

软件测试试题

- 1.下面说法正确的是 (C)。
 - A. 经过测试没有发现错误说明程序正确
 - B. 测试的目标是为了证明程序没有错误
 - C. 成功的测试是发现了迄今尚未发现的错误的测试
 - D. 成功的测试是没有发现错误的测试
- 2.不属于白盒测试的技术是 (C) 。
 - A. 语句覆盖
 - B. 判定覆盖
 - C. 边界值分析
 - D. 基本路径测试
- 3.单元测试主要针对模块的几个基本特征进行测试，该阶段不能完成的测试是 (A)。
 - A. 系统功能
 - B. 局部数据结构
 - C. 重要的执行路径
 - D. 错误处理
- 4.软件测试过程中的集成测试主要是为了发现 (B) 阶段的错误。
 - A.需求分析
 - B.概要分析
 - C.详细设计
 - D.编码
- 5.软件测试不需要了解软件设计的 (D) 。
 - A.功能
 - B.内部结构
 - C.处理过程
 - D.条件
6. (C) 方法根据输出对输入的依赖关系设计测试用例。
 - A.路径测试
 - B.等价类
 - C.因果图
 - D.边界值分析
- 7.通常，在 (D) 的基础上，将所有模块按照设计要求组装成系统
 - A.组装测试
 - B.系统测试
 - C.验收测试
 - D.单元测试
- 8.实际的逻辑覆盖测试中，一般以 (C) 为主设计测试用例。

A. 条件覆盖 B. 判定覆盖 C. 条件组合覆盖 D. 路径覆盖

9.使用白盒测试方法时，确定测试数据应根据（ A ）和指定的覆盖标准。

A.程序内部逻辑 B.程序的复杂度 C.使用说明书 D.程序的功能

10.与设计测试用例无关的文档是（ A ）。

A.项目开发计划 B.需求规格说明书 C.设计说明书 D.源程序

11、软件测试技术可以分为静态测试和动态测试，下列说法中错误的是（ D ）

A. 静态测试是指不运行实际程序，通过检查和阅读等手段来发现程序中的错误。

B. 动态测试是指实际运行程序，通过运行的结果来发现程序中的错误。

C. 动态测试包括黑盒测试和白盒测试。

D. 白盒测试是静态测试，黑盒测试是动态测试。

12、在软件测试阶段，测试步骤按次序可以划分为以下几步：（ A ）

A. 单元测试、集成测试、系统测试、验收测试

B. 验收测试、单元测试、系统测试、集成测试

C. 单元测试、集成测试、验收测试、系统测试

D. 系统测试、单元测试、集成测试、验收测试

13、系统测试中主要用到的测试技术是（ B ）

A. 回归测试 B. 黑盒测试 C. 白盒测试 D. 功能测试

14、对软件的性能测试、（ B ）测试、攻击测试都属于黑盒测试。

A. 语句 B. 功能 C. 单元 D. 路径

15、在用白盒测试中的逻辑覆盖法设计测试用例时，有语句覆盖、分支覆盖、条件覆盖、

判定-条件覆盖、条件组合覆盖和路径覆盖等，在下列覆盖中，（ D ）是最强的覆盖准则。

A. 语句覆盖 B. 条件覆盖 C. 判定-条件覆盖 D. 路径覆盖

16、验收测试主要是由 (A) 来完成的。

A. 用户 B. 测试人员 C. 开发人员 D. 高级经理

17、在用白盒测试中的逻辑覆盖法设计测试用例时，有语句覆盖、分支覆盖、条件覆盖、判定-条件覆盖、条件组合覆盖和路径覆盖等，其中 (A) 是最弱的覆盖准则。

A. 语句覆盖 B. 条件覆盖 C. 判定-条件覆盖 D. 路径覆盖

18、下面是一段求最大值的程序，其中 datalist 是数据表，n 是 datalist 的长度。

```
int GetMax(int n, int datalist[ ])
{
    int k=0;
    for ( int j=1; j<n; j++ )
        if ( datalist[j] > datalist[k] ) k=j;
    return k;
}
```

请问该程序段的 McCabe 环路复杂性为多少？ (A)

A. 2 B. 3 C. 4 D. 5

二、填空题：

1. 软件测试是为了尽可能多地发现软件中存在的 错误，将 测试结果 作为纠错的依据。

2. 白盒测试中，控制流测试是面向程序的 结构，数据流测试是面向程序的 变量。

3. 集成测试中的两种集成模式是 非增量测试模式 和 增量测试模式。

4. 单元测试的依据是 详细设计说明书 和 源程序。

5. 测试用例由 输入数据 和预期的 输出数据 两部分组成。

三、判断题：

- (×) 1.为了提高测试效率，应该取一切可能的输入数据作为测试数据。
- (×) 2.发现错误多的程序模块，残留在模块中的错误也多。
- (√) 3.发现错误多的程序模块，残留在模块中的错误也多。
- (√) 4.等价类划分方法将所有可能的输入数据划分成若干部分，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。
- (×) 5.系统测试的目标是要找出软件在与系统其他部分协调工作时出现的所有故障。

。

四、简答题：

1.为什么说软件测试是软件开发中不可缺少的重要一环，但不是软件质量保证的安全网？

软件测试是软件开发中不可缺少的重要一环，原因是：测试的工作量约占整个项目开发工作量的 53%~87%左右，如果是关系到人的生命安全的软件，测试的工作量还要更大。

软件测试是对软件需求分析、设计规格说明、编码的最终复审，是软件质量保证的关键步骤。

但软件测试不是软件质量保证的安全网，因为软件测试只能发现错误，不能保证没有错误。

2.请描述静态测试和动态测试的区别。

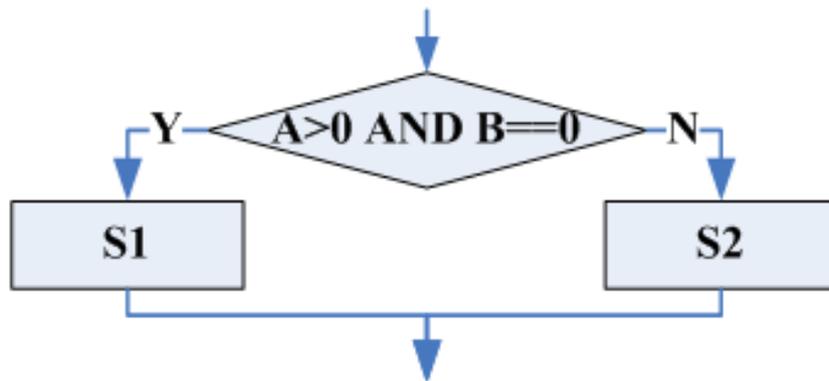
是否运行程序。静态测试是不运行被测试程序，对代码通过检查、阅读进行分析。动态测试是真正运行被测程序，在执行过程中，通过输入有效的测试用例对其输入与输出的对应关系进行分析。

3. 单个组件经过代码审查和测试，其有效性已经得到了全面验证，请解释为什么仍然需要进行集成测试。

单个组件正常工作并不意味着所有组件集成在一起可以正常工作，因为组件相互连接时接口会引起许多新问题，集成测试正是将通过单元测试的各个组件组装在一起进行综合测试，以便发现与接口有关的各种错误。

五、设计题：

1. 如图显示某程序的逻辑结构。试为它设计足够的测试用例，分别实现对程序的判定覆盖、条件覆盖和条件组合覆盖。



覆盖种类	需满足的条件	测试数据	期望结果
判定覆盖	A > 1, B = 0	A = 2, B = 0	执行 S1
	A > 1, B ≠ 0 或 A ≤ 1, B = 0 或 A ≤ 1, B ≠ 0	A = 2, B = 1 或 A = 1, B = 0 或 A = 1, B = 1	执行 S2
	以下四种情况各出现一次		

条件覆盖	A>1	B=0	A=2, B=0	执行 S1
	A≤1	B≠0	A=1, B=1	执行 S2
条件组合覆盖	A>1, B=0		A=2, B=0	执行 S1
	A>1, B≠0		A=2, B=1	执行 S2
	A≤1, B=0		A=1, B=0	执行 S2
	A≤1, B≠0		A=1, B=1	执行 S2

2.某城市电话号码由三部分组成。它们的名称和内容分别是：地区码：空白或三位数字；前缀：非‘0’或‘1’的三位数字；后缀：4位数字。假定被测程序能接受一切符合上述规定的电话号码(如：(635)805-9321)，拒绝所有不符合规定的电话号码。根据该程序的规格说明，作等价类的划分，并设计测试方案。

输入条件	有效等价类	无效等价类
地区码	1.空白；	5. 有非数字字符；6.少于三位数字；

	2.三位数；	7.多于三位数字。
前 缀	3.从 200 到 999 之间的三位	8.有非数字字符；9.起始位为 ‘0’ ； 10.起始位为 ‘1’ ； 11.少于三位数字； 12.多于三位数字。
后 缀	4.四位数字。	13.有非数字字符；14.少于四位数字； 15.多于四位数字。

方 案	内容			输 入	预期 输出
	地区码	前缀	后缀		
1	空白	200 ~ 999 之 间的三位数字	四位数字	()276-2345	有效
2	三位数字		四位数字	(635)805-9321	有效
3	有非数字字符			(20A)723-4567	无效
4	少于三位数字			(33)234-5678	无效
5	多于三位数字			(5555)345-6789	无效
6		有非数字字符		(345)5A2-3456	无效
7		起始位为 ‘0’		(345)012-3456	无效
8		起始位为 ‘1’		(345)132-3456	无效
9		少于三位数字		(345) 92-3456	无效
10		多于三位数字		(345)4562-3456	无效

11			有非数字字符	(345)342-3A56	无效
12			少于四位数	(345)342- 356	无效
13			多于四位数	(345)562-34567	无效

3.在 flight4A 或 flight4B 中自己设计测试用例 1，完成任意一个 GUI 检查点的设置，并要求设置多个检查点的内容。设计测试用例 2，这是一个数据驱动的测试用例，实现过程为：首先在 flight4A 中新建一个预约，输入航班的日期、起飞地、目的地、姓名和机舱类型，并完成插入预约的操作；再将此基本测试转换为数据驱动测试，在数据表中为姓名和机舱类型设计不同的 3 个数据值，执行测试并查看测试结果。设计测试用例 3：将上面设计的两个测试，建立一个批处理测试。

黑盒测试：

例如，有一个处理单价为 5 角钱的饮料自动售货机，软件测试用例的设计规格说明如下：若投入 5 角钱或 1 元钱的硬币，押下【橙汁】或【啤酒】的按钮，则相应的饮料就送出来。若售货机没有零钱找，则一个显示【零钱找完】的红灯亮，这时在投入 1 元硬币

并押下按钮后，饮料不送出来而且 1 元硬币也退出来；若有零钱找，则显示【零钱找完】的红灯灭，在送出饮料的同时退还 5 角硬币。”

1) 分析这一段说明，列出原因和结果：

1. 售货机有零钱找

2. 投入 1 元硬币

3. 投入 5 角硬币

4. 押下橙汁按钮

5. 押下啤酒按钮

结果： 21. 售货机【零钱找完】灯亮

22. 退还 1 元硬币

23. 退还 5 角硬币

24. 送出橙汁饮料

25. 送出啤酒饮料

2) 画出因果图所有原因结点列在左，所有结果结点列在右。建立中间结点，表示中间

状态：

11. 投入 1 元硬币且押下饮料按钮

12. 押下【橙汁】或【啤酒】的按钮

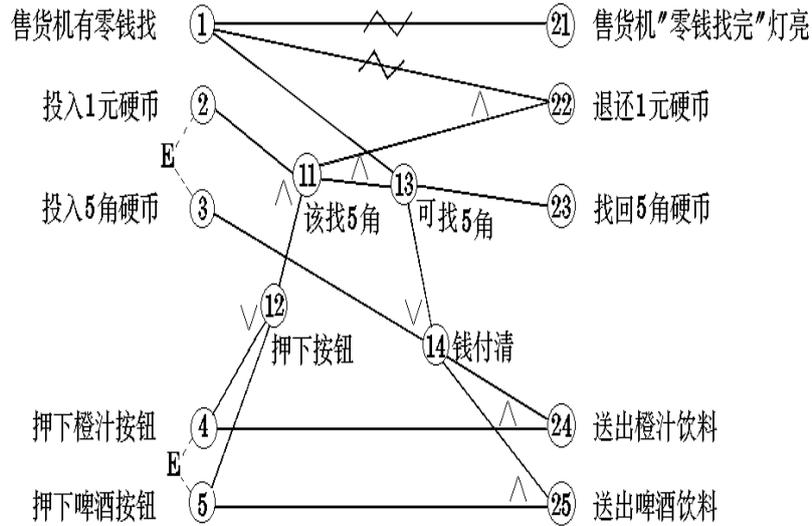
13. 应当找 5 角零钱并且售货机有零钱找

14. 钱已付清

3) 由于 2 与 3，4 与 5 不能同时发生，分别加上约束条件 E。

4) 因果图转换成判定表。

5) 在判定表中选择测试用例。



序号	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1	2			
条件	①	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	②	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	
	③	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	
	④	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	0	0	1	1	0	0	0	
	⑤	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
中间结果	⑪					1	1	0		0	0	0		0	0	0									1	1	0		0	0	0		0	0	0
	⑫					1	1	0		1	1	0		1	1	0									1	1	0		1	1	0		1	1	0
	⑬					1	1	0		0	0	0		0	0	0									0	0	0		0	0	0		0	0	0
	⑭					1	1	0		1	1	1		0	0	0									0	0	0		1	1	1		0	0	0
结果	⑳					0	0	0		0	0	0		0	0	0									1	1	1		1	1	1		1	1	1
	㉑					0	0	0		0	0	0		0	0	0									1	1	0		0	0	0		0	0	0
	㉒					1	1	0		0	0	0		0	0	0									0	0	0		0	0	0		0	0	0
	㉓					1	0	0		1	0	0		0	0	0									0	0	0		1	0	0		0	0	0
	㉔					0	1	0		0	1	0		0	0	0									0	0	0		0	1	0		0	0	0
测试用例						Y	Y	Y		Y	Y	Y		Y	Y									Y	Y	Y		Y	Y	Y		Y	Y		