



考点	重要程度	占分	题型
6.1实现	★★★	2~4	选择、填空
6.2软件测试基础	★★★★	4~8	填空、选择、简答
6.3-6.5测试技术	★★★	4~6	填空、选择
6.6白盒测试	★★★★	4~8	选择、填空、大图
6.7黑盒测试	★★★★	4~8	选择、填空、大题
6.8调试	★★★	2~4	选择、填空
6.9软件可靠性	★★★	2~4	选择、填空

9.1 估算软件规模

软件项目管理:所谓**管理**就是通过计划、组织和控制等一系列活动,合理地配置和使用各种资源,以达到既定目标地过程。

估算软件规模的**方法**:

代码行技术:根据以往开发经验估计实现一个功能需要的源程序行数。

优点:代码行数计算容易。

缺点:

- ① 程序只是一部分用来作为软件整个规模不太合理。
- ② 不同语言实现同一软件代码行数不一致。
- ③ 不适用于非过程语言。

功能点技术:以功能点为单位度量软件规模。

9.2 工作量估算

工作量单位通常是人月(pm)。

工作量估算模型:

静态单变量

动态多变量



视频讲解更清晰
仅3小时

6.2软件测试基础

软件测试的**方法**:

黑盒测试法:

将软件看作一个黑盒子, 不考虑其内部结构和处理过程, 只按照规格说明书的规定, 测试软件是否能够正确接收输入数据, 产生正确的输出数据。即测试程序是否正确的实现了其功能。又称为“**功能测试**”。

白盒测试法:

完全知道程序的内部结构和处理算法, 因此可以将程序看作一个透明的白盒子, 根据程序内部的逻辑结构测试程序内部的主要执行通路是否能够按照预定的要求正确工作。又称“**结构测试**”。

6.2软件测试基础

软件测试的**步骤**:

1. **单元测试** (模块测试) : 将每个模块作为一个单独的实体进行测试。发现的错误编码和详细设计阶段的错误
2. **子系统测试**: 将模块集成为一个子系统进行测试。着重测试模块的接口。
3. **系统测试**: 将子系统组装为一个完整的系统进行测试。子系统测试和系统测试总称为“集成测试”。
4. **验收测试** (确认测试) : 在用户的参与下, 往往使用实际的数据进行的测试。发现需求说明中的错误
5. **平行运行**: 同时运行新开发出来的系统和将被它取代的旧系统, 以便比较新旧两个系统的处理结果。

6.3单元测试

测试依据：详细设计文档

测试技术（设计测试用例的方法）：白盒测试技术

着重点：

- 1. 模块接口
- 2. 局部数据结构
- 3.重要的执行通路
- 4. 出错处理通路
- 5. 边界条件



视频讲解更清晰
仅3小时

6.4集成测试

目标：发现与接口有关的问题

实施者：独立的测试机构或第三方人员

集成方法：非渐增测试、渐增测试

自顶向下集成：从主控模块开始，
沿着程序的控制层次自顶向下移动，
逐步添加新的模块

自底向上集成：从最底层模块开始组装

自顶向下与自底向上相结合的方法：

上层模块使用自顶向下方法
下层模块采用自底向上方法

6.4集成测试

回归测试:

重新执行已经做过测试的某个子集，以保证程序的变化没有带来非预期的副作用。

6.5 确认测试

又称**验收测试**，目标是验证软件的有效性。

验证：为了保证软件正确的实现了某个特定要求而进行的一系列活动。

确认：为了保证软件确实满足了用户需求而进行的一系列活动。

- (1) **Alpha测试：**用户在开发者的场所，在开发者指导下进行。
- (2) **Beta测试：**，用户在用户场所进行，遇到问题报告给开发者，开发者进行修改。

6.6白盒测试

测试用例：测试输入数据和预期的输出结果。

测试方案：测试目的、测试用例的集合。

- **语句覆盖：**被测试程序中的每条语句至少执行一次。
- **判定覆盖：**使得被测试程序中每个判定表达式至少获得一次“真”值和“假”值
- **条件覆盖：**使得判定表达式中每个条件的各种可能的值至少出现一次。
- **判定/条件覆盖：**使得判定表达式中的每个条件的所有可能取值至少出现一次，并使每个判定表达式所有可能的结果也至少出现一次。
- **条件组合覆盖：**设计足够多的测试用例，使得每个判定表达式中条件的各种可能的值的组合都至少出现一次。
- **路径覆盖：**覆盖被测试程序中所有可能的路径。

6.6白盒测试

控制结构测试:

基本路径测试

条件测试

循环测试

6.7 黑盒测试

黑盒测试又称功能测试，着重测试软件的功能。

等价类划分法

- (1) 把程序的输入数据集合按输入条件划分为若干个等价类，每一个等价类相对于输入条件表示为一组**有效**或**无效**的输入。
- (2) 为每一等价类设计一个测试用例。

边界值分析法

输入等价类和输出等价类的边界就是应该着重测试的程序边界情况。选取的测试数据应该刚好**等于**、刚好**小于**、刚好**大于**边界值

6.8调试

调试（也称为纠错）是在测试发现错误之后排除错误的过程。

方法:

- 蛮干法
- 回溯法
- 原因排除法

结果:

找到了原因，然后改正和排除。

没找到原因，猜测一个原因，并设计附加测试用例来验证这个假设。

6.9软件可靠性

软件可靠性：程序在给定**时间间隔内**，按照规格说明书的规定成功运行的概率。

软件可用性：程序在给定**时间点**，按照规格说明书的规定成功运行的概率。



视频讲解更清晰
仅3小时

总结

- **软件测试基础**

- 目标
 - 准则
 - 测试方法
 - 测试步骤

- **单元测试**

- **集成测试**

- **确认测试**

- **白盒测试**

- 逻辑覆盖
 - 控制结构

总结

- **黑盒测试**

1. 等价类划分法
2. 边界值分析法
3. 错误推测法

- **调试**

调试途径