中山大学软件学院 06、07 级软件工程专业(2009 学年秋季学期)

《软件测试技术》期末试题答案(A 卷)

一、判断对错、并说明理由（每小题 5 分，共 50 分）

(1) A good tester relentlessly strives for perfection.

答：错。好的测试员知道何时完美无法企及，何时达到“够好”。
A1: False. A good tester knows when perfection isn't attainable and when "good enough" is reached.

(2) Visiting all the states that a program has assures that you've also traversed all the transitions among them.

答：错。想想游览遍布美国的 50 个不同的城市。可以制定到达每个城市的旅游计划，但是不可能走遍所有城市之间的道路——这将是走遍美国的所有道路。
A2: False. Think of visiting 50 different cities spread out across the entire United States You could plan a trip that would take you to each city, but it would be impossible for you to travel all the roads that connect to all the cities that would be all the roads in the entire country.

(3) It's an unfair test to perform stress testing at the same time you perform load testing.

答：错。任何测试都是合理的。软件测试员的任务是发现软件缺陷。但是，由于软件测试的实质，在这种条件下发现的任何缺陷可能都不会修复。
A3: False. No test is ever unfair. Your job is to find bugs. But, the realities of software testing may apply and any bugs you find in this situation may not be fixed.

(4) Static white-box testing can find missing items as well as problems.

答：对。遗漏的问题比普通的问题更重要，要通过静态白盒测试可以发现。当根据公布的标准和规范检查代码，在正式审查仔细分析时，遗漏的问题就显而易见了。

A4: True. Missing items are arguably more important than normal problems and can be found through static white-box testing. When the code is checked against published standards and guidelines and carefully analyzed in formal reviews, missing items become obvious.

(5) Always design your black-box test cases first.

答：对。基于对软件行为操作的认识程度来设计测试用例，然后利用白盒测试技术进行检查使其更加有效。

A5: True. Design your test cases based on what you believe the software is supposed to do. Then use white-box techniques to check them and make them most efficient.

(6) You can perform dynamic black-box testing without a product specification or requirements document.

对。该技术称为探索测试，基本上把软件用作产品说明书。这不是理想的过程，但是急了也能用。最大的风险是不知道特性是否被遗漏。
A6: True. The technique is called exploratory testing, and you essentially use the software as though it's the product spec. It's not an ideal process, but can work okay in a pinch. The largest risk is that you won't know if a feature is missing.

(7) All software must undergo some level of compatibility testing.

答：错。有少数独立使用、专用、不与任何外界打交道的软件不需要进行兼容性测试。但是，除此之外99%的软件都必须进行某种程度的兼容性测试。
A7: False. There will be a few rare, standalone, proprietary first versions of software out there that don't interact with anything. For the other 99 percent of the world, though, some level of compatibility testing will be necessary.

(8) All software has a user interface and therefore must be tested for usability.

答：对。即使嵌入再深的软件终将以某种形式显露在用户面前。不要忘记UI可以简单到一个开关和一个灯泡，可以复杂到飞行模拟器，即使软件在代码库中只有一个代码模块，其接口也要以变量和参数的形式显露在可以作为用户的程序员面前。
A8: True. Eventually, even the most deeply embedded software is exposed, in some way, to a user. Keep in mind that the UI may be as simple as a switch and a light bulb or as complex as a flight simulator. Even if the software is a single module in a code library, its interface, in the form of variables and parameters, is exposed to a programmer who can be a user, too.

(9) Testing error messages falls under documentation testing.

答：对。但这不仅仅是文档测试。信息的内容需要作为文档测试；而强制信息显示和保证显示信息的准确无误是代码测试的任务。
A9: True. But, it's not just documentation testing. The content of the message needs to be tested as documentation, but forcing the message to appear and assuring that the correct message is displayed is testing the code.

(10) An invasive tool is the best type because it operates closest to the software being tested.

答：错。入侵式工具在一些情形下可以提供更好的信息和控制，但是它具有可能影响软件和测试结果的不利一面。最好是仔细评估每种情况，选择最适用的工具，

且副作用最小。
A10: False. An invasive tool may provide you with better information and control in some situations, but it has the possible downside of affecting the software and the test results. It's best to carefully evaluate each situation and select the tool that works best, with the fewest side effects.

二、简答题（每小题 5 分，共 50 分）

(1) What's the difference between a tool and automation?

答：测试工具有助于测试，简化手工完全测试任务。自动化也是一种工具，但是它的执行不需要人工干预。想一想在木匠呼呼大睡时电锯和钉锤能盖好房子么？
A1: A test tool will help you test, making it easier for you to perform a manual testing task. Automation is also a tool, but it will run without your intervention. Think power saw and hammer building a house while the carpenter sleeps.

(2) Besides being more formal, what's the big difference between inspections and other types of reviews?

答：主要区别是，检查时在场的不是代码的原创者。这迫使另一个人完全理解要检查的软件。这比让他人只是审查软件寻找软件缺陷更有效。
A2: The key difference is that with inspections, a person other than the original author of the code is the presenter. This obliges another person to fully understand the software being inspected. It's much more effective than having others simply review the software for bugs.

(3) Can a software tester perform white-box testing on a specification?

答：是的，白盒测试就是使用如何设计影响如何测试的概念进行的。测试员可以参加焦点人群、易用性研究和市场会议，了解用于定义功能特性和整个产品的过程。但是这存在一定的风险，因为这些信息诱使测试员倾向于假定说明书时正确的。
A3: Yes, white-box testing is simply using information about how something is designed to influence how the testing is done. The tester could attend the project's focus groups, usability studies, and marketing meetings to understand the underlying process being used to design the features and the overall product. With that information she could more effectively test the resulting specification. There is a risk, though, that this information could bias the tester into assuming that the spec is correct.

(4) What's the goal of a software tester?

答：软件测试员的目标是尽可能早些找出软件缺陷，并确保其得以修复。
A4: The goal of a software tester is to find bugs, find them as early as possible, and make sure they get fixed.

(5) Why is it impossible to test a program completely?

答：除了极短小的简单程序，完全测试需要太多输入、输出和分支组合。此外，软件说明书也许不客观，可以用多种方式解释。
A5: With any software other than the smallest and simplest program, there are too many inputs, too many outputs, and too many path combinations to fully test. Also, software specs can be subjective and be interpreted in different ways.

(6) Explain what a tester should worry about with this line from a spec: The software will allow up to 100 million simultaneous connections, although no more than 1 million will normally be used.

答：可测试性，典型应用只有一百万个倒无关紧要。如果产品说明书有一亿种可能性，那么，一亿个连接都要测试。测试员需要设法测试那么多可能性，或者让说明书作者把最大可能性降低到接近典型应用的数目。
A6: Testability. It doesn't matter that typical usage is only 1 million connections. If the specification states that 100 million are possible, the 100 million must be tested. The tester needs to find a way to test something this large or get the spec writer to reduce the maximum possible number to something closer to what's typical.

(7) What are a few cautions to consider with a beta test program？

答：beta 测试不能代替有组织、有计划、有条理的测试方法——在通常意义上的软件缺陷寻找方面没有优势。测试员应该知道 beta 测试者的水平、设备，并要确保从测试中得到预期的结果。

A7: A beta test is no substitute for an organized, planned, methodical test approach it's not good at general bug finding. You should know who the beta testers are in regards to their experience level, equipment, and needs to ensure that you get what you expect out of the test.

(8) If there's no definitive right or wrong user interface, how can it be tested?

答：软件测试员应当检查其是否符合 7 个重要原则：符合标准和规范、直观、一致、灵活、舒适、正确和实用。
A8: Software testers should check that it meets seven important criteria: That it follows standards and guidelines, that it's intuitive, consistent, flexible, comfortable, correct, and useful.

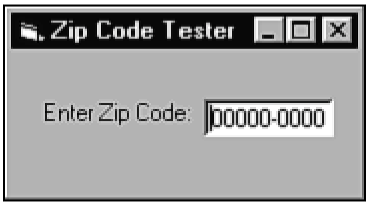
(9) What is gray-box testing?

答：灰盒测试是可以边看着代码、边利用代码的信息帮助测试。它不像白盒测试一样详细地检查代码。代码用来协助测试，但是测试并不完全基于代码。
A9: Gray-box testing is when you can take a peek at the underlying code and use that information to help you test. You're not examining it to the same level of detail as you

would with white-box testing. It's helping you test, but you're not basing all of your tests on it.

(10) Assume that you have a 10-character-wide ZIP code text box, such as the one shown in Figure 1. What equivalence partitions would you create for this text box?

Figure 1. A sample ZIP code text box that holds up to 10 characters.



答：至少应该有一下的等价划分，但是还可以相处更多：
合法的 5 位数字邮政编码。合法是指所有字符都是数值，不是指投入使用的现有邮政编码——但这可以构成另一个区间。
合法的 9 位数字（带连线的 9 位数字）邮政编码。
5 位以下数字。例如只有 4 位数字。
9 位以下数字。例如只有 8 位数字。
5 位以上数字。例如不带连线的 8 位数字。这是否与 9 位以下数字区间相同呢？
9 位以上数字，尽管不可能输入 9 位以上的带连线的数字，但是测试员应该尝试一下。
10 位数字，无连线。与 9 位以上数字区间稍有差别。
连线位置不对。
连线不止一条。
无数字和无连线。

A10: You should have at least these following equivalence partitions, although you may think of more:

- Valid 5-digit ZIP codes. Valid means that they're numeric digits, not that they are existing, in-use ZIP codes although that could be another partition.
- Valid 9-digit (9 digits with a dash) ZIP codes.
- Short 5-digit. Have only 4 numbers, for example.
- Short 9-digit. Have only 8 numbers, for example.
- Long 5-digit. Have 8 digits without a dash, for example. Hmm, is this the same as a short 9-digit partition?
- Long 9-digit. It may not be possible to type in more than 9 digits and a dash, but you should try.
- 10 digits, no dash. This is a little different than a long 9-digit partition.
- The dash in the wrong place.
- More than one dash.
- Non-digit and non-dash entries.