Q1. Which of the following are aims of UML?

a. To model system using OO concepts

b. To provide a process for software development

c. To support small-scale and large-scale analysis and design

d. To provide an insight into implementation mechanism

1. a, c

2. a, b

3. a, b, d

**4. a, c, d**

Q2. In which of the following phases of use-case driven process do you think use cases have a role?

a. requirement capture

b. analysis

c. design

d. implementation

e. test

1. a, b, c

2. a, b, c, d

3. b, d

**4. a, b, c, e**

Q3. If you are finding hard to identify the name of class and to write definition for it. What thing you should do?

1. ignore class completely

**2. do more analysis to get a better understanding of what is invaloved in the class**

3. write a definition for the class even if it is not very good

4. make it a friend class of some other main class

Q4. Which of the following are possible actors?

a. data inputter

b. GUI component

c. Another system

d. A printer

1. a, b, c

2. a, b, c, d

**3. a, b, d**

4. a, c

Q5. In use-case diagram, what is system illustrated by?

1. oval

**2. box**

3. circle

4. triangle

Q6. UML can be used as a way to represent only OO software systems

Correct Answer : F

Q7. Use cases can be included in any type of collaboration diagrams.

Correct Answer : F

Q8. collaboration diagram represents

**1. organization of objects**

2. messages on time scale

3. conceptual design

4. set of actions

Q9. In OOD primary abstraction mechanism is \_\_\_\_\_\_\_\_\_\_

1. function

**2. class**

3. object

4. hierarchy

Q10. requirement analysis \_\_\_\_\_\_\_\_\_\_\_\_\_

1. delivers a system in a series of versions

2. organizes abstraction

**3. builds a bridge between user and developer**

4. uses experimental software to better understand user requirements

Q11. polymorphism \_\_\_\_\_\_\_\_\_\_\_\_

1. organizes abstraction

2. builds a bridge between user and developer

3. delivers a system in a series of versions

**4. works with encapsulation and inheriatance to simplify flow of control**

Q12. prototyping model \_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. delivers a system in a series of versions

2. builds a bridge between user and developer

**3. uses experimental software to better understand user requirements**

4. works with encapsulation and inheriatance to simplify flow of control

Q13. storage management is not a part of version management

Correct Answer : F

Q14. data flow diagrams are part of design phase of SDLC

Correct Answer : T

Q15. Which of the following is reason of project failure?

1. finite resources

**2. inaccurate estimates of cost and time**

3. others are competing to do the job cheaper and faster

4. none of the above

Q16. What manifests in the patterns of choices made among alternative ways of expressing an algorithm is

1. a data flow diagram

2. coding style

3. a data dictionary

**4. a flow chart**

Q17. \_\_\_\_\_\_\_\_\_\_\_\_\_\_ is method for estimating software

1. COCOMO

2. function point analysis

3. use case estimation

**4. all of the above**

Q18. pickup odd one out of the following

**1. component assembly model**

2. spiral model

3. incremental model

4. iterative model

Q19. which of the following types of test plans is most likely to arise from requirement specification process?

1. system integration testing plan

**2. acceptance test plan**

3. sub-system integration test plan

4. module test plan

Q20. pick up the odd one out of the following

1. data flow diagram

**2. object identification**

3. structural decomposition

4. E-R diagrams

Q21. Parts of design principle are

1. correctness, robustness, efficiency, flexibility, understandable

**2. correctness, robustness, efficiency, flexibility, reusibility**

3. flexibility, correctness, robustness, efficiency, standard

4. flexibility, correctness, robustness, efficiency, security

Q22. largest time is spent on which of the software development phase?

1. testing

**2. enhancement**

3. bug fixing

4. analysis and design

Q23. Which of the following can be a reason for project failure?

1. Finite resources

**2. Inaccurate estimates of cost & time**

3. Others competing to do the job cheaper & faster.

4. None of the above

Q24. \_\_\_\_\_\_\_\_\_\_\_ is a method for estimating the software

1. COCOMO

2. Function Point Analysis

3. Use Case Estimation

**4. All of the above**

Q25. Quality control

**1. focuses on inspections, testing & removal of defects before release**

2. is a set of planned & strategic actions to provide confidence that a product or service will satisfy requirements of quality

3. is to check system for internal errors.

4. All of the above.

Q26. The elements of the software architecture of a computing

system include

1. software components

2. class diagrams

3. connectors expressing relationships between software components

4. entity relationship diagrams

1. 1 & 2

**2. 1 & 3**

3. 1, 3 & 4

4. 1, 2, 3 & 4

Q27. Pick the odd one out

**1. Component assembly model**

2. Spiral Model

3. Incremental Model

4. Iterative Model

Q28. Software Engineering is concerned with \_\_\_.

1. process

2. methods

3. tools

**4. all of the above**

Q29. Ability to deal with exceptional conditions e.g. invalid input, improper handling, power failure, disk crash etc.

1. Effeciency

**2. Robustness**

3. Reliability

4. Correctness

Q30. Maintainability is the ease with which a software can

1. be corrected if an error is encountered

2. adapted if its environment changes

3. enhanced if the customer desires a change in requirements

**4. all of above**

Q31. Which of the following factors of a Software Product may not contribute much directly to its maintainibility?

1. Understandability

2. Flexibility

**3. Security**

4. Testability

Q32. The Software Development Life Cycle covers activities from

1. Feasibility Study to Installation

**2. Requirements Phase to Testing**

3. Requirements Phase to Maintenance

4. Project Initiation to Software Retirement

Q33. An approved feasibility study is a deliverable out of

1. Systems design

**2. Preliminary investigation**

3. Systems development

4. Systems analysis

Q34. The type of testing carried out along with coding is called

1. system testing

**2. unit testing**

3. pretesting

4. stress testing

Q35. Checklists, grid charts, and decision tables are all tools used in the \_\_\_\_\_\_\_\_\_\_ step

1. preliminary investigation

**2. systems analysis**

3. systems development

4. systems implementation

Q36. The present system is studied in depth during the \_\_\_\_\_\_\_\_\_\_ phase of the systems life cycle.

1. preliminary investigation

**2. systems analysis**

3. systems design

4. systems development

Q37. The type of software maintainence which is done to remove bugs or defects in the software is called

**1. Corrective Maintainence**

2. Adaptive Maintainence

3. Regressive Maintainence

4. Perfective Maintainence

Q38. The SDLC Model most suitable for small projects with clear requirements is

1. Spiral Model

2. Incremental Model

**3. Waterfall Model**

4. Prototyping Model

Q39. The SDLC Model most suitable for small projects with unclear requirements is but not many technical risks is

1. Spiral Model

2. Incremental Model

3. Waterfall Model

**4. Prototyping Model**

Q40. Because of the cascade from one phase to another, the model of software development process is known as

1. Evolutionary model

2. Formal model

**3. Waterfall model**

4. None of the above

Q41. RAD Model is high speed implementation of

**1. Waterfall Model**

2. Spiral Model

3. Prototyping model

4. Component Assembly model

Q42. Which of the following is not a feature of RAD

1. Well understood, constrained & modularizable requirements

2. Component based construction & use of 4 GL

3. Use of multiple teams each developing separate function

**4. Project has high technical risks**

Q43. The majority of the lifetime of a program is spent in the \_\_\_\_\_\_\_\_\_\_ phase

**1. Maintenance**

2. Analysis

3. Design

4. Testing

Q44. Pick up the odd one out of the following process models

1. Component assembly model

2. Prototypiong Model

3. Spiral model

**4. Waterfall Model**

Q45. \_\_\_\_\_\_\_\_\_\_ uses powerful development software and small, highly trained teams of programmers.

1. Prototyping

**2. RAD**

3. Coding

4. Modeling

Q46. Arrange the following Requirements subphases in the correct order a.Documentation b. Analysis c. Validation d. Elicitation

1. a, b, c, d

**2. d, b, a, c**

3. d, c, a, b

4. b,a, d c

Q47. External Entities in a Context Diagram may be A) People B) Other Software Systems C) Hardware D) Databases

1. Only A & D

2. Only B & C

3. Only A, B & D

**4. A,B, C & D**

Q48. Which of the following is seen in the DFD but not in the Context Diagram

1. Data Sources

2. Data Flows

**3. Data Stores**

4. Users

Q49. DFD gives idea about flow of \_\_\_\_\_\_\_\_ & flowchart gives idea of the flow of \_\_\_\_\_\_\_\_\_\_\_\_

1. processes, decisions

2. control, data

3. logic, control

**4. data, control**

Q50. A data flow diagram is not a

1. logical model of a system

2. good guide to a system

**3. representation of the physical system**

4. All of these options

Q51. Example of a Semantic Data model is

1. data flow diagram

2. Context Diagram

**3. Entity Relationship Diagram**

4. all of the above

Q52. Automated CASE tools like PSL/PSA do not help in

1. Requirements Documentation

2. Requirements Validation

3. Requirements Analysis

**4. Requirements Elicitation**

Q53. Which of the following is not a characteristic of a good SRS document?

1. Unambigious

2. Verifiable

**3. Redundant**

4. Consistent

Q54. The ways of describing specifications at different levels of detail include

1. requirements definition

2. requirements specification

**3. both a and b options**

4. None of these options

Q55. A system developed to give end users a concrete impression of the system capabilities is called

1. Semantics

2. model

**3. prototype**

4. abstraction

Q56. The requirement engineering process has the following stages, Except

1. Feasibility study

2. Requirement analysis

**3. Implementation**

4. Requirement definition

Q57. Notations used to specify the external characteristics, architectural structure, and processing details of a software system include I. Data Flow Diagrams II. HIPO diagrams III. Structure Charts

1. I and II Only

2. III Only

**3. I, II and III**

4. None of the above

Q58. Find the odd one out

1. Axiomatic Specification

2. Algebraic Specification

3. Z Specification

**4. Data Flow Diagram**

Q59. Planning the modular program structure & control relationships between modules is called

1. Architechtural Design

2. High Level Design

3. System Design

**4. all of the above**

Q60. Conception & planning out of externally observable characteristics of a software is called

1. External Design

2. User Interface Design

**3. Both a and b options**

4. None of the above

Q61. Concept of Abstraction is used in

1. Reuirements phase

2. Design Phase

3. Testing Phase

**4. all of the above**

Q62. Providing a logical reference to the data object without concern for the underlying representation is

1. Procedural Abstraction

**2. Data Abstraction**

3. Control Abstraction

4. None of the above

Q63. The number of subordinate modules controlled by a module is called its

1. control range

**2. fan out**

3. fan in

4. width

Q64. Functional Independence is not achieved by

**1. Coupling**

2. Modularity

3. Information Hiding

4. Any of the above

Q65. If two modules pass a data structure across their interface they exhibit

**1. Stamp Coupling**

2. Data Coupling

3. Content Coupling

4. Control Coupling

Q66. Use of global data areas or global variables may lead to

1. Stamp Coupling

**2. Common Coupling**

3. Content Coupling

4. Control Coupling

Q67. The strength of relationship between which of the following elements of a module is examined to evaluate module cohesion

1. function declarations, function definations & calls

2. variable declarations

3. data definitions

**4. all of the above**

Q68. Which is the most undesirable form of cohesion from the following options

1. Sequential

**2. Coincidental**

3. Temporal

4. Communicational

Q69. The graphical tool commonly used to represent the system architechture is called

1. Context Diagram

**2. Structure Chart**

3. Architechtural Plan

4. Event Table

Q70. The method of deriving the structure chart from the DFD is called

1. Factoring

2. Factor Analysis

**3. Transform Analysis**

4. all of the above

Q71. Which iof the following is true about structure chart notations?

1. There should be only one module at the top

2. There should be at the most one control arrow between two modules

3. The sequence or order of tasks is not represented

**4. All of the above**

Q72. Using \_\_\_\_\_\_\_\_\_\_ a programmer can detail the logic of the program

**1. pseudocode**

2. software

3. context diagram

4. data flow diagram

Q73. The external interface design process should be \_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. developer centered

**2. user centered**

3. administrator centered

4. management centered

Q74. Typographical errors and/or incorrect use of the programming language is referred to as

1. logic errors

**2. syntax errors**

3. run time errors

4. A bug

Q75. \_\_\_\_\_\_\_\_\_\_\_\_\_\_ is a programming method which combines data and instructions for processing that data into a self-sufficient block that can be used in other programs.

1. modular programming

2. top down design

**3. object oriented programming**

4. structured programming

Q76. \_\_\_\_\_\_\_ is the process of locating and eliminating program errors.

1. editing

2. correcting

**3. debugging**

4. testing

Q77. Changes made to the software to accommodate changes to its environment is called

1. perfective maintainence

2. regressive maintainence

**3. adaptive maintainence**

4. corrective maintainence

Q78. Changes made to the software to extend it beyond its original functionality is called

**1. perfective maintainence**

2. regressive maintainence

3. adaptive maintainence

4. corrective maintainence

Q79. Major changes made to software after long periods is also called software reengineering or

1. perfective maintainence

**2. regressive maintainence**

3. adaptive maintainence

4. corrective maintainence

Q80. Effective Software Project Management focusses on

1. People

2. Problem

3. Process

**4. all of above**

Q81. Arrang the following in the correct sequence of software estimation a. Schedule Estimation b. Effort Estimation c. Cost Estimation d. Size estimation

1. b, c, a, d

2. c, a, b, d

**3. d, b, a, c**

4. a, c, d, b

Q82. Final Function point count calculated for project will result in the smallest LOC if implemented in

1. Assembly

2. C

3. C++

**4. Visual Basic**

Q83. The value of COCOMO cost driver attribute for lower than average Reliability requirement will be

1. Greater than 1

2. Equal to 1

**3. Less than 1**

4. None of these

Q84. The crtitcal path of PERT/CPM chart cannot be

1. the path with the longest duration

2. more than one unique path

**3. path on which any delays are allowed**

4. path with same earliest and latest starts for all activites

Q85. Project schedule can be illustrated using

1. DFD and ERD

2. Bar chart

3. Activity chart

**4. Both b and c options**

Q86. The total float for an activity is

1. the total duration of the activity

2. the difference between the earliest finish time and earliest start time

**3. the difference between the latest finish time and the earliest finish time**

4. the difference between the latest finish time and the earliest start time

Q87. \_\_\_\_\_\_\_\_ shows the dependencies between the different activities making up a project.

**1. PERT chart**

2. Bar chart

3. Staffing Plan

4. Pi chart

Q88. The minimum time required to finish the project can be estimated by considering the \_\_\_\_\_\_ path in the activity graph

1. Shortest

**2. Longest**

3. Average

4. SPT

Q89. Which of the following is true as per Putnam model

1. Staffing Pattern peaks at Coding & Unit testing

2. Schedule compression increases effort in proportion to fourth power

3. Expanding the schedule gives extreme saving in effort

**4. all of the above**

Q90. Chief Programmer Teams are suitable for projects

1. with research orientation

**2. with high modularity**

3. with high creativity

4. None of these

Q91. Which of the follwing are Software Risk Components

1. Performance

2. Cost

3. Schedule

**4. all of the above**

Q92. Arrange the following activities in Risk Assesment in the correct sequence a. Prioritization b. Identification c. Analysis

1. b, a, c

**2. b, c, a**

3. a, b, c

4. c, a, b

Q93. Risk Assesment Table is based on categorization by

1. Risk Components

2. Risk Impact

**3. Both a and b options**

4. None of the above

Q94. Judging the seriousness of a risk by evaluating its probability along with its consequences is called

1. Risk analysis

2. Risk Projection

3. Risk Estimation

**4. all of the above**

Q95. The RMMM plan is generally included in the

1. Feasibility Study

**2. Project Plan**

3. SRS Document

4. Project Legacy

Q96. Example of Software Configuration Items (SCI) is

1. SRS

2. Code

3. User manual

**4. all of the above**

Q97. A change request has to be evaluated for

1. its technical merit

2. cost & schedule impacts

3. side effects

**4. All of these options**

Q98. \_\_\_\_\_\_\_\_\_\_\_ ensures that a set procedure is followed to make any changes to the software

1. Configuration Identification

**2. Configuration Control**

3. Baselining

4. all of the above

Q99. Configuration Management is

1. framework actvity

2. umbrella activity

**3. one time activity**

4. None of the above

Q100. As per SEI CMM oganizations which do not have any KPAs present & stable are considered at

**1. Level 1**

2. Level 2

3. Level 3

4. Level 4