Linköpings universitet

Exam
TDDD04: Software Testing
Wednesday January 13, 2009

- No aids beyond writing equipment are accepted.
- Write clearly! Please use only one side of each paper and don't address more than one question per page.
- Justify your answers!
- Leave room for comments during grading.

Good luck!

- Inga hjälpmedel förutom skrivmedelar tillåtna. Skriv tydligt!
- Skriv bara på en sida av pappret och behandla bara en uppgift per pappersblad.
- Ge dina svar tydliga motiveringar.
- Lämna plats for kommentarer vid rättning.

Lycka till!

Mariam Kamkar: 013-281949

Department of Computer and Information Science
SE-581 83 Linköping, Sweden

(I) Basic Definitions:

1. Describe/Define the terminologies below: (2 p)
(a) Error
(b) Fault
(c) Test
(d) Test case

2. Name different levels of testing. (1 p)

(II) Unit & Integration Testing:

3. Identify the Equivalence Classes (EC) for the following specification. (2 p)

   Specification: the program accepts two to five inputs which are 3 digit integers greater than 100.

4. A right triangle is a triangle that has a 90 degree angle as one of its angles. One common addition to the triangle problem is to check for right triangles. Three sides constitute a right triangle if the Pythagorean relationship is satisfied: $c^2 = a^2 + b^2$. This change makes it convenient to require that the sides be presented in increasing order, i.e., $a \leq b \leq c$.

   Develop a decision table **and** test cases for the right triangle problem. (5 p)

5. Pairwise Testing: (2 p)
   (a) What is Pairwise testing?
   (b) When is it proper to use it?

6. Describe and give example for the following integration testing. (3 p)
   (a) Top-Down testing
   (b) Sandwich testing

7. Describe test stubs and drivers. When are they needed, and why? (2 p)

8. You work as a testing consultant and have accepted an assignment to help the Good Software Company. During your initial meeting, the CEO of the company says "Our customers have reported quite a number of bugs on our latest release. I'm astonished - before the release our testers told me that their unit tests covered more than 95% of the lines of code and that they had been busy clicking through the GUI for more than a week". Explain to the CEO:

   (a) Why bugs still occur in code covered by tests? (2 p)

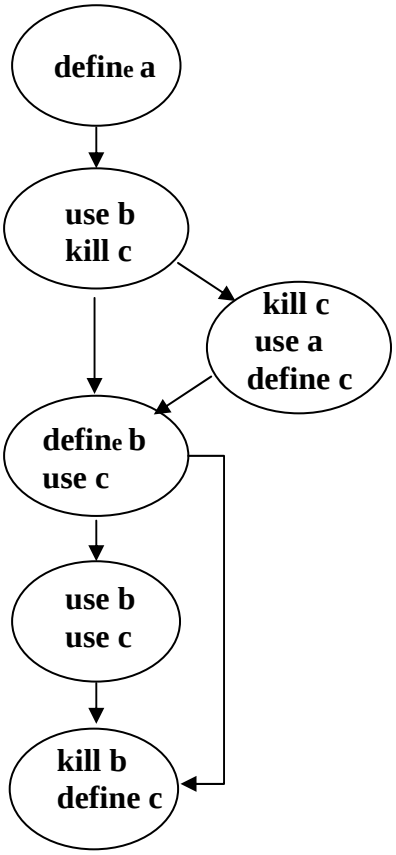
Department of Computer and Information Science
SE-581 83 Linköping, Sweden

(b) The benefits of using a tool, such as Abbot, to perform GUI tests. (2 p)

9. Early data flow analysis often is centered on a set of faults that are known as define/reference anomalies.

   Given the following notations and the control flow graph annotated with define-use-kill information, for each variable examine the define-use-kill patterns along the control flow graph and the kind of anomaly it could generate. (3 p)

   - **d**: defined, created, initialized, etc.
   - **k**: killed, undefined, released
   - **u**: used for something
   - **~d**: the variable does not exist, then it is defined
   - **~u**: the variable does not exist, then it is used
   - **~k**: the variable does not exist, then it is killed



(III) System Testing:

10. For the following causes and effects, design a cause-effect graph **and** propose a decision table for testing the software. (5 p)

    C1: Command is credit
    C2: Command is debit
    C3: Account number is valid
    C4: Transaction amount is valid

    E1: Print "invalid command"
    E2: Print "invalid account number"
    E3: Print "debit amount not valid"
    E4: debit account
    E5: credit account

11. Describe the following testing paradigm. (2p)
    (a) Scripted testing
    (b) Exploratory testing

12. Name **and** explain four attributes that describe the quality of a test case (good test case). (2 p)

13. What is regression testing? (1 p)

14. Given that it's impossible to test a program completely, what information do you think should be considered when deciding whether it is time to stop testing? (1 p)

15. Name and describe 3 kinds of Acceptance Testing. (3 p)

(IV) Test Automation:

16. Describe following Scripting techniques. (3 p)
    (a) Linear
    (b) Structured
    (c) Keyword-driven

17. Refactoring: (3 p)

    a) What is refactoring?
    b) When to refactor?

Department of Computer and Information Science
SE-581 83 Linköping, Sweden