SNS COLLEGE OF TECHNOLOGY

Kurumbapalayam (Po), Coimbatore – 641 035

An Autonomous Institution

Accredited by NAAC – UGC with 'A++' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE NAME: 23TSO101 SOFTWARE TESTING (OPEN ELECTIVE)

III YEAR / VI SEMESTER

Unit 1 - INTRODUCTION

Topic 4 : Testing axioms and Basic definitions

FOUNDATION

What Are Axioms in Testing?

Axioms are **fundamental, self-evident truths** or principles accepted without proof. In software testing, these axioms serve as the bedrock that guides how we understand and approach the entire testing process.

They form the foundation for building reliable, effective testing strategies that stand up to real-world challenges and constraints.
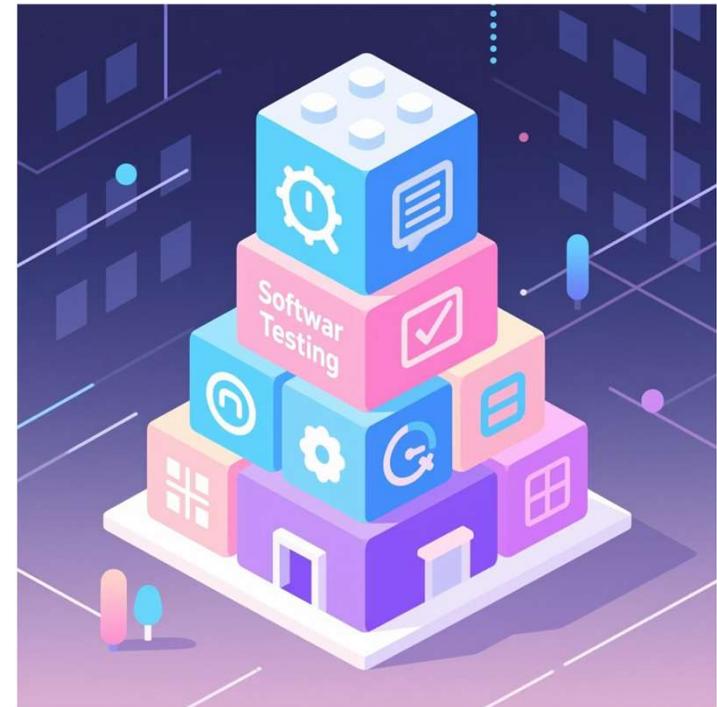
Why Testing Axioms Matter

Focus & Clarity

Help testers concentrate on what is realistically achievable within project constraints

Prevent Misconceptions

Counter common myths, such as the belief that testing can prove software is completely bug-free

Strategic Framework

Provide a structured approach to prioritize testing efforts and effectively manage project risks

Testing Shows the Presence of Defects

Testing can reveal bugs lurking in your code, but it **cannot guarantee their complete absence**. Even the most extensive testing campaigns cannot prove software is 100% defect-free.
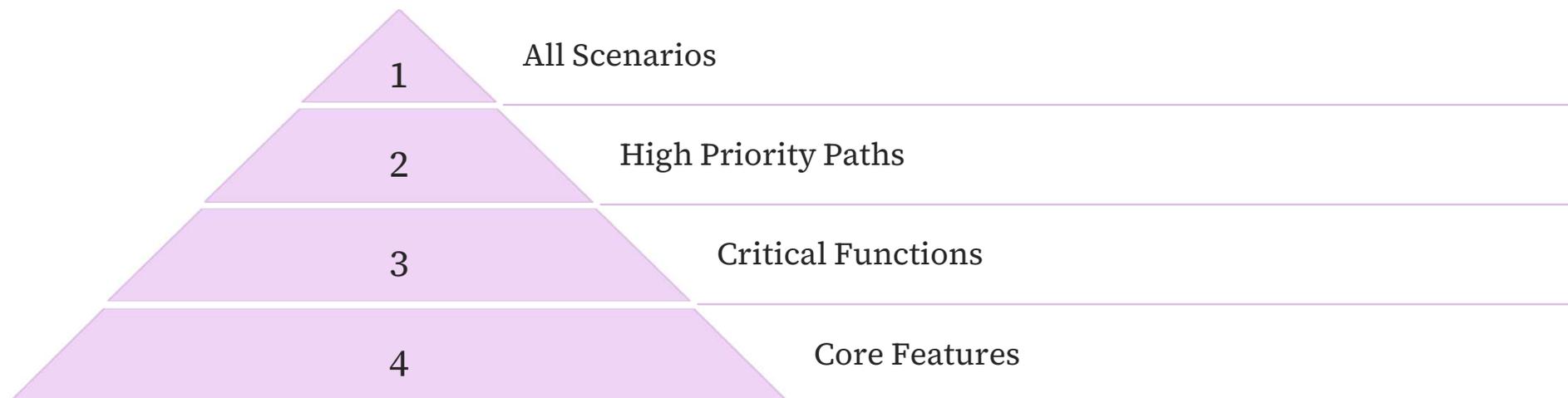
Real-world example: Multiple comprehensive test cycles may uncover dozens of bugs, yet some defects will inevitably remain hidden until discovered in production.

"Testing shows the presence, not the absence of bugs." — Edsger Dijkstra

1 *AXIOM 2*

Exhaustive Testing is Impossible

Level	Label
1	All Scenarios
2	High Priority Paths
3	Critical Functions
4	Core Features

Testing every possible input combination, execution path, and scenario is **impractical due to combinatorial explosion**. As features multiply, test cases grow exponentially—making complete coverage impossible within realistic time and budget constraints.

Smart testers must **prioritize strategically** based on risk assessment, potential business impact, and likelihood of failure in production environments.

Software Testing is Context-Dependent

Testing approaches vary dramatically depending on software type, industry domain, regulatory requirements, and user environment. What works for one project may be entirely inappropriate for

E-Commerce Platform	Medical Device Software
• High traffic load testing	• Rigorous regulatory compliance
• Payment gateway security	• Safety-critical validation
• User experience optimization	• Real-time performance testing
• Cross-browser compatibility	• Extensive documentation

Tailoring your testing strategy ensures maximum relevance, efficiency, and value for each unique project context.

Not All Bugs Found Will Be Fixed

In real-world software development, some defects are intentionally deferred or left unfixed due to cost considerations, acceptable risk levels, or minimal business impact.

Additionally, **fixing bugs can introduce new issues**—regression defects that may be worse than the original problem. Teams must carefully weigh trade-offs when deciding which bugs to address.

⚠

Critical Bugs

Fixed immediately—system crashes, data loss, security vulnerabilities

🕐

Minor Issues

Deferred to future releases—cosmetic UI glitches, rare edge cases

Activities

Activity 1:

Match the term to its meaning:

A	B
1. Verification	a. Software behaves wrongly when executed
2. Defect	b. "Are we building the product right?"
3. Failure	c. Flaw introduced in the software

Answer: 1-b, 2-c, 3-a

Basic Definitions: Key Testing Terms

Test Case

A specific set of conditions, inputs, and expected outcomes designed to verify whether a particular feature or function works correctly

Test Plan

Comprehensive document outlining testing scope, strategy, approach, required resources, timeline, and deliverables for a project

Defect

Any flaw, error, or deviation from specified requirements discovered during the testing process that impacts functionality

Test Execution

The process of running prepared test cases, observing actual results, and systematically recording outcomes for analysis

The Goal of Testing

Reduce Risk

Find critical defects early in the development lifecycle when they're cheaper and easier to fix

Meet Standards

Ensure software meets user requirements, business objectives, and quality standards

Balance Resources

Achieve thoroughness while respecting cost, time, and resource constraints

Embrace Testing Axioms for Success

Accept Reality

Perfect testing is impossible, but **effective, strategic testing is achievable** and delivers real value

Guide with Principles

Use these axioms to develop **smart, risk-based testing strategies** that maximize impact

Focus on Delivery

Prioritize delivering **reliable software that meets real-world user needs** and business goals

Manage Uncertainty

Remember: Testing is fundamentally about **managing uncertainty, not eliminating it** entirely

Mind Map

TESTING AXIOMS AND BASIC DEFINITIONS /23TSO101
SOFTWARE TESTING / Mr.K.Karthikeyan/CSE/SNSCT
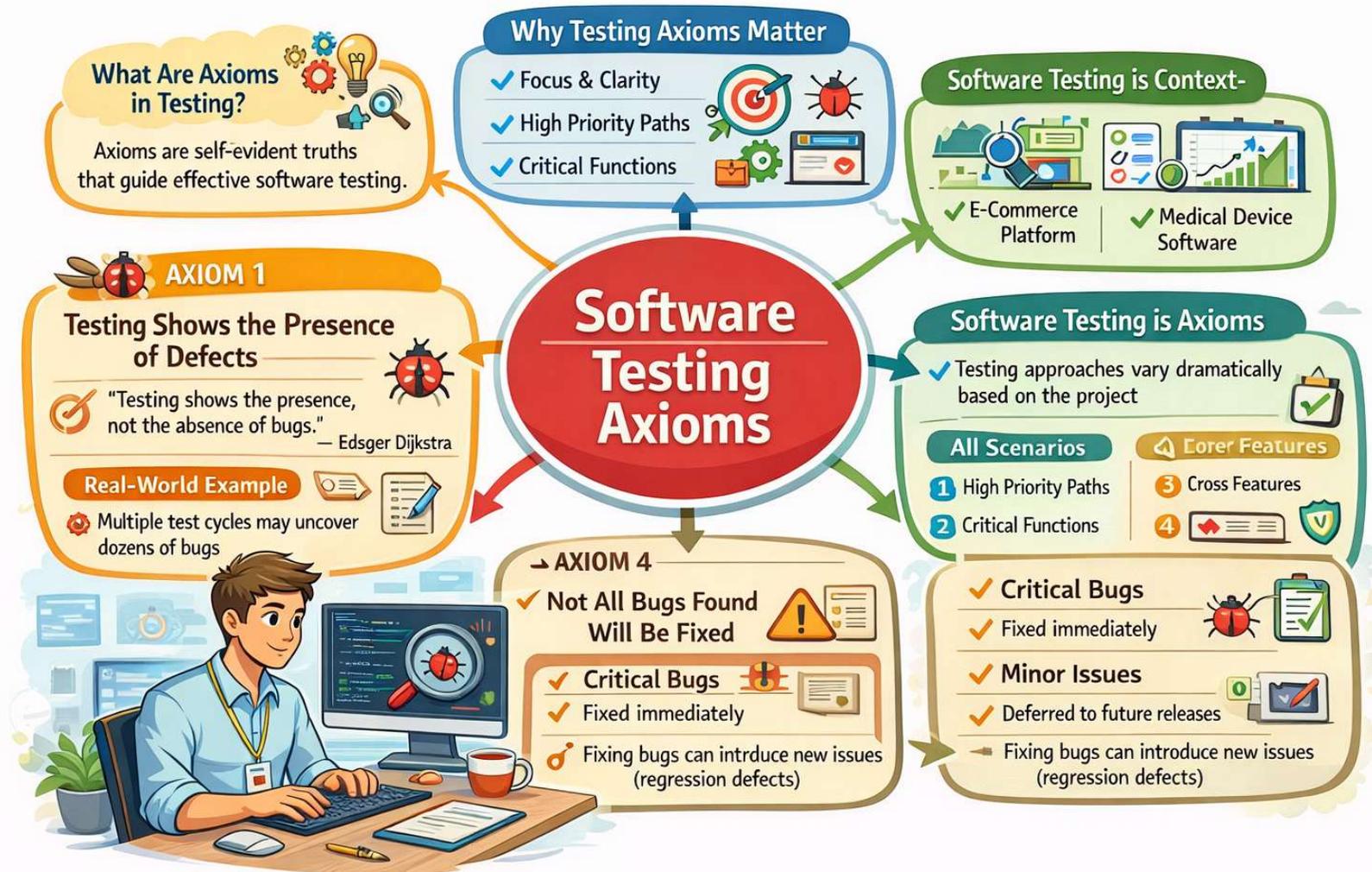
MCQs — Testing Axioms & Basic Definitions

1

Testing shows the _____ of defects, not their _____.

A. absence, presence

B. presence, absence

C. removal, existence

D. solution, problem

Answer: B

2

Which of the following is TRUE about exhaustive testing?

A. Always possible

B. Possible only for small systems

C. Eliminates all defects

D. Required for certification

Answer: B

TEXT BOOKS	
1.	Srinivasan Desikan and Gopalaswamy Ramesh, ―Software Testing – Principles and Practices, Pearson Education, 2006.
REFERENCES	
1.	Ilene Burnstein, ―Practical Software Testing, Springer International Edition, 2003
2.	Edward Kit, Software Testing in the Real World – Improving the Process‖, Pearson Education, 1995
3.	Boris Beizer, Software Testing Techniques – 2nd Edition, Van Nostrand Reinhold, New York, 1990.
4.	Aditya P. Mathur, ―Foundations of Software Testing _ Fundamental Algorithms and Techniques, Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008.

THANK YOU