

试卷代号:2258

座位号

中央广播电视大学 2009—2010 学年度第一学期“开放专科”期末考试

信息系统测试 试题

2010 年 1 月

题号	一	二	三	四	总分
分数					

得分	评卷人

一、选择题(每小题 2 分,共 30 分)

- 关于软件测试的目的,下列说法哪个是错误的()。
 - 测试是程序的运行过程,目的在于发现错误
 - 一个好的测试用例在于能够发现至今未发现的错误
 - 一个成功的测试是发现了至今未发现的错误的测试
 - 测试的目标是以最少的时间和人力改正软件中潜在的各种错误和缺陷
- 下列哪种测试方法不属于人工测试()。
 - 软件审查
 - 代码审查
 - 人工走查
 - 计算机运行程序
- 下列哪种测试在系统测试阶段进行()。
 - 有效性测试
 - 验收测试
 - 可靠性测试
 - α 测试与 β 测试
- ()不是单元测试要解决的问题。
 - 性能测试
 - 模块接口
 - 出错处理
 - 边界条件
- 下面测试方法属于静态分析方法的是()。
 - 功能确认与接口测试
 - 覆盖率分析
 - 性能分析
 - 生成引用表

6. 在软件开发初期,即需求分析阶段制定的测试计划是()。
- A. 详细测试计划
 - B. 概要测试计划
 - C. 需求分析测试计划
 - D. 测试实施计划
7. 下列哪种说法是正确的()。
- A. 验收测试是以用户为主的测试,软件开发人员和质量保证人员也应参加,并由用户参加设计测试用例和分析测试的输出结果
 - B. 有效性测试是在模拟的环境下,运用白盒测试方法,验证所测软件是否满足需求规格说明书列出的要求
 - C. α 测试是由软件用户在实际使用环境下进行的测试,主要目的是测试系统的可支持性
 - D. β 测试是由用户在开发环境下进行的测试,在确认测试阶段后期对软件进行测试
8. ()不是软件测试的原则。
- A. 提早原则
 - B. IPO 原则
 - C. 独立测试原则
 - D. 无错误原则
9. 下面哪种情况不是软件正确性的描述()。
- A. 程序编写得无语法错误
 - B. 程序运行中未发现明显的运行错误
 - C. 程序中无不适当语句
 - D. 程序运行中只能通过正确数据,并给出正确的结果
10. 等价分类法的关键是()。
- A. 选择测试用例
 - B. 找出等价类的边界条件
 - C. 划分等价类
 - D. 通过输入条件,确定无效等价类
11. 可以为管理人员和决策者提供管理信息的信息系统是()。
- A. 事务处理系统
 - B. 管理信息系统
 - C. 决策支持系统
 - D. 指挥信息系统
12. 单元测试一般由开发人员在编码阶段完成,因而单元测试通常采用的测试方法()。
- A. 黑盒测试方法
 - B. 静态分析方法
 - C. 动态测试方法
 - D. 白盒测试方法

13. 动态测试的功能包括()。
- A. 程序错误分析、一致性检查
 - B. 程序错误分析、性能分析、内存分析
 - C. 生成引用表、程序错误分析、一致性检查
 - D. 确认与接口测试、覆盖率分析、性能分析、内存分析

14. 对 Web 服务系统进行测试时,要让它们以某种特定的方式运行代码,这种测试超越了功能验证测试,目的是要弄清楚被测试的 Web 服务是不是不仅能做我们认为它能做的事,而且在超出正常的情况下仍然能够继续正常运行。这种测试是()。

- A. 连接速度测试
- B. 压力测试
- C. 负载测试
- D. 可用性测试

15. 将基于功能的和基于实现的测试设计结合在一起的测试,我们称这种测试为()。

- A. 白盒测试
- B. 黑盒测试
- C. 灰盒测试
- D. 基于故障的测试

得 分	评卷人

二、判断题(正确的划√,错的划×,每题 2 分,共 20 分)

1. 数据是人们对原始事实进行加工处理以后得到的结果。()
2. 信息系统测试只需要检查和验证系统是否完成系统功能,无需从最终用户的角度进行安全性和可用性测试。()
3. 软件测试是为了证明程序无错而运行程序的过程。()
4. 单元测试大量采用黑盒测试方法;而通常的集成测试也只是对整个编译后的程序做基于白盒测试的集成测试。()
5. 对一个成功的软件来说,只有经过编码、测试、再编码、再测试,这样一个不断的循环过程,才能获得较高的产品质量。()
6. 一个 Web 应用系统是网络集约的,Web 应用系统不是为了某个或某些特定用户量身定做的,而且一般是采取非持续演化的模式。()
7. 即使在小型软件测试项目上,也可能有数千个测试用例,在项目开发期间内避免多次执行同样的测试,从而造成人力、物力、财力上的浪费。()
8. 测试用例,就是以发现错误为目的而精心设计的一组测试数据和测试执行步骤。()

9. 软件问题报告是测试实施过程中最重要的文档。它记录了软件问题发生的环境,如各种资源的配置情况,软件问题的再现步骤以及软件问题性质的说明。更重要的是它还记录着软件问题的处理进程。()

10. 当具有“验证问题”职责的角色认为问题已真正得到修改或合理的解释时,状态就转移到“解决”状态,否则就转移到“打开”状态或者“关闭”状态。()

得分	评卷人

三、简答题(每小题 6 分,共 30 分)

1. 列举软件缺陷的特征?
2. 软件测试的对象是什么?
3. 什么是 Web 信息系统?
4. 什么是黑盒测试? 什么是白盒测试? 二者主要有何区别?
5. 设计测试用例的原则是什么?

得分	评卷人

四、应用题(每小题 10 分,共 20 分)

1. 黑盒测试中测试用例的设计方法有哪几种? 各自的特点是什么?
2. 白盒测试中都有哪几种覆盖方法? 各自的作用是什么?

试卷代号:2258

中央广播电视大学 2009—2010 学年度第一学期“开放专科”期末考试

信息系统测试 试题答案及评分标准

(供参考)

2010 年 1 月

一、选择题(每小题 2 分,共 30 分)

- | | | | | |
|-------|-------|-------|-------|-------|
| 1. D | 2. D | 3. C | 4. A | 5. D |
| 6. B | 7. A | 8. D | 9. D | 10. C |
| 11. B | 12. D | 13. D | 14. B | 15. C |

二、判断题(正确的划√,错的划×,每题 2 分,共 20 分)

5,8,9 是正确的,其余是错误的。

三、简答题(每小题 6 分,共 30 分)

1. 答:符合下列 5 种特征之一的软件错误认为是软件缺陷。

- (1)软件未达到软件产品需求说明书中指明的要求。
- (2)软件出现了软件产品需求说明书中指明不会出现的错误。
- (3)软件功能超出了软件产品需求说明书中指明的范围。
- (4)软件未达到软件产品需求说明书中虽未指明但应达到的要求。
- (5)测试人员认为难以理解、不易使用、运行速度缓慢或者最终用户认为不好的问题。

2. 答:需求分析、概要设计、详细设计以及编码等各阶段所得到的文档,包括需求规格说明、概要设计说明、详细设计说明以及源程序,都应该是软件测试的对象。

3. 答:Web 应用系统就是在—台服务器上运行 Web 应用的服务器端程序,从而使与服务器建立连接的所有客户机都能共享的使用 Web 应用系统提供的内容和功能。

4. 答:黑盒测试是基于产品的外部功能来规划测试,检查程序各个功能是否实现,并检查其中的错误的一种测试。

白盒测试是基于产品的内部结构来规划测试,检查程序内部操作是否按规定运行,各部分代码是否被充分覆盖一种测试。

5. 答:设计测试用例基本的原则是:

- (1)一个好的测试用例在于能够发现至今没有发现的错误;
- (2)测试用例应由测试输入数据和与之对应的预期输出结果这两部分组成;
- (3)在测试用例设计时,应当包含合理的输入条件和不合理的输入条件。

四、应用题(每小题 10 分,共 20 分)

1. 解:黑盒测试技术中设计测试用例的几种基本方法有:等价类划分、边界值分析、错误推测法和因果图。

(1)等价类划分是一种典型的黑盒测试方法。使用这一方法时,完全不考虑程序的内部结构,只依据程序的需求规格说明书来设计测试用例。等价类划分方法把所有可能的输入数据,即程序的输入域划分成若干部分,然后从每一部分中选取少数有代表性的数据作为测试用例。

(2)边界值分析是对等价类划分方法的补充。在软件设计和程序编写中,常常对于规格说明书中的输入域或输出域的边界不够注意,以致形成一些差错。长期的测试工作经验告诉我们,大量的错误都是发生在输入或输出范围的边界上,而不是发生在输入输出范围的内部。因此,针对各种边界情况设计测试用例,可以查出更多的错误。

(3)错误推测法是一种基于经验和直觉推测程序中所有可能存在的各种错误,从而有针对性的设计测试用例的方法。基本思想是,列举出程序中所有可能有的错误和容易发生错误的特殊情况,根据他们选择测试用例。

(4)由于等价类划分方法没有考虑到输入情况的各种组合,也没有考虑到各个输入情况之间的相互制约关系,所以,我们采用因果图方法解决这个问题。

采用因果图方法能够高效地选择测试用例,同时还能指出程序规格说明中存在着什么问题。

2. 解:由于覆盖测试的目标不同,逻辑覆盖包括下列几种常用的覆盖方法:

(1)语句覆盖:

语句覆盖就是设计若干个测试用例,运行被测程序,使得每一可执行语句至少执行一次。

(2)分支覆盖(又称为判定覆盖):

分支覆盖就是设计若干个测试用例,运行被测程序,使得程序中每个判断的取真分支和取假分支至少经历一次。

(3)条件覆盖:

条件覆盖就是设计若干个测试用例,运行被测程序,使得程序中每个判断的每个条件的可能取值至少执行一次。

(4)判定—条件覆盖:

判定—条件覆盖就是设计足够的测试用例,使得程序中每个判断中每个条件的可能取值至少执行一次,每个判断中的每个分支至少执行一次。

(5)路径覆盖

路径测试就是设计足够的测试用例,使程序中所有可能的路径都执行一次。