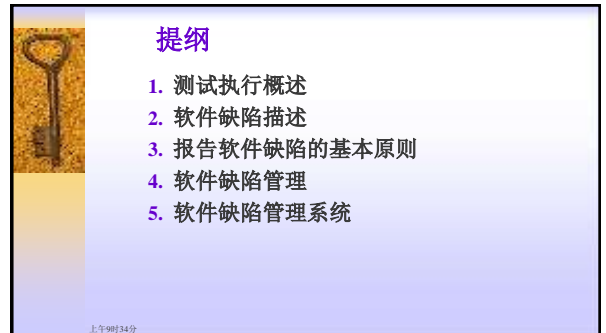
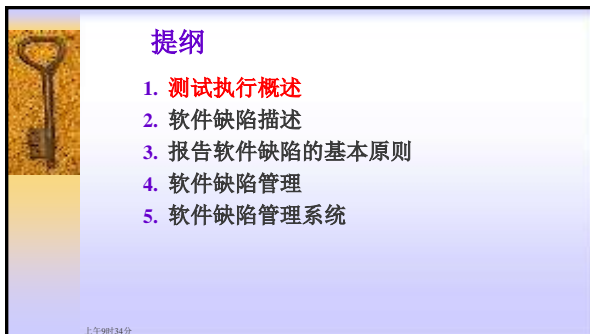




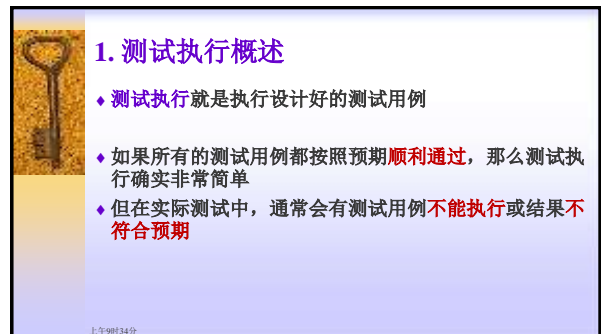
1



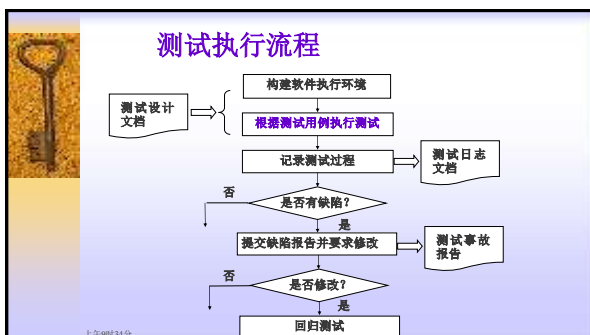
2



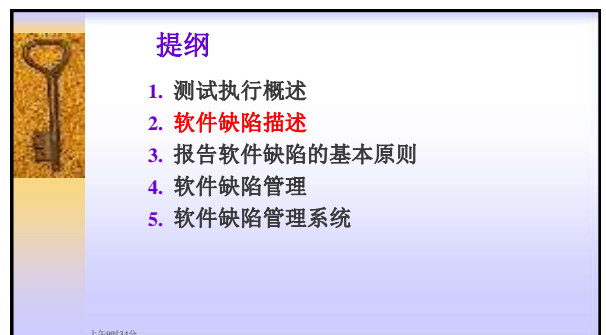
3



4



5



6

2. 软件缺陷描述

- 在测试中通常会产生与预期结果不一致的测试结果，这就产生了**软件缺陷**
- 对**软件缺陷进行管理**是软件测试执行的重要任务

2.1 有效的软件缺陷描述

- 短小**：只解释事实和演示、描述缺陷必需的细节
- 单一**：每一个报告只针对一个缺陷
- 明显和通用**：使用程序员容易看懂的、通用的简单步骤描述缺陷
- 再现**：缺陷报告必须展示其再现能力——按照预定步骤可以使软件缺陷再次出现。
有时很困难：偶然冲突，意外数据混乱等

2.2 软件缺陷的属性

序号	缺陷属性	描述
1	标识符	标识某个软件缺陷的唯一编号
2	描述	发生缺陷软件的版本、模块、环境以及触发的过程，产生的现象
3	缺陷类型	软件缺陷的分类，比如功能性缺陷，性能缺陷等
4	严重性	软件缺陷对于软件质量的破坏程度
5	优先级	缺陷被处理的紧急程度
6	状态	缺陷所处的生命周期
7	起源	缺陷引起的故障或事件第一次被检测到的阶段，包括：需求，构架，设计，编码，测试，发布等
8	再现性	缺陷是否可以再现

2.3 软件缺陷的严重性

- 严重性通常被分为4个级别

序号	严重级别	描述
1	致命缺陷	系统 主要功能完全丧失 ，比如系统崩溃，数据丢失，数据损毁
2	严重缺陷	系统 主要功能部分丧失 ，比如操作错误，结果错误，功能遗漏
3	一般缺陷	系统 次要功能部分丧失 ，错别字，系统布局不合理，罕见故障
4	微小缺陷	不影响系统的正常使用 ，可能是操作的不方便，改进建议等

2.4 软件缺陷的优先级

- 软件缺陷的优先级通常被分为4个级别

序号	优先级	描述
1	紧急	立即修复 ，阻止进一步测试
2	高	在 产品发布之前必须修复
3	中	如果时间允许则进行修复
4	低	可能会修复，但也可能放弃修复

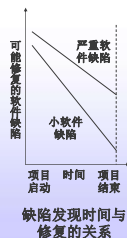
提纲

- 测试执行概述
- 软件缺陷描述
- 报告软件缺陷的基本原则**
- 软件缺陷管理
- 软件缺陷管理系统

3. 报告软件缺陷的基本原则

◆ 软件缺陷报告的基本原则是：

1. 尽快报告软件缺陷，修复软件缺陷的风险随着时间的推移大大增加
2. 有效描述软件缺陷



上午08时34分

13

3. 报告软件缺陷的基本原则

3. 在报告软件缺陷时不作评价。缺陷报告应该针对产品，而不能针对程序员，只陈述事实
4. 跟踪软件缺陷，测试员的责任是保证缺陷被正确的报告，并且得到应有的重视；良好的测试员发现并记录软件缺陷，优秀测试员发现并记录软件缺陷后，继续跟踪其修复的全过程

上午08时34分

14

提纲

1. 测试执行概述
2. 软件缺陷描述
3. 报告软件缺陷的基本原则
4. 软件缺陷管理
5. 软件缺陷管理系统

上午08时34分

15

4. 软件缺陷管理

- ◆ 为了保证能够修复所有记录的缺陷，测试组织内需要建立一套完整的缺陷管理过程和规则
- ◆ 要对软件缺陷进行有效管理，首先要理解软件缺陷生命周期的概念，根据软件缺陷的生命周期对其进行管理

上午08时34分

16

4.1 软件缺陷生命周期

- ◆ 任何事物都有自己的发生、发展、死亡的过程，这就是生命周期，软件缺陷也一样，也有自己的生命周期
- ◆ 管理软件缺陷的目的是为了消除软件缺陷，要消除软件缺陷，必须要知道软件缺陷的形态变迁，即其产生、存在直到消亡的整个过程，这就是软件缺陷生命周期
- ◆ 软件缺陷生命周期描述的是软件缺陷状态的转换

上午08时34分

17

4.2 软件缺陷的各种状态

1. 打开状态：缺陷首先被测试员发现，登记下来后转交给程序员修复的状态
2. 解决状态：程序员修复了代码，软件回到测试员手中的状态
3. 关闭状态：测试员重新对软件进行测试，确认缺陷已经修复的状态
4. 审查状态：项目管理员确定缺陷是否应该修复的状态，这是一个附加状态
5. 推迟状态：审查可能认定缺陷应该在将来进行修复的状态

上午08时34分

18

4.3 最简单的软件缺陷生命周期

- 缺陷被发现后，提交给开发人员，缺陷处于“打开”状态；
- 开发人员进行缺陷修复，此时缺陷处于“解决”状态；
- 缺陷修复后，再次进行测试，发现此缺陷已经不再存在，则为“关闭”状态。

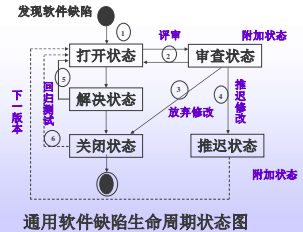


上午08时34分

19

4.4 通用软件缺陷生命周期状态图

Ron. Patton的“软件测试”教材中，描述了缺陷状态的各种情况，及其之间可能的6条不同状态转换路径



上午08时34分

20

一些缺陷管理工具中的缺陷状态

- 在BugFree中，Bug的状态依次为：Active、Resolved、Closed
- 在Bugzilla中，使用Status和Resolution记录Bug的状态：
 - 状态（Status）分类
 - 待确认的（Unconfirmed）/新提交的（New）/已分配的（Assigned）/问题未解决的（Reopened）/待返测的（Resolved）/待归档的（Verified）/已归档的（Closed）
 - 处理（Resolution）分类
 - 已修改（Fixed）/不是Bug（Invalid）/无法修改（Won't fix）/以后版本解决（Later）/保留（Remind）/重复（Duplicate）/无法重现（Workforme）

上午08时34分

21

提纲

1. 测试执行概述
2. 软件缺陷描述
3. 报告软件缺陷的基本原则
4. 软件缺陷管理
5. 软件缺陷管理系统

上午08时34分

22

5. 软件缺陷管理系统

- 在测试过程中发现的缺陷，会记录在缺陷记录报告中
- 需要管理和跟踪每一个软件缺陷报告，确保软件缺陷得以修复
- 软件缺陷管理系统就用来组织和跟踪软件缺陷报告

上午08时34分

23

5.1 软件缺陷登记的要素

1. 标识符——缺陷报告的唯一ID，用于定位和引用
2. 简述——用简明扼要的语言陈述缺陷
3. 事实描述——详细描述缺陷的相关信息和出现步骤
4. 影响——表明缺陷的严重性和优先级
5. 条件——产生缺陷的软硬件条件

上午08时34分

24

5.2 手工软件缺陷报告和跟踪

缺陷报告名称	课程网站系统测试缺陷			编号	O-1
软件名称	SCUCC	编译号	001	版本号	1.1
测试人员	张三	日期	2020/2/1	指定处理人	李四
硬件平台	IBM Server		操作系统 windows server 2000		
严重程度	1	优先级	高		
缺陷概述	教师不能完成学生信息增加				
详述描述: 1. 2. 3.					
处理结果	已修改		处理日期	2020/2/5	处理人 李四
修改记录	1. 2. 3.				
返测人	张三	返测版本	1.2	返测日期	2020/2/8
返测记录	1. 2. 3.				

上午09时34分

25

5.3 自动软件缺陷报告和跟踪

- ◆ 采用专用的**软件缺陷管理系统**可以有有效的跟踪各个软件缺陷的生命周期状态，便于更好的查看和修改软件缺陷。

- Mantis
- BugZilla
- BugFree
-

上午09时34分

26

5.4 BugFree中的缺陷记录



上午09时34分

27

BugFree测试结果统计



28

5.5 缺陷管理系统Bugzilla <https://www.bugzilla.org/>



29

The End
Any Question?



30