

28/11/2018 (Regular)



5th Sem. B.Tech & B.Tech Dual Deg.
SE IT-3003
(E&CS, CSE, IT, CSCE, CSSE)

AUTUMN END SEMESTER EXAMINATION-2018
5th Semester B.Tech & B.Tech Dual Degree
SOFTWARE ENGINEERING
IT-3003

[For 2017(L.E.), 2016 & Previous Admitted Batches]

Time: 3 Hours

Full Marks: 60

Answer any Six questions including question No.1 which is compulsory.

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.

1. (a) Explain if the following statement true or false -There are [2×10]
well defined steps through which a problem is solved in
exploratory style.
- (b) Without developing SRS document, an organization
might face severe problems. (True / False)Justify.
- (c) What are the non-functional requirements which need to
be captured well advance in a software project?
- (d) Explain the role of abstraction in software design.
- (e) List the advantages ad limitations of prototype life cycle
model.
- (f) "Statement coverage based testing is stronger testing
strategy than branch coverage based testing." (True/ False)
Justify your answer.
- (g) How debugging is different from testing? List the types of
debugging.
- (h) What do you understand by adaptive maintenance?
Explain with example.

- (i) Differentiate between function oriented design and object oriented design.
- (j) What is the difference between code walkthrough and code inspection?

2. (a) Suppose you are the project manager of a software project requiring the following activities: [4]

Activity No.	Activity Name	Duration (weeks)	Immediate Predecessor
1	Obtain requirement	3	-
2	Analyze operation	3	-
3	Define subsystem	1	1
4	Develop database	3	1
5	Make decision analysis	2	2
6	Identify Constraints	1	5
7	Build Module 1	7	3, 4, 6
8	Build Module 2	11	3, 4, 6
9	Build Module 3	17	3, 4, 6
10	Write report	9	6
11	Integration and Test	7	7, 8, 9
12	Implementation	1	10, 11

- (a) Draw the Activity Network representation of the project.
 - (b) Identify the critical path.
 - (c) Determine Earliest Start, Earliest Finish, and Latest Start, Latest Finish for every task.
- (b) (i) What do you understand by software configuration management? What is the need for it? [4]
- (ii) Explain how Mixed control team is evolved from Chief-programmer team and Democratic team.

3. (a) What do you understand by CMM? Discuss the various Key Process Areas present in various CMM levels. How CMM is different from ISO? [4]
- (b) What are the shortcomings of LOC in software size estimation? In what way Function point metric overcomes them? Explain the factors we consider for function point metric. [4]
4. (a) What do you understand by the term cohesion and coupling in the context of software design? [4]
Explain various types of cohesion and coupling with suitable examples. Explain the desired degree of cohesion and coupling for a good software design.
- (b) i) Perform Structured Analysis and Structured Design by using the operations of a simple lemonade stand. The list of activities are as given below: [4]
Customer Order, Serve Product, Collect Payment, Produce Product, Store Product, Order Raw Materials, Pay for Raw Materials, Pay for Labour
- ii) Briefly discuss the significance of a data dictionary in structured analysis.
5. (a) What are the differences between Agile and Water fall model? Explain the scrum methodology in details. [4]
- (b) Define a test case. Discuss about the following techniques to perform black box testing with suitable example. [4]
- I. Equivalence Class Partitioning
 - II. Boundary value Analysis

6. (a) How Unified Modeling Language is helpful in object-oriented design? Draw a UML Sequence diagram and UML Activity diagram for “*Borrowing a Book from a library*” use case. [4]

- (b) What is Cyclomatic Complexity? Draw the control flow diagram of the below program segment and find the Cyclomatic complexity. [4]

```
int find-maximum (inti, int j, int k)
{
    int max;
    if (i > j) then
        if (j > k) then max = i;
        else max = k;
    else (j > k) then
        If (j > k) then max = j;
        else max = k;
    return (max); }
```

7. (a) Briefly explain the various reliability metrics of software products. [4]

- (b) How software reverse engineering is different from software re-engineering? Discuss about the different cosmetic changes made during the software reverse engineering. [4]

8. Write short notes on any TWO of the following: [4 × 2]

- (a) Verification and Validation.
- (b) Pert Chart and Gantt Chart
- (c) UML Diagrams
- (d) Top down and Bottom Up Integration Testing
