Table 1-A: Decomposition for Testing Basic Concepts and Definitions

A. Testing Basic Concepts and Definitions	*A1. Testing-related terminology*	Definitions of testing and related terminology
		Faults vs. Failures
	A2. Theoretical foundations	Test selection criteria/Test adequacy criteria (or stopping rules)
		Testing effectiveness/Objectives for testing
		Testing for defect removal
		The oracle problem
		Theoretical and practical limitations of testing
		The problem of infeasible paths
		Testability
	A3. Relationships of testing to other activities	Testing vs. Static Analysis Techniques
		Testing vs. Correctness Proofs and Formal
		Verification
		Testing vs. Debugging
		Testing vs. Programming
		Testing within SQA
		Testing within Cleanroom
		Testing and Certification

Table 1-B: Decomposition for Test Levels

B. Test Levels	*B1. The target of the test*	Unit testing
		Integration testing
		System testing
	B2. Objectives of testing	Acceptance/qualification testing
		Alpha and Beta testing
		Conformance testing/ Functional testing/ Correctness testing
		Installation testing
		Reliability achievement and evaluation by testing
		Regression testing
		Performance testing
		Stress testing
		Back-to-back testing
		Recovery testing
		Configuration testing
		Usability testing

Table 1-C: Decomposition for Test Techniques

C. Test Techniques	C1: (criterion “base on which tests are generated”)	*C1.1 Based on tester’s intuition and experience*	Ad hoc
		C1.2 Specification-based	Equivalence partitioning
			Boundary-value analysis
			Decision table
			Finite-state machine-based
			Testing from formal specifications
			Random testing
		C1.3 Code-based	Reference models for code-based testing (flow graph, call graph)
			Control flow-based criteria
			Data flow-based criteria
		C1.4 Fault-based	Error guessing
			Mutation testing
		C1.5 Usage-based	Operational profile
			SRET
		C1.6 Based on nature of application	Object-oriented testing
			Component-based testing
			Web-based testing
			GUI testing
			Testing of concurrent programs
			Protocol conformance testing
			Testing of distributed systems
			Testing of real-time systems
			Testing of scientific software
	C2: (criterion “ignorance or knowledge of implementation”)	*C2.1 Black -box techniques*	Equivalence partitioning
			Boundary-value analysis
			Decision table
			Finite-state machine-based
			Testing from formal specifications
			Error guessing
			Random testing
			Operational profile
			SRET
		C2.2 White-box techniques	Reference models for code-based testing (flow graph, call graph)
			Control flow-based criteria
			Data flow-based criteria
			Mutation testing
	C3 Selecting and combining techniques		Functional and structural
			Coverage and operational/Saturation effect

Table 1-D: Decomposition for Test Related Measures

D. Test Related Measures	*D.1 Evaluation of the program under test*	Program measurements to aid in planning and designing testing
		Types, classification and statistics of faults
		Remaining number of defects/Fault density
		Life test, reliability evaluation
		Reliability growth models
	D.2 Evaluation of the tests performed	Coverage/thoroughness measures
		Fault seeding
		Mutation score
		Comparison and relative effectiveness of different techniques

Table 1-E: Decomposition for Managing the Test Process

E. Managing the Test Process	*E.1 Management concerns*	Attitudes/Egoless programming
		Test process
		Test documentation and workproducts
		Internal vs. independent test team
		Cost/effort estimation and other process Measures
		Termination
		Test reuse and test patterns
	E.2 Test activities	Planning
		Test case generation
		Test environment development
		Execution
		Test results evaluation
		Problem reporting/Test log
		Defect tracking