MCA/D07
Software Engineering
MCA -104

Time : 3 Hours **MM:50**

Note:- Attempt Five questions, selecting at least One Question from each unit.

UNIT-I

1 **What is Prototyping? In what types of applications it should be used? How does it differ from spiral model? Also explain its advantages and disadvantages over spiral model.**

2(a) **Explain briefly Putnam Model. What are limitations of this model?**

(b) **Discus why the no. of engineers required for a software projectg cannot be calculated as a simple dividing of effort estimate (in PM) by the nominal time estimate (in months).**

3(a) **What is the aim of software engineering? What does the discipline of software engineering? Discuss.**

(b) **Define function points. Compare LOC and FP metrics. Compute the function points for a project for the following characteristics:**

No. of using inputs : 40
No of using outputs : 2 times of inputs
No of enquires : ½ time of outputs.
No of files : NIL
No of external interfaces : 02
Assume all complexity factors are average.

4(a) **Explain software complexity. List and explain the various factors responsible for software complexity. How complexity can drive through MaCab's rule?**

(b) **What are the rules used for deriving a Gantt chart from a PERT chart, and vice-cersa? How complete are our rules? Explain.**

(c) **Discuss various S/W quality factors.**

5(a) **What is Risk management? List out various risks in developing a project.**

(b) **Discuss various S/w quality factors.**

6 **What is modular coupling? Discuss its various types. Why coupling should be minimum among modules and cohesion maximum within module?**

UNIT-III

7(a) **What do you mean by fundamental of S/W testing? Explain with example.**

(b) **Discuss various levels oftesting.**

8(a) **Discuss the benefits of black-box testing and white-box testing. What are the different black-box and white-box testing techniques?**

9 **Why is S/w maintenance? Why maintenance is time consuming activity? Discuss its various types.**

10 **Explain with example?**

(a) **Top down and Bottom up Design**

(b) **Regression and Acceptance Testing**

(c) **Testing and Debugging.**