting risk analysis may not be warranted when put alongside the total cost of the product.
 - c. Members of the development team may not be competent risk analysts.
 - d. It is complex and relatively difficult to follow strictly.
 - e. It promotes quality assurance through prototyping at each stage in systems development.
25. During which phase of the software life cycle do you first consider maintenance?
- a. implementation
 - b. testing
 - c. requirement
 - d. from the very beginning
 - e. design