CSE6324 - Software Verification & Validation

Tutorial 3

1. Why is it impossible to test a program completely?

   ANS: A program cannot be exhaustively tested because real world software has a lot of codes, and it is not possible to test every code. Sometimes the system cannot test completely because of time and budget limitations. Also, sometimes it has to interact with other systems, so it is not possible to test completely.

2. If you were testing a simulation game such as a flight simulator, what do you think would be more important to test – its accuracy or its precision?

   ANS: Both. Because accuracy refers to how well the simulator's behavior matches real-world flight physics. This is required for realistic training and simulations. On the other hand, Precision refers to the amount of precision and granularity in the flying model. For example, a high-precision simulator may use fine-grained controls to correctly replicate the behavior of individual aircraft systems.

3. Is it possible to have a high-quality but a low-reliability product? Provide an example of such product.

   ANS: 3D-Printed Cutting-Edge Technology Products.
   Made using modern 3D printing materials, providing amazing design features and customization choices. The 3D printing process itself may create possible flaws, especially in the early phases of the technology for a certain material or design. Issues with layer adhesion or intrinsic material qualities may lead to: Breakage under stress: The product may not endure normal wear and tear or accidental drops.
   Functional limitations: Certain functionalities may not function properly due to printing errors.

4. Provide reason(s) why testers are not popular among the project team members and suggest some means for the testers to stay in good terms with their fellow teammates.

   ANS: testers are not popular among the project team members because they may become famous among team members often, they find faults and errors on work. Some team members might feel that testing delays the project timeline.

some means for the testers to stay on good terms with their fellow teammates are clearly telling issues and suggestions in a conservative way. Participate in team discussions and decision-making process.

5. Explain what is meant by Pesticide Paradox as used in the context of testing.

ANS: The pesticide paradox shows how constantly employing the same test cases reduces effectiveness over time. Imagine spraying the same insecticide on insects; they will ultimately develop resistance. Similarly, software might become "resistant" to tests that only reveal already known problems.
To resolve the pesticide paradox:
Evolve Test Strategies: Review and update test cases on a regular basis to reflect new functionality and potential concerns.
Investigate Different Testing Techniques: To detect unknown problems, use a range of testing methodologies (for example, boundary value analysis and equivalence class splitting).
Automate repetitive tests. Automate regression testing allows testers to take more exploratory and creative methods.

6. Define the following terms:
(as in IEEE Standards: IEEE Standard for Software Test Documentation, IEEE Standards for Verification & Validation)

Minimum Tasks
Life Cycle Processes
Test Effort

Here are the definitions based on IEEE standards:

- **Minimum Tasks:** The essential testing activities required to achieve a specific testing objective within a life cycle process (LCP). These tasks represent the minimum coverage needed, but they can be expanded for a more comprehensive test strategy.
- **Life Cycle Processes (LCP):** Structured processes within the software development life cycle (SDLC) that involve planning, creating, executing, and documenting software tests. Examples include requirements review, design review, unit testing, integration testing, and system testing.
- **Test Effort:** The total amount of resources (time, personnel, budget) allocated to software testing activities throughout the LCP. It's essential to estimate test effort accurately for project planning and resource allocation.