

# Foundation Level Learning Objectives

Learning objectives are indicated for each section in the syllabus and classified as follows:

- K1: remember, recognize, recall
- K2: understand, explain, give reasons, compare, classify, categorize, give examples, summarize
- K3: apply, use
- K4: analyze

These are the Foundation Level Learning Objectives:

## 1. Learning Objectives for Fundamentals of Testing

### 1.1 Why is Testing Necessary? (K2)

- LO- Describe, with examples, the way in which a defect in software can  
1.1.1 cause harm to a person, to the environment or to a company (K2)
- LO- Distinguish between the root cause of a defect and its effects (K2)  
1.1.2
- LO- Give reasons why testing is necessary by giving examples (K2)  
1.1.3
- LO- Describe why testing is part of quality assurance and give examples of  
1.1.4 how testing contributes to higher quality (K2)
- LO- Explain and compare the terms error, defect, fault, failure and the  
1.1.5 corresponding terms mistake and bug, using examples (K2)

### 1.2 What is Testing? (K2)

- LO- Recall the common objectives of testing (K1)  
1.2.1
- LO- Provide examples for the objectives of testing in different phases of  
1.2.2 the software life cycle (K2)
- LO- Differentiate testing from debugging (K2)  
1.2.3

### 1.3 Seven Testing Principles (K2)

- LO-1.3.1 Explain the seven principles of testing (K2)

### 1.4 Fundamental Test Process

- LO- Recall the five fundamental test activities and respective tasks from  
1.4.1 planning to closure (K1)

### 1.5 The Psychology of Testing (K2)

- LO-1.5.1 Recall the psychological factors that influence the success of testing  
(K1)
- LO-1.5.2 Contrast the mindset of a tester and of a developer (K2)

## 2. Learning Objectives for Testing Throughout the Software Life Cycle

### 2.1 Software Development Models (K2)

- LO- Explain the relationship between development, test activities and work  
2.1.1 products in the development life cycle, by giving examples using  
project and product types (K2)
- LO- Recognize the fact that software development models must be adapted  
2.1.2 to the context of project and product characteristics (K1)
- LO- Recall characteristics of good testing that are applicable to any life  
2.1.3 cycle model (K1)

### 2.2 Test Levels (K2)

- LO- Compare the different levels of testing: major objectives, typical objects  
2.2.1 of testing, typical targets of testing (e.g., functional or structural) and  
related work products, people who test, types of defects and failures to  
be identified (K2)

### 2.3 Test Types (K2)

- LO- Compare four software test types (functional, non-functional, structural  
2.3.1 and change-related) by example (K2)
- LO- Recognize that functional and structural tests occur at any test level  
2.3.2 (K1)
- LO- Identify and describe non-functional test types based on non-  
2.3.3 functional requirements (K2)
- LO- Identify and describe test types based on the analysis of a software  
2.3.4 system's structure or architecture (K2)
- LO- Describe the purpose of confirmation testing and regression testing  
2.3.5 (K2)

### 2.4 Maintenance Testing (K2)

- LO- Compare maintenance testing (testing an existing system) to testing a  
2.4.1 new application with respect to test types, triggers for testing and  
amount of testing (K2)
- LO- Recognize indicators for maintenance testing (modification, migration  
2.4.2 and retirement) (K1)

- LO- Describe the role of regression testing and impact analysis in maintenance (K2)
- 2.4.3

### **3. Learning Objectives for Static Techniques**

#### **3.1 Static Techniques and the Test Process (K2)**

- LO- Recognize software work products that can be examined by the different static techniques (K1)
- 3.1.1
- LO- Describe the importance and value of considering static techniques for the assessment of software work products (K2)
- 3.1.2
- LO- Explain the difference between static and dynamic techniques, considering objectives, types of defects to be identified, and the role of these techniques within the software life cycle (K2)
- 3.1.3

#### **3.2 Review Process (K2)**

- LO- Recall the activities, roles and responsibilities of a typical formal review (K1)
- 3.2.1
- LO- Explain the differences between different types of reviews: informal review, technical review, walkthrough and inspection (K2)
- 3.2.2
- LO- Explain the factors for successful performance of reviews (K2)
- 3.2.3

#### **3.3 Static Analysis by Tools (K2)**

- LO- Recall typical defects and errors identified by static analysis and compare them to reviews and dynamic testing (K1)
- 3.3.1
- LO- Describe, using examples, the typical benefits of static analysis (K2)
- 3.3.2
- LO- List typical code and design defects that may be identified by static analysis tools (K1)
- 3.3.3

### **4. Learning Objectives for Test Design Techniques**

#### **4.1 The Test Development Process (K3)**

- LO- Differentiate between a test design specification, test case specification and test procedure specification (K2)
- 4.1.1
- LO- Compare the terms test condition, test case and test procedure (K2)
- 4.1.2
- LO- Evaluate the quality of test cases in terms of clear traceability to the requirements and expected results (K2)
- 4.1.3
- LO- Translate test cases into a well-structured test procedure specification at a level of detail relevant to the knowledge of the testers (K3)
- 4.1.4

#### **4.2 Categories of Test Design Techniques (K2)**

- LO- Recall reasons that both specification-based (black-box) and structure-based (white-box) test design techniques are useful and list the common techniques for each (K1)
- 4.2.1
- LO- Explain the characteristics, commonalities, and differences between specification-based testing, structure-based testing and experience-based testing (K2)
- 4.2.2

#### **4.3 Specification-based or Black-box Techniques (K3)**

- LO- Write test cases from given software models using equivalence partitioning, boundary value analysis, decision tables and state transition diagrams/tables (K3)
- 4.3.1
- LO- Explain the main purpose of each of the four testing techniques, what level and type of testing could use the technique, and how coverage may be measured (K2)
- 4.3.2
- LO- Explain the concept of use case testing and its benefits (K2)
- 4.3.3

#### **4.4 Structure-based or White-box Techniques (K4)**

- LO- Describe the concept and value of code coverage (K2)
- 4.4.1
- LO- Explain the concepts of statement and decision coverage, and give reasons why these concepts can also be used at test levels other than component testing (e.g., on business procedures at system level) (K2)
- 4.4.2
- LO- Write test cases from given control flows using statement and decision test design techniques (K3)
- 4.4.3
- LO- Assess statement and decision coverage for completeness with respect to defined exit criteria. K4)
- 4.4.4

#### **4.5 Experience-based Techniques (K2)**

- LO- Recall reasons for writing test cases based on intuition, experience and knowledge about common defects (K1)
- 4.5.1
- LO- Compare experience-based techniques with specification-based testing techniques (K2)
- 4.5.2

#### **4.6 Choosing Test Techniques (K2)**

- LO- Classify test design techniques according to their fitness to a given context, for the test basis, respective models and software characteristics (K2)
- 4.6.1

## 5. Learning Objectives for Test Management

### 5.1 Test Organization (K2)

- LO- Recognize the importance of independent testing (K1)
- 5.1.1
- LO- Explain the benefits and drawbacks of independent testing within an organization (K2)
- 5.1.2
- LO- Recognize the different team members to be considered for the creation of a test team (K1)
- 5.1.3
- LO- Recall the tasks of typical test leader and tester (K1)
- 5.1.4

### 5.2 Test Planning and Estimation (K3)

- LO- Recognize the different levels and objectives of test planning (K1)
- 5.2.1
- LO- Summarize the purpose and content of the test plan, test design specification and test procedure documents according to the 'Standard for Software Test Documentation' (IEEE Std 829-1998) (K2)
- 5.2.2
- LO- Differentiate between conceptually different test approaches, such as analytical, model-based, methodical, process/standard compliant, dynamic/heuristic, consultative and regression-averse (K2)
- 5.2.3
- LO- Differentiate between the subject of test planning for a system and scheduling test execution (K2)
- 5.2.4
- LO- Write a test execution schedule for a given set of test cases, considering prioritization, and technical and logical dependencies (K3)
- 5.2.5
- LO- List test preparation and execution activities that should be considered during test planning (K1)
- 5.2.6
- LO- Recall typical factors that influence the effort related to testing (K1)
- 5.2.7
- LO- Differentiate between two conceptually different estimation approaches: the metrics-based approach and the expert-based approach (K2)
- 5.2.8
- LO- Recognize/justify adequate entry and exit criteria for specific test levels and groups of test cases (e.g., for integration testing, acceptance testing or test cases for usability testing) (K2)
- 5.2.9

### 5.3 Test Progress Monitoring and Control (K2)

- LO- Recall common metrics used for monitoring test preparation and execution (K1)
- 5.3.1
- LO- Explain and compare test metrics for test reporting and test control (e.g., defects found and fixed, and tests passed and failed) related to purpose and use (K2)
- 5.3.2
- LO- Summarize the purpose and content of the test summary report document according to the 'Standard for Software Test Documentation' (IEEE Std 829-1998) (K2)
- 5.3.3

### 5.4 Configuration Management (K2)

- LO-5.4.1 Summarize how configuration management supports testing (K2)

### 5.5 Risk and Testing (K2)

- LO- Describe a risk as a possible problem that would threaten the achievement of one or more stakeholders' project objectives (K2)
- 5.5.1
- LO- Remember that the level of risk is determined by likelihood (of happening) and impact (harm resulting if it does happen) (K1)
- 5.5.2
- LO- Distinguish between the project and product risks (K2)
- 5.5.3
- LO- Recognize typical product and project risks (K1)
- 5.5.4
- LO- Describe, using examples, how risk analysis and risk management may be used for test planning (K2)
- 5.5.5

### 5.6 Incident Management (K3)

- LO- Recognize the content of an incident report according to the 'Standard for Software Test Documentation' (IEEE Std 829-1998) (K1)
- 5.6.1
- LO- Write an incident report covering the observation of a failure during testing (K3)
- 5.6.2

## 6. Learning Objectives for Tool Support for Testing

### 6.1 Types of Test Tools (K2)

- LO- Classify different types of test tools according to their purpose and to the activities of the fundamental test process and the software life cycle (K2)
- 6.1.1
- LO- Explain the term test tool and the purpose of tool support for testing (K2)
- 6.1.2

### 6.2 Effective Use of Tools: Potential Benefits and Risks (K2)

- LO- Summarize the potential benefits and risks of test automation and tool support for testing (K2)
- 6.2.1
- LO- Remember special considerations for test execution tools, static analysis, and test management tools (K1)
- 6.2.2

### **6.3 Introducing a Tool into an Organization (K1)**

LO- State the main principles of introducing a tool into an organization (K1)

6.3.1

LO- State the goals of a proof-of-concept for tool evaluation and a piloting phase for tool implementation (K1)

6.3.2

LO- Recognize that factors other than simply acquiring a tool are required for good tool support (K1)

6.3.3