AUTUMN MID SEMESTER EXAMINATION-2018

School of Computer Engineering

KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
DEEMED TO BE UNIVERSITY, BHUBANESWAR-24

Software Engineering
[IT-3003]

Time: 1½ Hours **Full Mark: 20**

Answer any four 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.

Q.1. [5×1]

(a) What are functional and non-functional requirements?

1.a) Solution	
Marks distribution	Definition : 0.5 Mark Example : 0.5 Mark
Solution	Please refer book

(b) Why spiral model is known as meta model?

1.b) Solution	
Marks distribution	Proper explanation : 1 Mark
Solution	Subsumes all discussed models: 1) a single loop spiral retains the step-wise approach of the waterfall model. 2)uses an evolutionary approach -- iterations through the spiral are evolutionary levels. 3) enables understanding and reacting to risks during each iteration along the spiral, uses prototyping as a risk reduction mechanism

(c) Identify the different types of problems an analyst may come across during requirement analysis.

1.c) Solution	
Marks distribution	Explaining any 2 out of below three : 1 Mark
Solution	Once requirement gathering part is completed, the next activity known as requirement analysis can be started. During this process, the analysts will analyze the requirements as collected from the different possible users. The analysts try to resolve the following problems during requirement analysis: 1.**Inconsistencies**: Sometimes the requirements obtained from different users may contradict each other. This must be resolved by the analyst by talking to the respective users. They try to come to a common requirement that may be acceptable to the concerned users. 2.**Incompleteness**: Sometimes the requirements given by the users may not be complete. Some portion of the requirements may not be said by the users, due to their clack of clarity. Here, the analyst again tries to resolve the issue by talking to the concerned user. 3**.Ambiguity:** When the requirement is not written clearly then it may lead to multiple interpretation. This is called ambiguity and the requirements are known as ambiguous requirements. The analyst has to resolve this also.

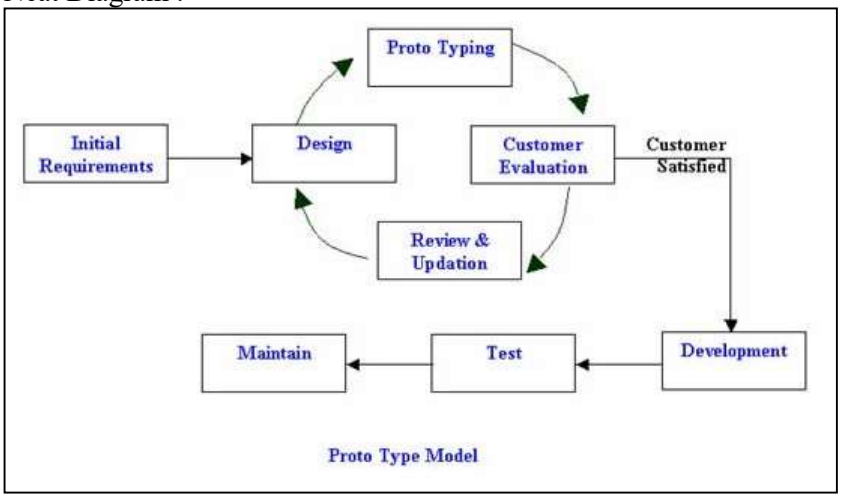
(d) What are the reasons for software crisis? Explain.

1.d) Solution	
Marks distribution	Any 3 to 4 out of below points : 1 Mark
Solution	1. Reasons For Software Crises: 2. Increase in size of software. 3. Increase in cost of developing a software. 4. Increased complexity of the problem area. 5. Project management problem. 6. Lack of understanding of the problem and its environment. 7. Duplication of efforts due to absence of automation in most of the software development activities. 8. Increasing skill shortage 9. Lack of adequate training in software engineering 10. Lack of communication between software developers and users.

(e) What are the roles of a software project manager ?

1.e) Solution	
Marks distribution	Mentioning 4 imp points : 1 Mark
Solution	The role of the project manager encompasses many activities including: • Planning and Defining Scope • Activity Planning and Sequencing • Resource Planning • Developing Schedules • Time Estimating • Cost Estimating • Developing a Budget • Creating Charts and Schedules • Risk Analysis,Managing Risks and Issues • Monitoring and Reporting Progress • Team Leadership • Working with Vendors • Scalability, Interoperability and Portability Analysis • Controlling Quality

Q.2. (a) Suppose you are the project manager in a software organization. During requirement gathering you found that the end users are not clear about their requirements and even after several iterations of discussion the specification document is not finalized. This problem will motivate you to follow which software life cycle models? Justify and discuss about the life cycle model.

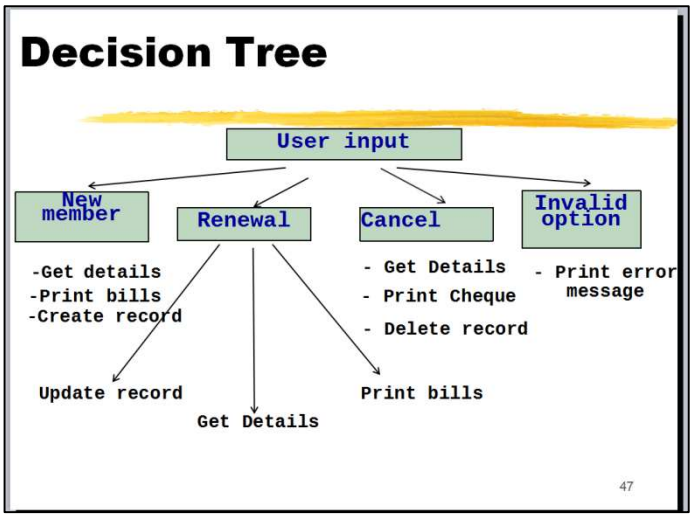
2. a) Solution	
Marks distribution	Correct type identification with justification : 1 Mark Discussion of the life cycle : 0.75 Mark Neat Diagram : 0.75 Mark
Solution	Correct type identification with justification : Prototype model Discussion of the life cycle : Please Refer Book Neat Diagram : 

[2.5]

(b) Write down the functional requirements for an automated library management system which supports the functions like add new member, renew membership and cancel membership. Please also create the suitable decision tree.

[2.5]

2, b) Solution	
Marks distribution	Writing functional Requirements : 2 Decision Tree : 0.5 Mark
Solution	**R-1 Add a new member** **Description:** To add a new member, the details of the guest like name, age, gender, address, contact number, email id etc. Are entered. This is stored in a database and a unique membership number is generated for the guest. **R1.1 Select "Add New Member" option** Input: "Add new member" option Output: User prompted to enter their details **R1.2 Membership generation** Input: Guest details as specified above. Output: Unique Membership Number **R-2 Renew Membership** Description: A registered user can renew their annual membership by entering the membership number and password. The system will validate the membership and display the membership expiry date. The user can renew the membership by selecting the subscription period and making the payment. The system will display the updated expiry date and print the bill. **R2.1 Select "Login" option** Input: Membership number and password Output: Display the member home page.

	Processing: Password validation, search the membership details from the database and display, if the password is valid. In case of invalid password, the member is asked to enter the details again. **R2.2 Select "Renew membership" option** Input: Member's choice to renew membership Output: Confirmation of the membership renewal and printing of bill. Processing: Payment validation **<u>R-3 Cancel Membership</u>** Description: Here, the guest is asked to enter their membership details. The system will display the member home page. Member opts for membership cancellation. The system will confirm the cancellation and print the cheque, if any. **R3.1 Select "Login" option** Input: Membership number and password Output: Display the member home page. Processing: Password validation, search the membership details from the database and display, if the password is valid. In case of invalid password, the member is asked to enter the details again. **R3.2 Select "Cancel Membership" option** Input: Member's choice to cancel membership. Output: Confirmation of membership cancellation and printing of cheque. 

Q.3. (a) Explain the concept of function point. How it is different from LOC? [2.5]

3. a) Solution	
Marks distribution	Concept of function point : 1.25 Mark Difference from LOC : 1.25 Mark
Solution	Concept of function point: A Function Point (FP) is a unit of measurement to express the amount of business functionality, an information system (as a product) provides to a user. FPs measure software size from the problem specification. They are widely accepted as an industry standard for functional sizing.

	The approach is to identify and count a number of unique function types: • external inputs (e.g. file names) • external outputs (e.g. reports, messages) • queries (interactive inputs needing a response) • external files or interfaces (files shared with other software systems) • internal files (invisible outside the system) FP=4 * #inputs + 5 * #Outputs + 4 * #inquiries + 10 * #files + 10 * #interfaces Difference from LOC : 1) FP can be computed easily from the problem specification itself but in case of LOC the size can be determined only after the code has been fully written. 2) Lines of code is a software used to measure the amount of code in a software program. It estimates the amount of exertion that is needed to develop a program. But coding is only one part of the entire development cycle hence this is a major drawback of LOC. Function point analysis is used to measure software size by computing its performance supplied to the use. It does so by breaking down the system into smaller elements so they can be better evaluated and are easier to understand.

(b) Consider a project with the following functional units: [2.5]
30 simple user forms and 20 average complexity user inputs.
40 update operations with screen display.
20 simple enquires and 15 complex enquiries)
User files = Customer Information, daily transaction details and 4 more.Number of external interfaces = 4
Assuming all complexity adjustment factors as average. Calculate the function points for the project.

Solution	
Marks distribution	Step1 : 1 Mark Step2: 1 Mark Step3: 0.5 Mark
Solution	**Step-1**: UFP = (30*3 + 20*4) + 40*5 + (20*3 + 15*6) + 6*10 + (4*7) = 170 + 200 + 150 + 60 + 28 = 608 **Step-2:** Considering the complexity adjustment factors of average complexity, TCF can be computed as below: TCF = 0.65 + (0.01*DI); where, 0<= DI<= 84 = 0.65 + (0.01*56); As the CAF is of average complexity. = 1.21 **Step-3**: Finally, the adjusted function point, FP = 608 * 1.21 = 735.68

	*Please note : With average complexity factor , DI can be considered as (3*14=42) or (4*14=56) With 42,* *TCF = 1.07* *FP=608*1.07 = 650.56*

Q.4. (a) Differentiate between basic COCOMO and intermediate COCOMO. [2.5]

4. a) Solution	
Marks distribution	Explaining basic and Intermediate COCOMO : 1 Mark Difference : 1.5 Mark
Solution	Explaining basic and Intermediate COCOMO : Please refer Book Difference : Though both, basic COCOMO and intermediate COCOMO estimates the effort and duration for the software project, they are different from each other. The basic COCOMO makes an assumption that effort and development time solely depends on the problem size. Thus it calculates the approximate estimation. On the other hand, intermediate COCOMO says that the effort and development time not only depend on the problem size but also on several other factors which are known as cost drivers. It also considers the 15 cost drivers, divided across four categories named as Product attributes, personal attribute, computer attribute and development environment; while calculating the effort and duration for the software project. Formula is to be mentioned. .

(b) Assume that the size of an organic type software product has been estimated to be 90,000lines of source code. Assume that the average salary of software engineers be Rs. 20,000/- per month. Determine the effort required to develop the software product, the nominal development time and productivity using COCOMO. (the constants are a or a1=2.4, b or a2=1.05, c or b1=2.5, d or b2=0.38)

[2.5]

4b)Solution	
Marks distribution	Effort : 1 Mark Duration: 1 Mark Productivity: 0.5 Mark
Solution	Given Product Type: Organic Source Code length: 90 KLOC Avg. Salary: Rs. 20000/- Per month For organics type software product, basic COCOMO says: **Effort, E** = 2.4*(KLOC)^1.05 PM = 2.4*(90)^1.05 = 270.499 PM **Duration, TDev** = 2.5*(Effort)^0.38 Months = 2.5*(270.499)^0.38 = 20.99 Months **Productivity**= KLOC/Effort = 90/270.499 = 0.33 KLOC/PM

Q.5. (a) Write short notes (any two) [2.5 +2.5]

I. Agile development process

5. I) Solution	
Marks distribution	Concept of Agile process : 1 Mark Diagram + Mention of Scrum/Extreme program : 1 Advantage : 0.5 Mark
Solution	Please refer Book .

II. Critical path and slack time.

5 II) Solution	
Marks distribution	Definition : 1.5 Mark Example : 1 mark
Solution	Please refer Book .

III. Software Project Management Plan (SPMP) document.

5 III) Solution	
Marks distribution	All important sections to be mentioned : 2.5 Marks
Solution	Please refer Book .

IV. Criteria of a good SRS document.

5 IV) Solution	
Marks distribution	Criteria : 2 Mark Example : 0.5 mark
Solution	Please refer Book .

------xxxxxxxxx-------