INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal - 500 043, Hyderabad, Telangana

Examinations Control Office

Examination	B TECH VI SEMESTER END EXAMINATIONS REGULAR JUNE 2025 REG UG20		
Month & Year	1-Jun	**Date**	23/06/2025
Course Name	SOFTWARE QUALITY ASSURANCE AND TESTING		
Course Code	ACIC02	**E-Code**	3451

Instructions to Evaluators

- Evaluators should spend at least 3-5 minutes on one answer booklet during the evaluation.
- Evaluators should cross check that marks are allotted for all the attempted questions.
- The marks should be assigned fairly according to the mark distribution specified in the scheme of evaluation.
- For questions that were attempted incorrectly, evaluators are required to award zero marks.
- The evaluator must give a proper justification in case of any mistakes identified in the marks provided.

START WRITING FROM HERE

Q.No.

1a) Ans: Software testing is a process where software of the system is tested. This testing ensures the proper working of the software.
Some of the primary objectives to be archived during software testing are:

1) Functionality:- The tested software should be functional. It should be feasable to implement.

2) Reliability: The software must be reliable. It should support the hardware/It should compliment the hardware.
It tolerate errors. It should have a way to deal with failures.
The software should be robust and unhinged even after multiple uses.

3) Useability:- The software that we are testing at the end of the ~~day~~ day

Q.No.

is used by consumer. So while developing the software we should consider the customer/clients opinion. The software must be user friendly. The UI and design should be simple.
Try avoiding the use of dark and contrast colors. Use bright/Light colors.

4) Efficiency:
The software must run efficiently. The efficiency is the key for better customer satisfaction.
The time and work done should be optimal.
Software is said to be efficient when the software takes less time and completes more work without error in that less time.

5) Maintainability:
While designing a software one must keep in mind

Q.No.

- that the software should be easy to maintain. If the maintaince is less than more number of users start using your software.
Less maintaince means more customer satisfaction.

6) Probability:

The probability of the software to mess up should be less.

Testing contribute a lot to software reliability and customer satisfaction.
By testing we try to replicate the real world situations in a closed and controlled environment.
By replicating these situations we try to understand the possible output we get from it.
If we don't get a desired output or encounter an error/bug we immediately try to fix and patch up that part of software by doing this the software becomes robust which make that software more reliable and make the customer satisfy more.

Q.No.	
2a) Ans:	Function testing is a type of Software testing where we test the different-different parts of the code to make sure that function of the code works fine and efficiently without encountering any errors.

There are different types of functional testing. They are:

1) Incremental
2) Top-down
3) Bottom-up
4) Sandwitch
5) Big-bang.

1) Incremental:

This is a type of testing where it test all the units by combining them. It checks whether all the units are working together without contradicting each other or not.

Q.No.

2) Top-down:

Just like in trees software testing also has a top-down functional testing. Here all the code from top by combining one by one is tested. Here in this process we use studs as a placeholder for an empty variable to not to encounter any errors while testing.

3) Bottom-up:

This technique is a bottom to up technique. Here the testing starts from bottom and goes all the the way up to the starting of the code. Here in this process we use drives as a placholder for empty variables while testing. We use it as it ensures no errors while testing.

4) Sandwich:

This technique is a combination of both Top-down and Bottom-up techniques. Here we increment values one by one while testing.

Q.No.

This process uses both studs and drives for placeholders as it uses both techniques.

5) Big-bang integration:

Here in this technique we integrate all the code one by one and test it all at once. It would be a bit ~~defi~~ tough / time consuming to find error using this technique. But we can exactly pinpoint the error using this technique.

Q.No.

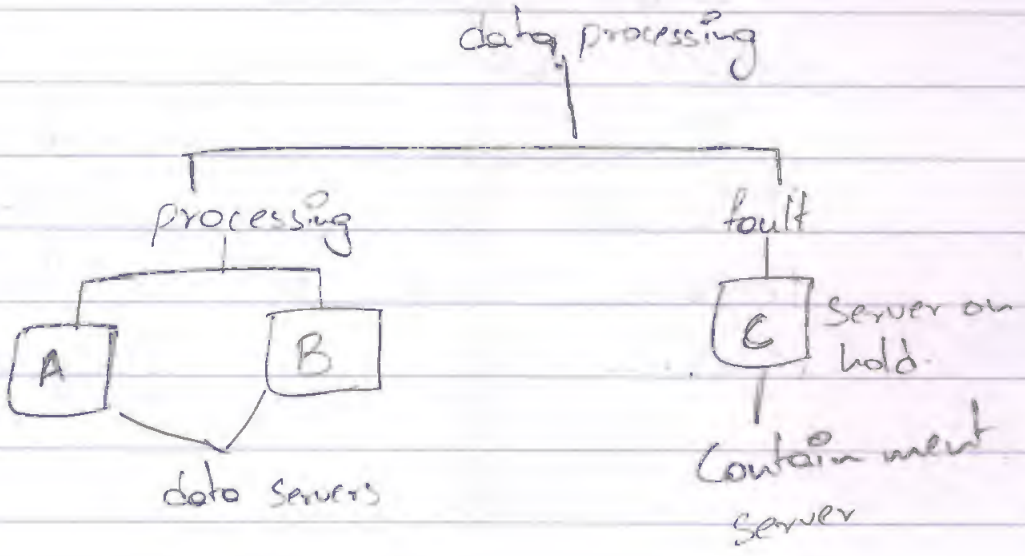
7 b) Ans-

i) Software fault tolerance:

Software fault tolerance is a mechanism where we operate the software even after encountering an error.

Basically it is a mechanism which ensures the proper working of software even after encountering an error.

So when software is under stress it checks the softwares ability to give correct outputs even after encounter an error or fault in the given data.



ii) Safety assurance:

Safety assurance is a mechanism to protect your data / save your data.

Q.No.

When as software encounter a bug/error/fault in the data software safety assurance is a mechanism that ensures that the safety of the data is ensured. It protects the consumers data from being stolean or lost or miss used.

iii) Failure Containments-

Failure Containment is a mechanism where the faults/errors of the software are contained. Through this process or it becomes easy for us to detect error.

Q.No.

7a) Ans:-

Root cause Analysis:

Root cause analysis or RCA is a powerful mechanism which is used to prevent defects from occuring in the software.

Root cause analysis is a mechanism where a tester check each and every line of code with pressition and control.
The tester checks for errors which happend and happening in the system. Now the tester understands the root cause of the system by analyzing each and every line of code of that software.
Root cause Analysis follows a certain steps:

1) Check:-

The first step is to check for abnormalities /error in the system.

2) Detect errors:-

The next step is to detect the error that is in the software.

Q.No.

3) Analyze the errors

Analyze the error. Study the error in the system.

4) Understands

Understand why the error occurs. What would be the cause of that error.

5) Repairs

As now you understood the error now repair the software in such a way that error would never happen again. and also fix any anomilities / abnormal code from the system should be removed.

As we scan the entire code in root couse analysis we also fix the error or threats which may occur in future by doing this the chances of encountering a defect would be close to zero.

Q.No.

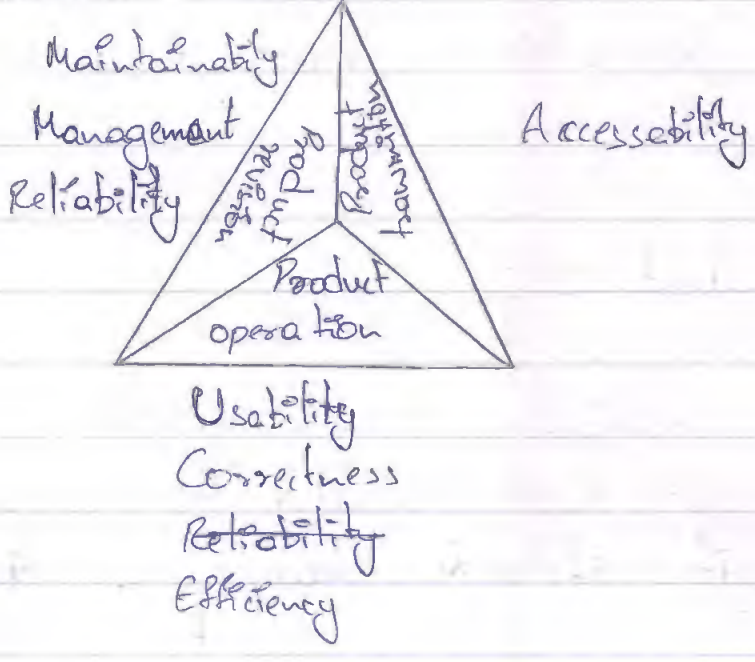
5a) Ans

McCall's Model:

McCall's Model or McCall's quality factors is one of the oldest model. We use these quality factors to test whether a software is good or not. There are mainly 3 specific factors to determine the quality of the software.

They are:-

1) Product operation
2) Product revision
3) Product transmition



1) Product operations

This part of the model is one of the most important part as the customer sees this part of the product.

Q.No.

The key takeaways in this part of the module is how the product works.
The working of the product must be easy to use.
It should be easy to understand.
The operations must an always give a correct output.
There should not be any wrong output for give input.
The product must be efficient in handeling the operation.
The operations must be user friendly.

2) Product revisions.

This part of the model tests the software to be robust.
The product should be reliable as it should have a good fault tolerance, safety assurance and should contain the faults.
This makes the product more reliable.
Not only reliable but the product

Q.No.

should also be easy to maintain it should not become a hassetle for the customer to maintain the product you sold.
the product should work fine even with little to no maintance.
Not only maintining the product it should also be easy to manage It should manage heavy tasks without errors just fine.

3) Product transmittions

This part of the product is also important as it helps in accessing your product.
Your product should be easily available. Easy to access even for new user.

Q.No. 1

(b) (b)

Ans

Verification:

Verification is a process in software testing where we check and verify the input and output are correct or not.

When a software is verified it does not mean that the software is valid. Verifying is just cross checking the process which does not mean that it is correct or it is a void process.

Validation:

Validation is a process in software testing where we check and also correct if the process is correct or not.

Software is also verified when the software is valid.

It not only cross checks but also corrects if needed.

Q.No.

Eg:-

data base A	data base B
2+2 = 5	2+2 = 5

Since both are same it is verified but the actual answer of 2+2 = 4 so the software is not valid.

Test automation in softer testing plays an an important role in todays testing.

As test automation requires atmost one team it is very cost effective for companies.

As test automation involves less human intervent ion the error would be greately minimized

As test automation is an automated process it saves a lot of time.

Q.No.

2b) Ans:

Boundary value Analysis (BVA):

As the name suggest it not only checks the boundary values of the code but also checks the compatability of the software with other systems.
This helps us in choosing which type of hardware is more suitable for the software.
This kind of analysis helps in choosing the write version of the software for hardware.
We can also make/check if one software is compatable with another.

Decision tables:

These are tables used in testing of software.
They give a more detailed and breify understanding of the software.
We can also check the compatability using these tables.

Q.No.

Eg:

Products	Version	Compatibility	Pass.
product x	Bluetooth V5.1	yes	✓
product y	Bluetooth V.6.1	No	X
product z	Bluetooth V.5.3	yes	✓

It ~~basic~~ basically simplifies the data to make quick and correct decisions.

The factors that influence software reliability quality are:

1) Cost Fault tolerance
2) ~~Prime~~ Saftey assurance
3) Fault containment.

1) Fault tolerance:

This is the important factor which does produce correct output despite encountering errors in data. This ensures correct output or smooth running even while encounting errors and bugs.

Q.No.

2) Safety assurances

This assures the safety of data and protection of important files so that outsiders can not steal your data.

3) Fault containments

This ensures that even if there is a fault the whole program is not destroyed. The fault is contained in a part of program.

Q.No.

5b) Ans

Software quality assurance (SQA) function is responsible for providing a quality software in agile and DevOps environments.

As software quality assurance ensures following things.

1) Functionality of the program is maintained across environments.

2) Correctness and correct output for the given input is maintained.

3) Reliability is at top priority.

4) Usability is ensured as it ensures user friendly interfaces and easy to use mechanisms.

5) Efficiency is maintained as the DevOps exceeds if it.

Q.No.

6) GUI testing is done to ensure all the UI components are working.

7) Data is compressed to archive more efficiency

Q.No.

4a)
Ans

Defect casual analysis is a type of software analysis which make sures that the software would not contain any kind of defects while running the program.

This process can be performed using RCA(Root cause analysis) as it is use to prevent defects from occuring.

Defect casual analysis is a step by step process with steps

1) Scanning the code for any abnormalities.

2) Detecting a defect.

3) Study and understand the defect. Know why the defect is occuring. Study the root cause for the defect.

4) Now try to repair the defect. First try to do this in the close enviroment. So if any error would orrise we can

Q.No.

easily fix or the main code would not be affected.

5) Now release a patch that would fix the original code.

Regression testing plays a very important role during system ~~maintain~~ maintenance

Q.No.

4b) Ans:

We do need integration testing to test our software.
As this type of testing helps us to combine and test individual units and their bond/relation between units.
There are various types of integration testing.
They are:
1) Top-Down
2) Bottom-Up
3) Sandwitch
4) Big-Bang

1) Top-Down:

The top down technique is an approach where we start from top and slowly integrate each unit step by step and test it.
Here we use studs as a placeholder for empty spots.

2) Bottom-Up:

The Bottom up technique is an approach where we start from bottom and slowly step by step integrate units and

Q.No.

test them.
Here we use drivers as placeholders for empty spots in the code.

3) Sandwich:-
This process uses both top-down and bottom-up methods to efficiently integrate the code.
Here we use both studs and drivers as empty spots as we use both top-down and bottom up approaches.

4) Big-Bang:-
This is a process where entire code is integrate all at once.
The entire code or big chunks (big parts) of code is tested using integration testing.

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

Q.No.	

ROUGH WORK

Content written here will not be considered for valuation