**Written Examination Question Paper**

**Module Name: Software Engineering Principles**

1. Which of the following is not an SDLC phase? (Select only one option)
   1. Design
   2. Coding
   3. Testing
   4. Refactoring
2. Mary has a problem. She is working on a project that has fixed requirements; it is not a mission critical project. But it focuses strictly on Documentation and planning. (Select only one option)
3. Waterfall model
4. Spiral model
5. Agile Model
6. RAD model
7. Select all the requirement gathering techniques that you have come across? (Select all that apply)
8. Questionnaires
9. Interviews
10. Use Case Modeling
11. Data Flow diagram
12. Select the wrong option from the options mentioned below about requirements? (Select only one option)
13. Requirement is a statement of need.
14. It describes the application's behavior and functionalities.
15. It gives guidelines for use of ADL Languages
16. Requirements helps developers understand what to build, and tells testers what they should test.
17. Select the features about the agile model? (Select three options)
18. Cross Functional Team
19. Effective Communication
20. Hours and Hours of planning
21. Responsiveness to change
22. Simon is on a project that is mission critical. It is a Research & Development based project. What model do you recommend for his project considering the fact that risk analysis is an important phase for his project? Select only one option)
23. Waterfall Model
24. Spiral Model
25. Agile
26. RAD model
27. Select the three roles that are a part of the agile strategy called as SCRUM. (Select three options)
28. Product Owner
29. Scrum Master
30. Team
31. Business analyst
32. Sprint backlog is a subset of your product Backlog? (STATE TRUE OR FALSE)
33. TRUE
34. FALSE
35. Agile methodology states that: Focus on customer collaboration over contract negotiation. (STATE TRUE OR FALSE)
36. TRUE
37. FALSE
38. Select one option that is not true about the agile methodology.
39. Focus on responsiveness to change over following a fixed plan.
40. Focuses on effective face to face communication.
41. Focus on building working software over comprehensive documentation.
42. Focuses on extensive documentation and planning.
43. Extreme programming is a type of spiral model. (STATE TRUE OR FALSE)
44. TRUE
45. FALSE
46. The waterfall model is an orthodox model that does not respond well to changes in requirements. (STATE TRUE OR FALSE)
47. TRUE
48. FALSE
49. Test Driven Development focuses on developing code first and then testing it later rigorously. (STATE TRUE OR FALSE)
50. TRUE
51. FALSE
52. Select what is not true about Coupling? (Select only one option)
53. Coupling is the degree of interdependency between the modules.
54. High degree of coupling is highly desirable.
55. High degree of coupling is not desirable.
56. Select the wrong option from the below mentioned options about Design Concepts. (Select only one option)
57. Refinement and Abstraction are complementary concepts.
58. Abstraction permits one to concentrate on a problem at some level of generalization without regard to irrelevant low level details.
59. Modularity focuses on the Divide and Conquer rule.
60. Modularity focuses on treating the program in a monolithic fashion.
61. Which of the following are the objectives of testing? (Select only three options.)
62. Meeting the standards and Guidelines.
63. Improving Quality
64. Filling in the time between Development and Product Delivery.
65. Providing stability to the system.
66. Which of the following are levels of testing? (Select three valid options)
67. Unit Testing
68. Component Integration testing
69. System Testing
70. Black Box testing
71. When a defect is found for the first time, it is labeled as a NEW defect. (STATE TRUE OR FALSE)
72. TRUE
73. FALSE
74. Select what is true about Black Box testing? (Select only one option)
75. It is done from the end user point of view.
76. It requires knowledge about the internal details of the system.
77. It is done by developers only.
78. Black box testing and white box testing are one and the same.
79. Select the valid software quality attributes. (Select all that apply)
80. Portability
81. Efficiency
82. Reliability
83. Functionality
84. Gozinto chart is a subset of the Work Break Down structure. (TRUE OR FALSE)
85. TRUE
86. FALSE
87. Linear Responsibility Matrix shows the relationship of personnel (who is responsible for what) and to identify where special coordination is necessary. (TRUE OR FALSE)
88. TRUE
89. FALSE
90. Which of the following are known and predictable risks? (Select Three option only)
91. Business Risks
92. Product size Risks
93. Failure in future due to unpredictable circumstances.
94. Technology to be built
95. Risk Mitigation is a risk avoidance activity. (TRUE OR FALSE)
96. TRUE
97. FALSE
98. Choose the correct option about Risks. (Select three options only)
99. Likelihood means the probability that the risk would occur.
100. Risk Impact means the impact that the risk would cause.
101. It is good to have a Contingency plan for risks.
102. It is not a good practice to have a contingency plan for risks.
103. Refactoring means cleaning up code. (TRUE OR FALSE)
104. TRUE
105. FALSE
106. Which of the following is correct about TEST DRIVEN DEVELOPMENT? (Select all the options that apply)
107. TDD produces fully testable code.
108. TDD gets running code very quickly.
109. Your code will always meet the design specifications.
110. TDD focuses on writing unit test cases first, and then developing the code.
111. Sequence Diagram and Collaboration Diagram are interaction diagrams. (TRUE OR FALSE)
112. TRUE
113. FALSE
114. Use Case diagram can be used as a tool for requirement gathering. (TRUE OR FALSE)
115. TRUE
116. FALSE
117. Which of the following is a correct feature about Use Case diagrams? (Select all the options that apply)
118. It helps in requirement analysis and high level design.
119. It models the context of a system.
120. Forward Engineering
121. Reverse Engineering
122. Does an organization develop one life cycle model?
123. for all the projects
124. for each project
125. for each domain
126. Pick up the odd one out of the following:
127. Software Design
128. Software Testing
129. Software Quality Assurance
130. Software requirements should not be.
131. Functional
132. Ambiguous
133. Consistent
134. Find the odd one out of the following:
135. stepwise refinement
136. structural design
137. information hiding
138. What manifests in the patterns of choices made among alternatives ways of expressing an algorithm is
139. a data flow diagram
140. coding style
141. a data dictionary
142. The decision logic is expressed by
143. data flow diagram
144. flow chart
145. structure chart
146. Validation is to check
147. Whether we are building the product right.
148. Whether we are building the right product.
149. The methodology of software development.
150. Corrective maintenance is to
151. Improve the system in some way without changing its functionality.
152. Correct the undiscovered errors.
153. Make changes in the environment.
154. Quality control
155. Focuses on inspections, testing and removal of defects before release.
156. Is a set of planned and systematic actions to provide confidence that a product or service will satisfy given requirements for quality.
157. Is to check the system for its interface errors.
158. Capability maturity model
159. Gives prescription for software process.
160. States what activities are necessary for success.
161. Describes how activities are to be performed.
162. Which software development model incorporates risk management?
163. waterfall model
164. spiral model
165. incremental model
166. Analysis phase is
167. not to actually solve the problem
168. not to determine exactly what must be done to solve the problem
169. to move quickly to program design
170. A data flow diagram is not a
171. logical model of a system
172. good guide to a system
173. representation of a physical system
174. Four important characteristics of a software product are
175. dependability, usability, reliability, robustness
176. maintainability, dependability, efficiency, usability
177. Supportability, maintainability, visibility, rapidity
178. Object models
179. should include details of the individual objects in the system
180. are part of design
181. are natural ways of reflecting the real world entities that are manipulated by the system?
182. Pick up the odd one out of the following:
183. data flow design
184. object identification
185. structural decomposition
186. Pick up one of the testing methods given below that is part of white-box testing:
187. Equivalence partitioning
188. boundary value analysis
189. basis path testing
190. The three classes of interface errors are:
191. interface misuse, interface misunderstanding, timing errors
192. interface misunderstanding, interface coupling, data transfer errors
193. interface coupling, timing errors, interface parameter errors
194. Find the activity which is not part of version management
195. controlled change
196. storage management
197. coding standard
198. Which is the non-technical factor of maintenance cost?
199. program age
200. programming style
201. program validation
202. Pick up the odd one out of the following process models
203. Component assembly model
204. Incremental model
205. Spiral model
206. Software quality assurance is
207. a multitier testing strategy
208. a measurement and reporting mechanism
209. an activity that is applied throughout the software process.
210. Verification is to check
211. whether we are building the right product
212. whether we are building the product right
213. neither of the above
214. Adaptive maintenance is
215. to improve the system in some way without changing its functionality.
216. the maintenance due to the changes in the environment.
217. the correction of undiscovered system errors.
218. Most common but least effective way of debugging is
219. brute force
220. backtracking
221. cause elimination
222. Equivalence partitioning is
223. a white-box testing method
224. a black-box testing method
225. neither white-box nor black-box testing method
226. Pick up the correct sequence of processes
227. Requirements, Analysis, Test case design, Design
228. Requirements, Test case design, Analysis, Design
229. Requirements, Analysis, Design, Test case design
230. Doing what is said one would do, is the definition for
231. Reliability
232. Quality
233. software plan
234. The typical elements of the requirements engineering process are
     * 1. Problem analysis
       2. software design
       3. Analysis of staffing needs
       4. External behaviour specification
235. i and iv
236. ii and iii
237. i, iii and iv
238. i, ii and iii
239. In object models, information hiding conceals
240. Operations
241. Attributes
242. Methods
243. state and behaviour