

**CEN 6070
Software Quality Assurance and Testing
Summer 2004**

Exam 1 - Solutions

Answer all questions on the exam sheets. Use backs of pages as needed

Name: Solutions

First Part. Questions from the Galin Text. (5 points each)

I.

- a. Galin states the *SQA environment* has five different characteristics. One is *required teamwork*. List and briefly describe three others.

Pages 7-8: Contractual conditions, Subjection to customer-supplier relationship, required teamwork, cooperation and coordination with other software teams, and interfaces with other software systems.

- b. Briefly discuss the difference between *quality control* and *quality assurance*.

Pages 28-29: Quality control is a set of activities designed to evaluate the quality of a developed or manufactured product. The main objective of quality assurance is to minimize the cost of guaranteeing quality by a variety of activities performed throughout the development and manufacturing processes/stages.

- c. McCall's factor model classifies all software requirements into 11 software quality factors. The eleven factors are grouped into three categories – product operation, product revision, and product transition. One factor in each category is:
- i. Product operation factors: Correctness, **reliability, efficiency, integrity, usability.**
 - ii. Product revision factors: Maintainability, **flexibility, testability.**
 - iii. Product transition factors: Portability, **reusability, interoperability.**

List at least one other factor in each of the three categories. (see above)

Pages 37-38.

- d. Galin lists six management responsibilities in SQA. One is *definition of the quality policy*. List three others.

Pages 70-71.

Effective follow-up of quality policy implementation
Allocation of sufficient resources to implement quality policy
Assignment of adequate staff
Follow-up of compliance of quality assurance procedures
Solutions of schedule, budget, and customer relations difficulties

- e. Galin lists four factors important in deciding the extent of contract review efforts. One is *Degree of staff acquaintance with experience in the project area*. List at least one more.

Pages 82-83.

Project magnitude; Project technical complexity; Project organizational complexity.

Second Part– Questions from the Lectures and Notes

I. From Mr. Greg Jackson's lecture (2 points each). **One correct answer for each question.**

1. What problems result from printing test results to the standard output stream?
 - a. Does not test the methods API.
 - b. Requires human judgment to analyze results.
 - c. Scroll blindness.
 - d. **B and C**
 - e. All of the above.
2. What is a JUnit test fixture?
 - a. **The resources, primitive variables, and objects required to run a test.**
 - b. A collection of related test cases.
 - c. A method to release resources that were part of a test case.
 - d. A and B
 - e. All of the above.
3. What calls the JUnit TestCase *setUp()* method?
 - a. Each test method explicitly calls *setUp()*.
 - b. **The JUnit framework calls *setUp()* prior to each test method.**
 - c. The test fixture calls the *setUp()* method.
 - d. The TestSuite calls the *setUp()* method once for each TestCase.
 - e. None of the above.
4. The JUnit TestCase *assertEquals(String expected, String actual)* method:
 - a. **Tests that the value of *expected* equals the value of *actual*.**
 - b. Tests that the Object *expected* is the same Object as the *actual* Object.
 - c. Only displays a message on the JUnit console when the assert passes.
 - d. Displays a message on the JUnit console when the assert passes or fails.
 - e. B and D.

5. When a JUnit TestCase requires data to be initialized in a database prior to running a test; the following is true:
- The data can be initialized in the TestCase `setUp()` method.
 - The database should be returned to its original state by using the TestCase `setUp()` method.
 - A JUnit TestDecorator can be used to initialize the data.
 - A and C.**
 - All of the above.

II. Using mathematical induction prove $1 + 3 + 5 + \dots + (2n-1) = n^2$, $n \geq 1$. (10 points)

i) $n=1$; $[(2(1)-1)] = 2 - 1 = 1 = 1^2 = 1$

ii). **Assume** $1 + 3 + 5 + \dots + (2n-1) = n^2$;
Show $1 + 3 + 5 + \dots + (2n-1) + [2(n + 1) - 1] = (n + 1)^2$

$$\begin{aligned}
 1 + 3 + 5 + \dots + (2n-1) + [2(n + 1) - 1] &= n^2 + [2(n + 1) - 1] \text{ by the Inductive Hypothesis} \\
 &= n^2 + 2n + 1 \\
 &= (n + 1)^2
 \end{aligned}$$

III. In class, we considered the informal specification “Consider a search function that accepts an array of integers and some integer key of its parameters. The array members are sorted in increasing order. The function returns the array index of the member of the array whose value is equal to that of the key. The original input is unchanged. If the key is not matched, return an array index that is one number larger than the array’s maximum index.” The formal specification below seems to adequately capture the intended computation.

function Search (X : INTEGER-ARRAY; Key: INTEGER) return INTEGER;
Pre: exists i in X FIRST.. X LAST: $X(i) = \text{Key}$ and for-all i, j in X FIRST.. X LAST:
 $i < j$ implies $X(i) \leq X(j)$
Post: X ’(Search (X , Key)) = Key and $X = X$ ’
Error: Search (X , Key) = X LAST + 1