**Written Examination Question Paper**

**Module Name: Software Engineering Principles**

1. Does an organization develop one life cycle model?
2. for all the projects
3. for each project
4. for each domain
5. Pick up the odd one out of the following:
6. Software Design
7. Software Testing
8. Software Quality Assurance
9. Software requirements should not be.
10. Functional
11. Ambiguous
12. Consistent
13. Find the odd one out of the following:
14. stepwise refinement
15. structural design
16. information hiding
17. What manifests in the patterns of choices made among alternatives ways of expressing an algorithm is
18. a data flow diagram
19. coding style
20. a data dictionary
21. The decision logic is expressed by
22. data flow diagram
23. flow chart
24. structure chart
25. Validation is to check
26. Whether we are building the product right.
27. Whether we are building the right product.
28. The methodology of software development.
29. Corrective maintenance is to
30. Improve the system in some way without changing its functionality.
31. Correct the undiscovered errors.
32. Make changes in the environment.
33. Quality control
34. Focuses on inspections, testing and removal of defects before release.
35. Is a set of planned and systematic actions to provide confidence that a product or service will satisfy given requirements for quality.
36. Is to check the system for its interface errors.
37. Capability maturity model
38. Gives prescription for software process.
39. States what activities are necessary for success.
40. Describes how activities are to be performed.
41. Which software development model incorporates risk management?
42. waterfall model
43. spiral model
44. incremental model
45. Analysis phase is
46. not to actually solve the problem
47. not to determine exactly what must be done to solve the problem
48. to move quickly to program design
49. A data flow diagram is not a
50. logical model of a system
51. good guide to a system
52. representation of a physical system
53. Four important characteristics of a software product are
54. dependability, usability, reliability, robustness
55. maintainability, dependability, efficiency, usability
56. Supportability, maintainability, visibility, rapidity
57. Object models
58. should include details of the individual objects in the system
59. are part of design
60. are natural ways of reflecting the real world entities that are manipulated by the system?
61. Pick up the odd one out of the following:
62. data flow design
63. object identification
64. structural decomposition
65. Pick up one of the testing methods given below that is part of white-box testing:
66. Equivalence partitioning
67. boundary value analysis
68. basis path testing
69. The three classes of interface errors are:
70. interface misuse, interface misunderstanding, timing errors
71. interface misunderstanding, interface coupling, data transfer errors
72. interface coupling, timing errors, interface parameter errors
73. Find the activity which is not part of version management
74. controlled change
75. storage management
76. coding standard
77. Which is the non-technical factor of maintenance cost?
78. program age
79. programming style
80. program validation
81. Pick up the odd one out of the following process models
82. Component assembly model
83. Incremental model
84. Spiral model
85. Software quality assurance is
86. a multitier testing strategy
87. a measurement and reporting mechanism
88. an activity that is applied throughout the software process.
89. Verification is to check
90. whether we are building the right product
91. whether we are building the product right
92. neither of the above
93. Adaptive maintenance is
94. to improve the system in some way without changing its functionality.
95. the maintenance due to the changes in the environment.
96. the correction of undiscovered system errors.
97. Most common but least effective way of debugging is
98. brute force
99. backtracking
100. cause elimination
101. Equivalence partitioning is
102. a white-box testing method
103. a black-box testing method
104. neither white-box nor black-box testing method
105. Pick up the correct sequence of processes
106. Requirements, Analysis, Test case design, Design
107. Requirements, Test case design, Analysis, Design
108. Requirements, Analysis, Design, Test case design
109. Doing what is said one would do, is the definition for
110. Reliability
111. Quality
112. software plan
113. The typical elements of the requirements engineering process are
     * 1. Problem analysis
       2. software design
       3. Analysis of staffing needs
       4. External behaviour specification
114. i and iv
115. ii and iii
116. i, iii and iv
117. i, ii and iii
118. In object models, information hiding conceals
119. Operations
120. Attributes
121. Methods
122. state and behaviour