

Software Testing Comprehensive Exam

Spring 2011

ANSWER ANY FOUR OF THE FOLLOWING FIVE 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 grade the answers to the first four questions and ignore the fifth.

1. There are several different definitions of *acceptance testing*. For purposes of this question, acceptance testing is done by a representative of the customer, and the goal of acceptance testing is to answer the question, "Should we pay for this program?"
What kinds of tests are you *unlikely* to run as part of an acceptance test effort and what kinds of bugs are you more likely to miss as a result? Give three examples of each and explain your reasoning.
2. A program is structured as follows:
 - It starts with a loop, the index variable can run from 0 to 20. The program can exit the loop normally at any value of the index.
 - Coming out of the loop, there is a case/switch statement that will branch to one of 10 places depending on the value of X. X is a positive, non-zero integer. It has a value from 1 to 10.
 - In 9 of the 10 cases, the program executes X statements and then goes into another loop. If X is even, the program can exit the loop normally at any value of its index, from 1 to X. If X is odd, the program goes through the loop 666 times and then exits. In the 10th case (I am explicitly NOT specifying which of the 10 values of X corresponds to the 10th case), the program exits.
 - (a) Ignore the possibility of invalid values of the index variable or X. How many paths are there through this program? Please show and/or explain your calculations.
 - (b) Which values of X should you test? Explain why you would test these values and why you would not test the others.
3. Consider *domain testing* and *specification-based testing*.
 - a. What kinds of bugs are you more likely to find with domain testing than with specification-based testing? Give at least two examples of types of bugs and explain why domain testing is more powerful for each kind.
 - b. What kinds of bugs are you more likely to find with specification-based testing than with domain testing? Give at least two examples of types of bugs and explain why domain testing is less powerful for each kind.
4. Define a scenario test and describe the characteristics of a good scenario test. Suppose that scenario testing is your primary approach to testing. What controls would you put into place to ensure good coverage? Describe at least three and explain why each is useful.
5. Why is it important to design maintainability into automated regression tests? Describe some design (of the test code) choices that will usually make automated regression tests more maintainable.