Exam *Software Testing And Risk assessment*

April 23, 2024, 8:45–11:45

Remarks

- Please provide all answers in English.
- Please write your name on all sheets of paper you want marked.
- You are allowed a 1 A4 cheat sheet, double printed/written.
- The maximum score is 43 points; your grade is calculated by $\texttt{grade} = 1 + \frac{9}{43}\texttt{\#points}$.

Good luck!

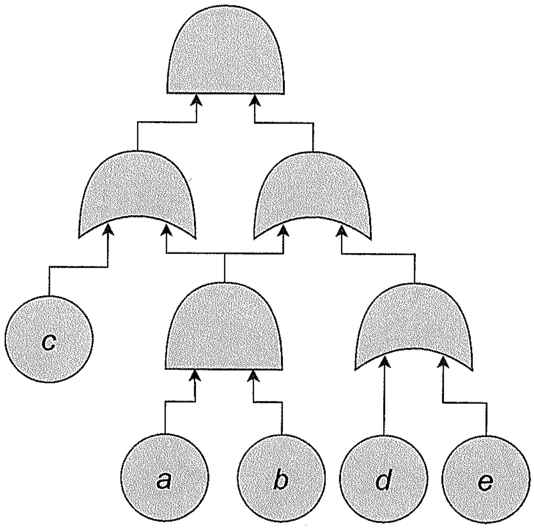
1 Risk management (5 points)

1. (1pt) What are the two main ingredients of every definition of Risk?
2. (1pt) What are the "Big 5" impact categories?
3. (3pt) Suppose your objective is "get a passing grade for the STAR exam". Give three risks associated with this objective, and for each risk a strategy to handle that risk. Sketch the risks and their strategies on the risk matrix.

2 Static fault trees (7 points)

Consider the fault tree below.

1. (1 point) List all of its minimal cut sets.
2. (1 point) Suppose $p_a = p_b = p_c = 0.5$ and $p_d = p_e = 0.25$. Approximate the failure probability using the cut set method; explain your answer.
3. (3 points) Give the BDD of this fault tree, with variable ordering $a < b < c < d < e$.
4. (2 points) Use the BDD to calculate the failure probability of the fault tree; explain your answer.



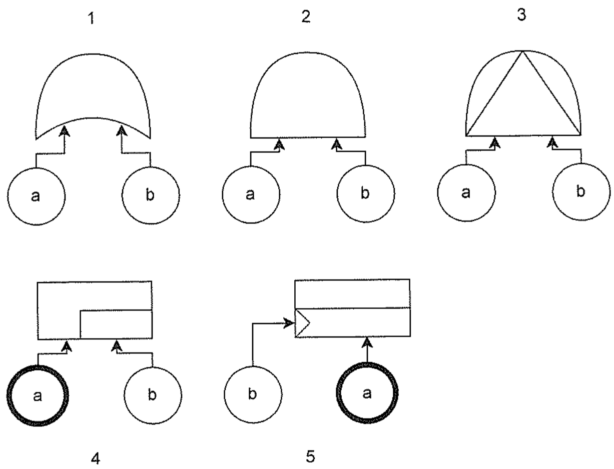
3 Dynamic fault trees (6 points)

A number of dynamic fault trees are pictured below; in each a has failure rate λ and b has failure rate μ. For clarity, the top event has a bold outline in DFTs 4 and 5.

(a) (2pts) Two of these DFTs are equivalent, that is, their TLE fails under the exact same conditions. Which ones? Explain your answer.

(b) (2pts) One of these DFTs (not the ones from (a)) has a nonzero probability of never failing at all, i.e., the failure probability does not go to 1 as time goes to ∞. Which one? Explain your answer.

(c) (2pts) Except for the DFT you mentioned in (b), order the remaining DFTs by their expected failure time. The two DFTs you mentioned in (a) will have the same failure time. Explain your answer.



4 Black box testing (5 points)

Consider the function `int Weirdsort(int a, int b)` that given integers a and b, returns them as a sorted list of two elements. The integers a and b have to be in $\{0, 1, 2, \cdots, 100\}$, and they cannot be equal; if either of these requirements is not met, an error is returned.

1. (2 points) Using equivalence partitioning, divide the input into suitable equivalence classes. Explain your answer. You may assume this function has already been type checked, so you do not have to check for non-integer values or the wrong number of arguments.
2. (1 point) Give a test suite that covers all equivalence classes described above.
3. (2 points) Extend your test suite using boundary value analysis. Explain your answer.

5 White box testing (6 points)

For each of the following, give an example of a program (in (pseudo)code) and a test suite that provides

1. (2pt) Statement coverage but not condition coverage;
2. (2pt) Condition coverage but not statement coverage;
3. (2pt) Path coverage but not statement coverage.

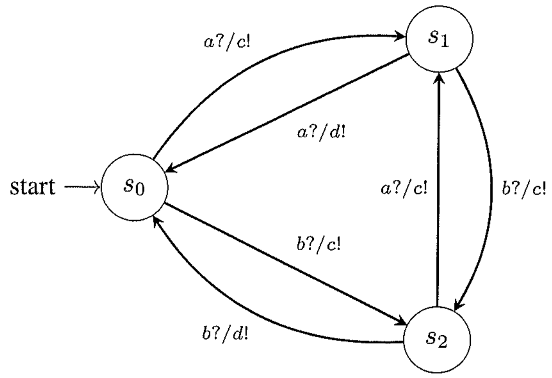
By coverage we mean 100% coverage. Explain your answers, including what you consider statements for statement coverage.

6 Mealy machines (7 points)

Consider the Mealy machine below.

1. (2 pts) Show that s_0 does not have a unique input-output sequence.
2. (2 point) Give a characterising set for each state.
3. (1 point) Give the part of a transition testing suite that tests the transition $s_2 \xrightarrow{b?/d!} s_0$.
4. (2 points) Under what assumptions on the implementation is a full transition testing suite:
   (a) sound?
   (b) complete?

   Explain your answer.



7 QLTS (7 points)

Consider the QLTS $\mathcal{Q}$ below; it has input set $L_I = \{a?, b?\}$ and output set $L_O = \{c!\}$.

1. (1 point) Give a test T with at least 2 **fail** states and 2 **pass** states such that $\mathcal{Q}$ passes T.
2. (1 points) Explain why $\mathcal{Q}$ **ioco** $\mathcal{Q}$ is not defined.
3. (2 points) Create a QLTS $\mathcal{I}$ such that $\mathcal{I}$ **ioco** $\mathcal{Q}$ is defined and holds.
4. (3 points) Let $\mathcal{A}$ and $\mathcal{B}$ be two QLTS such that $\mathcal{A}$ is input-enabled, and obtained from $\mathcal{B}$ by adding some transitions (their set of states, initial states, and input and output sets are also the same). Is it always true that $\mathcal{A}$ **ioco** $\mathcal{B}$? Give a proof or a counterexample.