

Software Testing Comprehensive Exam

November, 2002

ANSWER ANY FOUR OF THE FOLLOWING EIGHT QUESTIONS.

Do not attempt more than four questions on the exam. If you answer five, pick your best four and *cross out the fifth*, the one you don't want us to grade. If you provide more than four answers, we will pick the four that are the most convenient for us to grade and we will ignore the other(s). You will not get credit for the best four answers. If you are unlucky, we might grade only your worst four answers. Our selection of the appropriate four questions to grade will be final. If you don't like that, don't answer more than four questions.

1. Explain the idea of *fault injection* in software testing. What is fault injection? How is it used? What types of bugs does it find?
2. Explain the concept of output-based testing techniques. Name at least one attack that falls under the category of output-based testing and explain how the attack is carried out.
3. Data structures are used by software to temporarily store internal data. Explain how to test internally stored data. What types of data bugs can you name? What are the testing considerations associated with stored data?
4. Discuss how a software system works within an operating system and interacts with its environment. What are the different users within an environment that can interact with a system under test? What are the inputs that come from these users? How does the environment complicate the testing process?
5. Explain the definition of "coverage" in testing. Describe three different types of coverage that you can measure. For each one, describe a type of bug that you would be certain to find with this type of coverage and describe a type of bug that you might miss even if you achieved 100% of this type of coverage.
6. Consider testing the Microsoft Word function that lets you enter data into a table in the word processor.
 - How would you develop a list of risks for this capability? (If you are talking to people, who would you ask and what would you ask them?) (If you are consulting books or records or databases, what are you consulting and what information are you looking for in it?)
 - Why is this a good approach for building a list of risks?
 - List 5 risks associated with this function.
 - For each risk, briefly (very briefly) describe a test that could determine whether there was an actual defect and explain why this is a good test for this risk.

7. Ostrand & Balcer described the category-partition method for designing tests. Their first three steps are:

- 1) Analyze
- 2) Partition, and
- 3) Determine constraints

Apply their method to this function:

- I, J, and K are unsigned integers. The program calculates
 $K = \text{IntegerPartOf}(\text{SquareRoot}(I*J))$.

For this question, consider only cases in which you enter integer values into I and J. Do an equivalence class analysis from the point of view of the effects of I and J (jointly) on the variable K. List the values you would enter into I and J and explain why.

8. The *oracle problem* is the problem of finding a method that lets you determine whether a program passed or failed a test.

- Suppose that you were doing automated testing of the Microsoft word function, *Word Count* (which counts characters, words, paragraphs and pages in the document).
- Describe three different oracles that you could use or create to determine whether this feature was working. For each of these oracles:
 - identify a bug that would be easy to detect using the oracle,
 - identify another bug that your oracle would be more likely to miss
 - explain whether and why it would be hard or easy to use this oracle in automated testing.