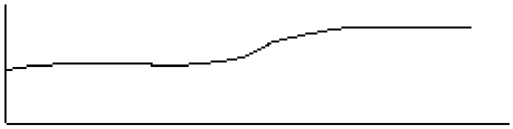
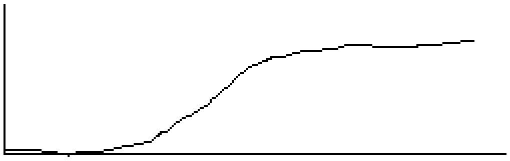
S.No.	
1.	a) What is SE-CMM? Explain all the levels with their associated Key Process Areas. b) It is often claimed that the quality of a design can be assessed in terms of three features: 'control structure', 'modularity' and ‘hierarchy’. Explain
2.	a) A particular software engineer and customer do not see eye-to-eye and their lack of trust during the requirements phase is jeopardizing the project. What issues would you explore to resolve the crisis? b) A water company requires a new computerized billing system and has invited tenders from software companies. You are a member of the team in a software company that is preparing to submit a tender. The team is currently trying to cost the development of the billing system. Your software company has already produced billing systems, but none were as large as the one the water company will require. Outline TWO distinct techniques for attempting to cost the development of this software project. Indicate, with reasons, which you would choose.

3.	a) IF A=B THEN
	X := 7;
	Y := C+D;
	WHILE X<Y DO
	X:=X+1;
	Z:= X+Y
	END WHILE;
	X:=Y;
	END IF;
	For the above given program part
	(i) Make the Flow Graph
	(ii) Make the list of independent paths
	(iii) Calculate cyclomatic complexity separately by using 1. No. of regions 2. Predicate nodes 3. Nodes & edges 4. Graph Matrix

4.	a) What will the following two graphs approximately convey? Justify each answer. (i) Function points in a project versus time (by weeks)



(ii) (Source coding errors)/(function points) versus time (by weeks)



b)

Write a note on the following terms

1. Messaging in OOPs
2. Ethnography in requirement engineering
3. Timeline chart
4. Task set selector value
5. Real Time Data Bases
6. Component reuse

5. Case Study : Super Requirements (Solve the questions at the end in context of the following)

In spring 2000, I was consulting for an e-business solution provider that was struggling over what features to include in its next major product. This company was unusual in that the vice presidents of product development and marketing seemed to respect each other. The development VP trusted the marketing VP’s assessment of the changing market dynamics. The marketing VP trusted the estimates that the development VP provided. Both VPs realized that their employees needed to work together to solve this marketing puzzle instead of attacking each other, as in most companies. The company had already had numerous brainstorming sessions with their existing and potential customers, which resulted in around 200 candidate requirements. To satisfy all those requirements would take around 18 months, but the customers who were depending on their foundation solution needed it in just six months. My first recommendation was to begin with fewer requirements. It would simply take too long to perform triage on all 200. I suggested that they group the requirements into small sets of features or super-requirements—that they could sell as a package to aid a customer in satisfying one or more business goals. The requirements within each set also had to be closely related from an implementation perspective. That is, the developers could easily visualize how to implement them as a package. After a two-hour discussion, we arrived at a list of some 30 super-requirements. We debated a variety of business strategies and finally converged on

	the concept of building a series of very small increments, while keeping customers informed of all our plans. Our goal was to develop a group of extremely loyal product users who would always know the date of the next release, the requirements the next release would satisfy, and so on. We invited select customers to a meeting at the corporate offices to explain our strategy and what we needed from them. It took a full day of discussion with the customers to arrive at a schedule of 10-week releases and their associated super-requirements. (The first release would take a bit longer because we had to build much of the supporting infrastructure.) We analyzed the first five releases separately to ensure that they were feasible. The customers were excited. Marketing was thrilled. Product development was confident that they could deliver each release on time. The first release of the product was created without a hitch, within budget, and with no surprises, and customers were satisfied with its functions. Unfortunately, soon after the first release, new customers emerged, market needs changed, and cooperation started to ebb within the company. As a result of the earlier meetings, product development was making many technical decisions that were based on the agreed-to releases and associated super-requirements. When needs changed, the esprit de corps began to erode. Rather than rethink prior decisions, they viewed changing needs as a threat to the success of the entire project. a) “This company was unusual”. Explain. b) What are super requirements? What are the various guidelines for making the super Requirements? c) How the new situation that emerged after the changing market scenario should be handled.
6.	In the current scenario of increasing market forces, changing requirements, conflicting personalities, inadequate comparison & analysis tools what are the various recommendations that practitioners should follow in software engineering & project management
7.	a)What are software metrics? What is software metric etiquette that should be followed by managers? b) Describe the change control procedure in the software configuration management