

第一章 软件测试概要

1. 什么是软件缺陷？ P1

软件中所有质量问题都称为软件缺陷

2. 软件缺陷的产生原因有哪些？ P3

软件需求分析不够全面 准确是导致软件缺陷的最主要原因,需求分析,软件设计和编码的过程中的失误也会导致软件缺陷.

3. 软件测试的目标是什么？ P4

为了检验软件系统是否满足用户的需求.

4. 测试和调试有何不同？ P4

测试时一个有计划可重复的过程,目的是为了发现软件的潜在错误和缺陷;调试是一个随机的不可重复的过程,目的是为了找出错误的原因和位置,并修复.调试一般在测试之后,但调试后又要测试,所以两者常交叉进行.

5. 为什么对软件进行穷举测试是不可能的？ P4

由于需求规格说明的复杂性和程序逻辑的复杂性.(测试程序中的输入条件的取值;测试程序中所有输入条件的取值的组合;测试程序中的所有路径;测试中所潜在的错误和缺陷等)

6. 为什么说测试自动化不是万能的？ P5

软件测试自动化可以提高测试的效率,但是成本较高,需要自动化测试工具,还需要测试人员编写的测试脚本等.

7. 什么是第三方测试？ P9

第三方测试机构是一个具有中介性质的服务机构,它能够通过自身

专业化的测试技术和测试团队为客户和软件开发商提供有价值的服务.(能够独立,客观公正的评价被测试软件.)

第 2 章 软件测试基础

1 . 什么是软件测试 ? P10

使用人工或自动的手段来运行或测定某个软件系统的过程,其目的在于检验它是否满足规定的需求或弄清预期结果和实际结果之间的差别.(IEEE)

2 . 软件测试的对象是什么 ? P11

程序,需求分析和设计工作,文档

3 . 什么是验证 ? 什么是确认 ? P11

验证即检验软件是否实现了预先定义的功能和其他特性,即判断软件开发每一阶段的活动是否成功的完成,各开发阶段形成的软件配置是否保持一致.

确认即有效性确认.目的在于判断交付使用的软件可否追溯到用户的需求.作用是检验软件产品功能及其他特性的有效性.

4 . 按开发阶段 , 软件测试可分成哪几类 ? P12

元测试,集成测试,确认测试,系统测试和验收测试.

5 . α 测试和 β 测试的含义分别是什么 ? 两者有何区别和联系 ? P13

α 测试是开发公司内部人员模拟各类用户所进行的软件产品测试

β 测试是用户进行的测试,但是不同于验收测试即决定是否接受软件并不是 β 测试的目的. β 测试的目的帮助开发方在产品发布前做最后的改进.

两者的测试人员不同目的不同.联系: α 测试调整的软件产品是 β 测试

5 . 白盒测试、黑盒测试、灰盒测试三者有何区别 ? P13

白盒测试 , 又称结构测试或者逻辑驱动测试 , 其代码对测试人员可见 ; 黑盒测试 , 又称功能测试或者数据驱动测试 , 其代码对测试人员不可见。灰盒测试是介于白盒测试和黑盒测试之间的测试。

6 . 什么是回归测试、功能测试、性能测试 ? P14

回归测试 : 为了验证对软件引入的修正的正确性及其影响而进行的测试。功能测试 : 也称行为测试 , 它根据产品的特征 , 操作描述和用户方案 , 测试一个产品的特性和可操作性以确定它们是否满足设计需求。性能测试 : 评价一个产品或组件与其性能要求是否符合的测试 , 包括负载 (压力) 测试 , 强度测试 , 容量测试 , 疲劳测试等。

7 . V 模型的局限性是什么 ? P17

使修复错误的代价大大增加 , 不利于提高软件开发和测试的效率。

8 . W 模型的优点和局限性是什么 ? P17

优点 : 有利于尽早地全面地进行测试 , 发现软件中存在的问题 , 也有利于全过程的测试。

缺点 : 无法支持迭代 , 自发性及变更调整。

9 . H 模型倡导的理念是什么 ? P19

将测试活动完全独立出来 , 形成一个完整独立的流程 , 以将测试准备活动和测试执行活动清晰地体现出来。

10 . 测试过程管理理念有哪些 ? P20

尽早测试 , 全面测试 , 全过程测试 , 独立的迭代的测试。

11．测试驱动开发的含义是什么？它的基本步骤是怎样的？ P21

TDD 的基本思路是通过测试推动整个过程开发的进行。也就是在明确要实现某个功能的开发后，首先思考如何对这个功能进行测试，并完成测试代码的编写；接着编写相关代码满足这些测试用例；然后循环添加其他功能，直到完成全部功能的开发。

基本步骤：明确当前要完成的功能点；快速完成针对此功能的测试用例的编写，编译测试代码，结果为通不过；编写对应的功能代码；测试代码编译通过；对代码进行重构，并保证全部测试代码运行通过；循环完成所有功能的开发。

12．测试驱动开发的原则是什么？ P23

测试隔离；专注于当前工作；防止过度的设计；及时补充测试列表；先写断言；及时重构。

13．软件测试的原则是什么？ P25

应尽早、全面、全过程、独立地开展测试活动；所有的测试标准应建立在满足用户需求的基础上，软件中的最严重错误是那些导致用户需求无法满足的错误。程序员应避免检查自己的错误；设计测试用例时，应该考虑合法的输入和不合法的输入，以及各种边界条件，特殊情况下要制造极端状态和意外状态；要充分注意测试错误的集中发现现象；对于测试中发现的错误要有一个确认的过程，严重的错误可以召开评审会进行分析讨论；制定严格的测试计划并执行之，杜绝测试的随意性；充分重视回归测试；妥善保管一切测试文档。

14．软件测试文档有哪些？ P26

测试计划，测试设计，测试用例，测试规程，测试事件报告，测试总结报告等。

第 3 章 黑盒测试方法

1 . 什么是测试用例？设计测试用例时应遵循哪些基本准则？ P36

测试用例是为特定目标开发的一组测试输入,执行条件和预期结果,其目的是测试程序中的某路经,核实程序或软件能否完成某个特定的功能需求.

基本准则:测试用例的代表性;测试用例的非重复性;测试结果的可判定性;测试结果的可再现性.

2 . 黑盒测试的目的是什么？ P38

发现以下错误:1.是否有不正确或遗漏了的功能;2.在接口上输入能否正确地接受;能否输出正确的结果;3 是否有数据结构错误或者外部信息访问错误;4.性能上是否能够满足要求;5.是否有初始化或终止性错误.

3 . 黑盒测试的优点和局限性是什么？ P38

优点:不考虑程序或软件的具体实现,若程序或软件内部发生了变化,原先的测试用例依然可用;用例的设计可以与软件的实现同时进行,加快软件测试与开发的速度.

局限性:黑盒测试只是从界面上进行测试,难以查找出错误的具体原因和位置;黑盒测试的唯一依据是软件的需求规格说明书,无法发现需求规格说明本身存在的问题.

4 . 典型的黑盒测试方法有哪些？ P39

等价类划分法,边界值分析法,错误推测法,因果图法,判定表法,正交试验法,场景法,功能图法等

5. 等价类划分法中,划分等价类的原则有哪些? P39

1.在规定了输入数据的取值范围的情况下,可以确定一个有效等价类和两个无效等价类.2.在规定了输入条件必须如何的情况下,可以确定一个有效等价类和一个无效等价类.3.在输入数据时一个布尔量的情况下,可以确定一个有效等价类和一个无效等价类.4.在规定了输入数据由 n 个值构成,并要求对每一个值进行测试时,可以确定 n 个有效等价类和一个无效等价类.5.在规定了输入数据必须遵守的规定下,可以规定一个有效等价类和若干个无效等价类.6.在已划分的等价类中,若各元素在程序中的处理方式不同,则应将此等价类进一步划分为若干个等价类.

6. 等价类划分法中,如何根据等价类来编写测试用例? P40

1.为每一个等价类规定唯一的编号,2.设计一个测试用例,使其尽可能多得覆盖尚未被覆盖的有效等价类,如此重复直至所有有效等价类均被测试用例所覆盖; 3.设计一个测试用例,使其覆盖一个尚未被覆盖的无效等价类.

7. 用边界值分析法设计测试用例,应遵循哪些原则? P42

1.若输入条件规定了范围,应针对刚达到此范围的值,以及刚超越此范围边界的值设计测试用例.2.若输出条件规定了值的个数,针对最大个数最小个数,比最大个数大 1 比最小个数小 1 等几种值设计测试用例.3.若输出条件规定了值的范围,应针对刚达到此值边界范围的值,

以及刚超越此范围边界的值设计测试用例.4.若输出条件规定了值的个数,针对最大个数最小个数,比最大个数大 1 比最小个数小 1 等几种值设计测试用例.5.若程序的规格说明中给出了输入域或者输出域是有序结合,应针对集合的第一个元素和最后一个元素设计测试用例.6.若程序中使用了内部数据结构,应针对该内部数据结构边界上的值设计测试用例.7.进一步分析规格说明,找出其他可能的边界情况,针对它们设计测试用例.

8 . 什么是错误推测法 ? P42

是根据测试人员的经验和直觉推测程序中可能存在的错误,有针对性地设计测试用例,可以验证事先推测是否正确.

9 . 如何利用因果图来设计测试用例 ? P45

1.分析程序的规格说明中哪些是原因,哪些是结果;2.分析程序的规格说明中的语义,确定原因与原因,原因与结果之间的关系,画出因果图;3.由于语法或者环境的限制,一些原因与原因之间,原因与结果之间的组合不能出现.对于特殊情况,用一些记号标明约束和限制条件;4.将因果图转化为判定表;根据判定表的每一列设计用例.

10 . 简述用正交试验设计方法设计测试用例的基本步骤。 P54

1.确定因子,即确定软件的运行结果有影响的因素.2.确定因子的取值范围;3.确定每个因子的水平.4.选择合适的正交表.5.设计测试用例表.

11 . 简述黑盒测试方法的综合使用策略。 P61

1.首先进行等价类的划分,包括输入条件和输出条件的等价划分,2.在任何情况下都必须使用边界值分析法;3.测试人员可以根据经验用错

误推测法追加一些测试用例;4.如果程序的功能说明中含有输入条件的组合情况,则一开始就选用因果图法和判定表法;5. 对于参数配置类软件,应用正交试验法选择较少的组合方式以达到最佳效果,并减少测试用例的数目;6.对于业务流清稀的系统可是使用场景法;7.当程序的功能较复杂,存在大量组合的情况下,可以考虑功能图法.

第 4 章 白盒测试方法

1. 白盒测试的含义是什么?有何优缺点? P64

白盒测试,又称结构测试或者逻辑驱动测试或是基于程序本身的测试.

优点:深入到程序内部进行测试,更易于定位错误的原因和具体位置。

缺陷:不能 查出程序中的设计缺陷;不能查出程序是否遗漏了功能或路径;可能发现不了一些与数据相关的错误。

2. 白盒测试分为哪几类? P65

分为静态测试和动态测试两大类。

3. 典型的白盒测试方法有哪些? P66

代码检查法,静态结构分析法,代码质量度量法,(静态测试方法)

逻辑覆盖法,基本路径测试法,控制结构测试和程序插桩(动态测试方法)

4. 逻辑覆盖法有哪些覆盖标准? P67

语句覆盖,判定覆盖,条件覆盖,判定-条件覆盖,条件组合覆盖和修正条件判定覆盖。

5. 基本路径测试的步骤是什么? P70

在程序的流图基础上,确定程序的环路复杂性,导出基本路径的集合,进而在其基础上设计测试用例。

6. 如何计算程序的环路复杂性? P71

$V(G)=E-N+2$ E 为流图中的边数, N 为流图中的节点数

$V(G)=P+1$ P 为流图中的谓词节点数

7. 循环结构有哪些类型? P76

简单循环, 嵌套循环, 并列循环和非结构循环

8. 如何对嵌套循环进行测试? P77

从最内层循环开始测, 此时所有外层循环的次数都取小值, 内层循环按照简单循环策略进行测试; 由内向外测试; 退到外一层的测试, 同上一步, 继续向外扩展测试, 直至测完。

9. 如何进行程序插桩? P78

第 5 章 单元测试

1. 什么是单元测试? P81

单元测试就是对软件基本组成单元的测试。

2. 单元测试的目的是什么? P82

1.验证代码能否达到详细设计的预期要求;2.发现代码中不符合编码规范的地方;3.准确定位发现的错误,以便排出错误。

3. 单元测试的测试方法有哪些? P82

白盒测试和黑盒测试,其中以使用白盒测试为主。

4. 什么驱动模块、桩模块? P83

驱动模块式用来代替被测试单元的上层模块的.驱动模块能接收测试数据,调用被测单元,也就是将数据传递给被测单元,最后打印测试的执行结果.

桩模块,又称存根模块,它用来代替被测单元的子模块.其目的是模拟现实被测试单元的接口.

5 . 如何建立单元测试的环境？ P85

构造最小运行调度系统,即构造被测单元的驱动模块;模拟被测单元的接口,即构造被测单元调用的桩模块;模拟生成测试数据及状态,为被测单元准备动态环境.

6 . 单元测试的内容有哪些？ P86

单元接口,局部数据结构,独立路径,出错处理,边界条件.

7 . 简述单元测试用例设计步骤。 P88

1.为被测单元运行设计测试用例;2.为正向测试设计测试用例;3.为逆向测试设计测试用例;4.为满足特殊要求设计测试用例;5.为代码覆盖设计测试用例.

第 6 章 集成测试

1 . 什么是集成测试？ P104

集成测试,也称组装测试,联合测试.在单元测试的基础上,应根据概要设计的要求将各单元组装成子系统或者系统,在单元组装过程中,应对单元整体测试,发现并清除在单元连接过程中出现的问题,确保集成到一起的各单元能够共同完成预期的功能,并达到要求的性能.

2 . 程序的接口有哪些类型？

函数接口,消息接口,类接口,其他接口(全局变量,配置表,注册信息,中断等).

3 . 典型的集成测试策略有哪些 ? P107

一次性集成方式,自顶向下的集成方式,自底向上的集成方式,混合式集成模式,核心系统现行集成方式,高频集成方式,基于消息/事件/线程的集成方式,基于使用的集成方式.

4 . 简述自顶向下的集成方式和自底向上的集成方式的区别 P108

自顶向下的集成方式是根据软件的模块结构图,按照控制层次从高到低的顺序进行集成,也就是从最顶层模块向下逐步集成,并在集成的过程中进行测试,直至组装成符合要求的最终软件系统.多用于软件的控制结构有较大的风险,需要早测.和在极限编程中使用探索式开发风格.

自底向上的集成方式是根据软件的模块结构图,按照控制层次从低到高的顺序进行集成,也就是从最底层模块向上逐步集成,并在集成的过程中进行测试,直至组装成符合要求的最终软件系统.多用于软件的高层接口变化较频繁,可测试性不强,和软件的底层接口比较稳定.

5 . 关键模块有哪些特征 ? P113

1.能够完成需求规格说明中的关键功能;2.在软件模块结构中处于较高层次;3.较复杂,易出现错误;4.有明确的性能要求;5.被频繁使用.

第 7 章 确认测试、系统测试和验收测试

1 . 什么是确认测试 ? P115

确认测试是验证软件的功能性能及其他特性是否达到需求规格说明

书的要求的测试.

2 . 什么是系统测试? 系统测试有哪些主要类型? P116

系统测试是将已经确认测试的软件作为基于整个计算机系统的一个元素,与系统中其他所有元素结合在一起,在实际运行环境下或模拟系统运行的环境下,测试其与系统中其他元素能否实现正确的连接,以满足用户的所有需求.

功能测试,性能测试(压力测试,负载测试,疲劳测试,强度测试,容量测试),安全性测试,健壮性测试,可靠性测试,配置测试和兼容性测试,用户界面测试,文档测试,安装测试和卸载测试.

3 . 性能测试有哪些类型? P117

压力测试,负载测试,疲劳测试,强度测试,容量测试

4 . 什么是文档测试? 检查的内容有哪些? P123

是指对提交给用户的文档进行测试.测试的内容有:文档的内容是否能让不同级别的读者理解.文档的内容是否适合读者.内容主题是否合适.图表的准确度和精确度如何.样例和事例是否与软件功能一致.拼写个语法是否准确.文档是否与其它相关文档的内容一致.

5 . 基于 Web 的系统测试有哪些类型? P131

功能测试.(链接测试,表单测试,对数据校验测试,Cookies 测试,数据库测试,对应用程序特定功能需求的测试,设计语言的测试) 用户界面测试.(导航测试,图形测试,内容测试,表格测试,整体界面测试.) 兼容性测试和配置测试.(平台测试,浏览器测试,分辨率测试,Modem 连接速率测试,打印机测试,组合测试) 安全测试.(目录设置测试,SSL 测试,

登录测试,日志文件测试,脚本语言安全性测试) 接口测试.(服务器接口测试,外部接口测试,对异常处理的测试)

6 . 什么是验收测试? P140

验收测试检验时最终软件产品与用户预期的要求是否一致,决定软件是否可被用户接受.其着重考虑软件是否满足合同规定的所有功能性能和其他特性.

第 9 章 软件测试自动化

1 . 简述人工测试和自动化测试的优缺点。 P161

人工测试优点:及时发现错误并进行修正.

人工测试缺点:工作量大,时间花费大

自动化测试优点:提高软件的测试效率,对测试配置进行有效的管理,并使测试配置在整个测试生命周期得到复用,整个测试过程的自动化管理,对测试的整个过程进行规范化定义,避免过分依赖个人,减少错误的发生率,减少人工测试,从而降低成本.可以执行一些手工不可能进行的测试,如模拟真实情况,使测试变得更加有趣.

自动化测试缺点:发现软件缺陷的能力是有限的,而且运作成本较高.

2 . 列举出常用的测试工具及其用途? P168

测试管理工具:TestDirector(基于 web 集成的全球测试管理工具),TestManager(管理所有测试活动和工件的核心平台),CVS(版本控制开源工具),功能测试工具:WinRunner(功能测试及回归测试工具)和 QuickTest Pro(B/S 系统的功能测试工具) 性能测试工具 :LoadRunner(预测系统行为和性能的负载测试工

具)WebLoad(Web 压力测试工具)白盒测试工具: XUnit 系列开源框架

第 10 章 软件质量和质量保证

1. 什么是软件质量? P216

软件质量是系统,部件或过程满足顾客或者用户需要或期望的程度,以及系统,部件或过程满足规定需求的程度.

2. 简述 ISO 9126 质量模型。 P217

该模型定义了功能性、可靠性、易使用性、效率、可维护性和可移植性六个影响软件质量的质量特性,每个质量特性又可能通过若干个子特性来测量,每个子特性在评价时要进行定义并实施若干度量。该模型使得软件最大限度地满足用户明确的和潜在的需求,且从用户、开发人员、管理者等各类人员的角度全方位地考虑软件质量。

3. 简述软件度量的内容。 P221

软件度量贯穿于软件开发的整个生命周期,是软件开发过程中进行理解、预测、评估、控制和改善的重要载体。软件度量包括项目度量、产品度量和过程度量三个方面。

度量维度	侧重点	具体内容
项目度量	理解和控制当前项目的情况和状态	规模,成本,工作量,进度,生产力,风险,顾客满意度
产品质量	理解和控制当前产品的质量	产品的功能性,可靠性,易使用性,效率,可维护性,可移植性
过程度量	理解和控制当前的情况和状态,对过程进行改进和预测	能力成熟度,管理,生命周期,生产率,缺陷植入率

4 . 简述软件能力成熟度模型 CMM 的 5 个等级。 P224

初始级 (1) 可重复级 (2) 已定义级 (3) 已管理级 (4) 优化级 (5)

5 . 什么是 CMMI ? P229

CMMI Capability Maturity Model Integration 能力成熟度集成模型

6 . 软件质量保证 SQA 的目的是什么 ? 其工作内容包括 ? P233

是向管理者提供对软件过程进行全面监控的手段 , 包括评审和审计软件产品和活动 , 验证它们是否符合相应的规程和标准 , 同时给项目管理者提供这些评审和审计结果 , 以反映产品和过程质量 , 提高项目透明度。

工作内容 : 计划 , 审计/证实 , 处理不符合问题。

7 . 简述软件质量保证与测试的区别。 P235

其共同点在于它们的最终目标一致 , 都是提高软件的质量。

二者的区别在于 :

任务不同。SQA 是审计软件过程的质量 , 而 SQC 则是检验每个阶段产品的质量。

层次不同。SQA 是站在比 SQC 更高的层次上保证软件质量 ; 而 SQC 关注的内容是局部的 , 而 SQA 关注的则是整个软件过程。因此 , SQC 人员的工作是受 SQA 人员的监督的 , 即 SQA 可保证测试工作是按照定义好的流程执行的。

<PIXTEL_MMI_EBOOK_2005>3

</PIXTEL_MMI_EBOOK_2005>

