


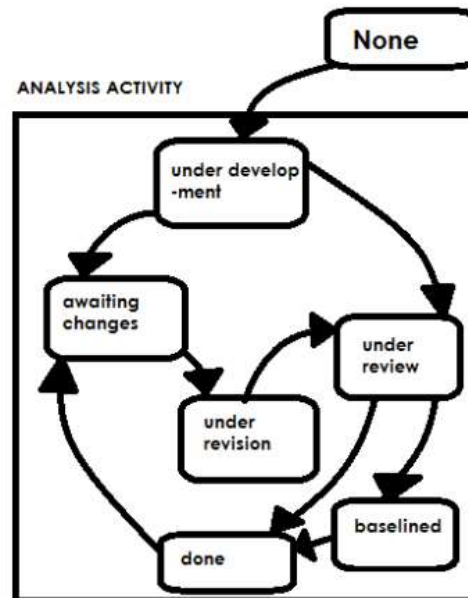
|  |   |
|--|---|
|  <p><b>SASTRA</b><br/>ENGINEERING · MANAGEMENT · LAW · SCIENCES · HUMANITIES · EDUCATION<br/>DEEMED TO BE UNIVERSITY<br/>(UPE 3 of the UGC Act, 1956)<br/>THINK MERIT   THINK TRANSPARENCY   THINK SASTRA</p> | <p><b>School of Computing</b><br/><b>Third CIA Examination – Apr '24</b><br/>Course Code: CSE215<br/>Course Name: Software Engineering<br/>Duration: 90 minutes Max Marks: 50</p> |
|--|---|

### PART A

#### Answer all the questions

10 x 2 marks = 20 marks

- Expand the following quality factors of the software: FURPS.  
FURPS— **functionality, usability, reliability, performance, and supportability**
- Derive all concurrent modeling phases.



Concurrent Process Model

- What are the steps of requirement engineering process?
  - Feasibility Study (Inception)
  - Elicitation of Requirements and Analysis (Elicitation)

- Specification of Software Requirements (Elaboration)
- Validation of Software Requirements (Negotiation)
- Management of Software Requirements (Specification and Requirements Management)

- How many levels are maintained in the CMMI of the software?

#### **CMMI Model – Maturity Levels :**

In CMMI with staged representation, there are five maturity levels described as follows :

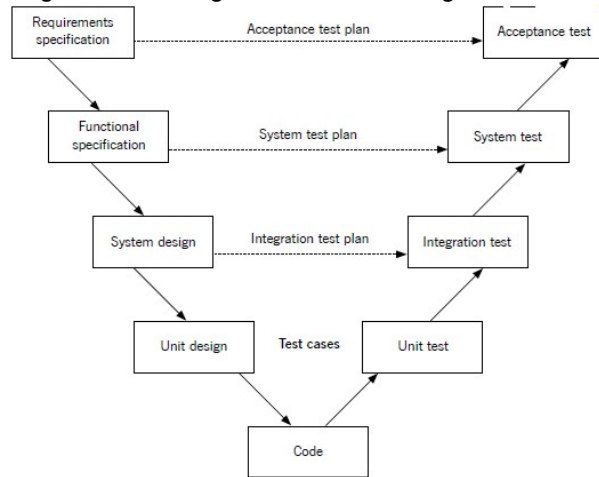
- Maturity level 1 : Initial**
  - Maturity level 2 : Managed**
  - Maturity level 3 : Defined**
  - Maturity level 4 : Quantitatively managed**
  - Maturity level 5 : Optimizing**
- Identify any two metrics of the software estimation from 4Ps.  
4 P's are:  
**Project:** LOC, process  
**People:** PM  
**Process:** Time, milestones, bugs  
**Product:** license keys, number of users, number of servers
  - State Beizer's four Behavior modeling testing methods. (**transaction flow, data flow, state transition, timing model**)
  - Design any four domain classes of "University Management system"  
Any four of (**student, course, dept, office, faculty, transport, examWing, foodStall, playground, library**)
  - What are the two phases of documentation testing? **1. Technical review, 2. Live testing**
  - Enlist all approaches of Integration testing. (**top down, bottom up, sandwich**)
  - What are the guidelines of defining equivalence partitioning classes?
    - If an input condition specifies a range, one valid and two invalid equivalence classes are defined.
    - If an input condition requires a specific value, one valid and two invalid equivalence classes are defined.
    - If an input condition specifies a member of a set, one valid and one invalid equivalence class are defined.
    - If an input condition is Boolean, one valid and one invalid class are defined.

## PART B

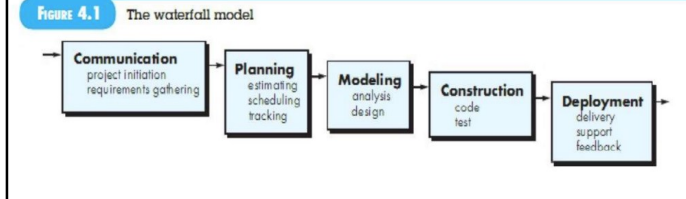
### Answer any two Questions

2 x 10 marks = 20 marks

11. Differentiate V-Model and Water fall model by its process flow diagram, advantages and disadvantages.



### Linear: Water Fall Model



12. Assume there is an another version of Open Office package which consist of additional package as "insert a new object with AR modeling on camera captured images". Identify major, inter tasks and attributes of given OO software for an OOTesting.

Insert-> new->object

AR->Model->add->picture

Camera->capture->image

Attributes: any 2 attributes of object(attr1,attr2)/

image(pixel,dimension,x,y, RGB...etc)

13. Construct L9 Orthogonal array for the given software problem:

A microprocessor's functionality has to be tested to the given factor:

Temperature: 100C, 150C and 200C.

Pressure : 2 psi,5psi and 8psi

Doping Amount :4%,6% and 8%

Deposition Rate : 0.1mg/s , 0.2 mg/s and 0.3mg/s at these three levels.

3 levels 4 factors, i.e  $(3)^4 = 81$  possible test cases, but L9 is

| Test case # | Temperature | Pressure | Doping amount | Deposition rate |
|-------------|-------------|----------|---------------|-----------------|
| 1           | 100C        | 2 psi    | 4%            | 0.1 mg/s        |
| 2           | 100C        | 5 psi    | 6%            | 0.2 mg/s        |
| 3           | 100C        | 8 psi    | 8%            | 0.3 mg/s        |
| 4           | 150C        | 2 psi    | 4%            | 0.1 mg/s        |
| 5           | 150C        | 5 psi    | 6%            | 0.2 mg/s        |
| 6           | 150C        | 8 psi    | 8%            | 0.3 mg/s        |
| 7           | 200C        | 2 psi    | 4%            | 0.1 mg/s        |
| 8           | 200C        | 5 psi    | 6%            | 0.2 mg/s        |
| 9           | 200C        | 8 psi    | 8%            | 0.3 mg/s        |

## PART C

### Answer to all Questions

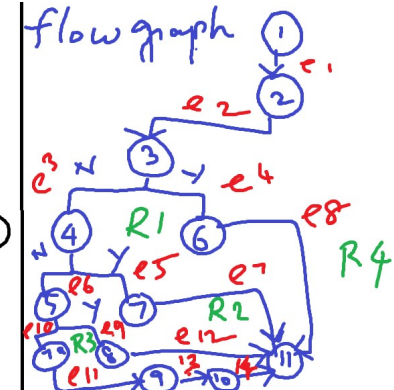
1 x 10 marks = 10 marks

- 14.a Draw flow graph, graph matrix of the following instruction set and find cyclomatic complexity. (2 marks)

$V(G) = E - N + 2 \Rightarrow 14 - 12 + 2 = 4$ ,  $P = 3 + 1$ ,  $R = 4$

```

Start
declare
float a=1.2,b=2.5, c=2.0;
Get a,b,c;
d=sqrt(b*b-4*a*c);
r1=d/2*a;
r2=-d/2*a;
if (d==0)
  print("roots are equal");
else if (d>0)
  print(" roots are real");
else if (d<0)
  print("roots are complex");
else print ("invalid inputs");
endif
print(r1,r2);
end
  
```



14.b. Perform selective path testing, control structure and loop testing of matrix operations such as 1. Matrix addition 2. Matrix subtraction 3. Matrix multiplication with necessary constraints and verification points. (3 marks)

Control structure testing is used to increase the coverage area by testing various control structures present in the program. The different types of testing performed under control structure testing are as follows-

1. Condition Testing ( $M=N$  or not)
2. Data Flow Testing (A matrix row == B matrix col)
3. Loop Testing (simple,concatenate,nested,1...N times)

**Simple Loop** – The following set of test can be applied to simple loops, where the maximum allowable number through the loop is n.

1. Skip the entire loop.
2. Traverse the loop only once.
3. Traverse the loop two times.
4. Make p passes through the loop where  $p < n$ .
5. Traverse the loop n-1, n, n+1 times.

**Concatenated Loops** – If loops are not dependent on each other, contact loops can be tested using the approach used in simple loops. if the loops are interdependent, the steps are followed in nested loops.

**Nested Loops** – Loops within loops are called as nested loops. when testing nested loops, the number of tested increases as level nesting increases. The following steps for testing nested loops are as follows-

1. Start with inner loop. set all other loops to minimum values.
2. Conduct simple loop testing on inner loop.
3. Work outwards.
4. Continue until all loops tested.

**Unstructured loops** – This type of loops should be redesigned, whenever possible, to reflect the use of unstructured the structured programming constructs.

14.c. How to conduct Model based testing (MBT) for the “online medicine ordering and delivery system” with atleast 3 UML models?

Write the steps and guidelines of MBT.(3 marks)

- Analyse an existing behavioral model for the software or create one.
- Traverse the behavioral model and specify the inputs that will force the software to make the transition from state to state
- Review the behavioral model and note the expected outputs as the software makes the transition from state to state
- Execute the test cases
- Compare actual and expected results and take corrective action as required

Recall that external events or stimuli

- 1. evaluate all use cases to fully understand the sequence of interaction within the system
- 2. Identify events that drive the interaction and understand how these events relate to specific objects
- 3.create a sequence for each use case
- 4. build UML state diagram for the system
- 5. Review behavioral model to verify accuracy and consistency

Any 3 UML diagrams(3 marks)

14.d. Differentiate real time system testing with stand alone conventional module based software testing. (2 marks)

- Timing of the data
- Parallelism of the tasks (process)
- Same data with different system may lead error
- Relationship between hw device and sw , example: xerox copier, jamming, toner low, RESET, DARKEN state

Check INTERRUPT: priority, process time, high volume?

CONVENTIONAL: event, inter task, modules, unit, integration

\*\*\*\*\*