

软件质量保证与测试

Software Quality Assurance and Testing

第 2 章 软件测试策略

2.2 软件测试方法和技术概述



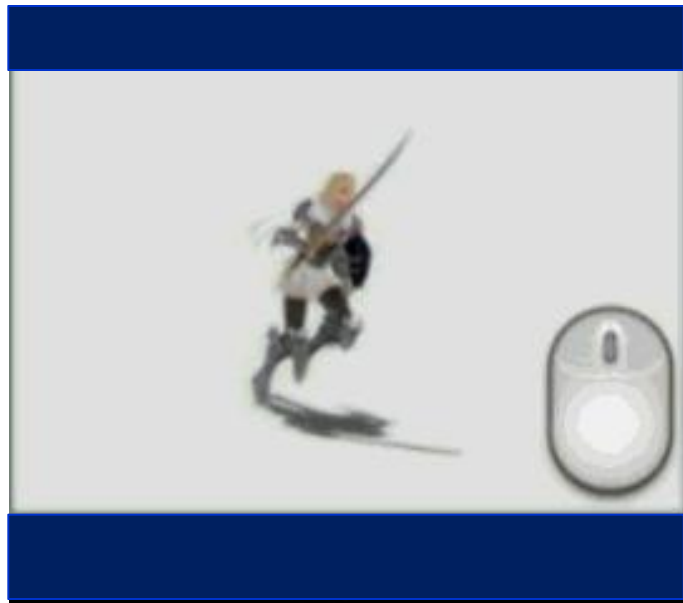
金陵科技学院

从是否需要执行程序的角度来区分

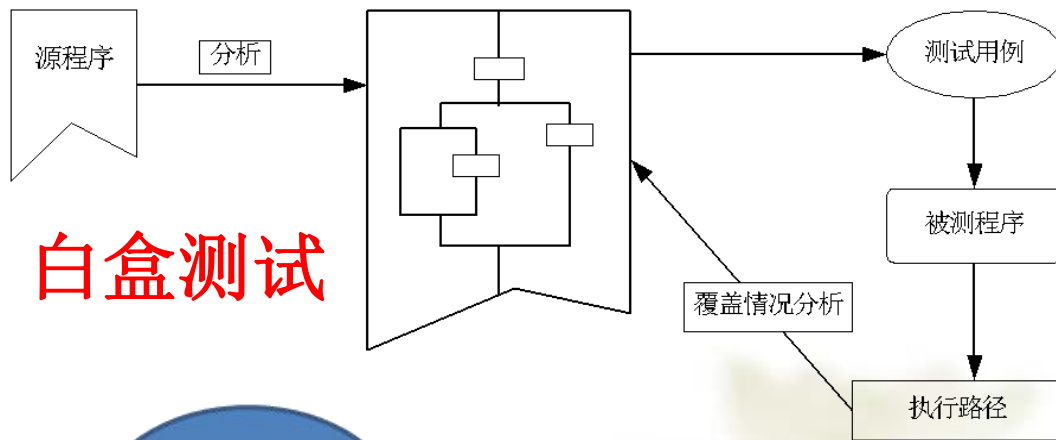
静态测试

```
#include<stdio.h>
max(float x,float y)
{
    float z;
    z=x>y?x:y;
    return(z);
}
main( )
{
    float a, b;
    int c,d;
    scanf("%f,%f",&a,&b);
    c=max(a,b);
    printf("Max is %d\n",c);
}
```

动态测试



从是否需要知道程序的内部结构来区分

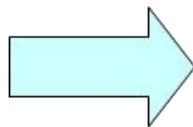


白盒测试

黑盒测试

黑盒

$X=2$



$y=4$

$Y=x*2$
 $Y=x+x$?

从测试执行者来区分



人工测试



自动化测试

静态测试

静态测试是指不需要执行被测程序，而是人工检查或者借助专用的软件测试工具来评审软件文档或程序，度量程序静态复杂度，检查软件是否符合编程标准，寻找程序的不足之处，降低错误出现的概率。

静态测试包括代码检查、静态结构分析、代码质量度量等。它可以由人工进行，也可以借助软件工具自动进行。

动态测试

动态测试是指通过运行被测程序，输入测试用例，检查运行结果与预期结果的差异，并分析运行效率、正确性和健壮性等性能。

这种方法由三部分组成：构造测试用例、执行程序并输入测试用例、记录并分析程序的输出结果。

测试用例

把用来测试程序的输入数据、相应的预期结果等称为测试用例。测试用例是对一项具体的测试任务的描述，完整的测试用例除了包括输入数据及预期结果之外，还应包括测试目标、测试环境、测试步骤、测试脚本等，并形成文档。

测试
用例

输入
数据

预期
结果

测试
环境

测试
步骤

.....

测试用例应该详细给出完成该项测试任务、执行该次测试过程所需的所有信息，因为设计测试用例和执行测试用例的可能不是同一个人。

静态测试 VS 动态测试

	优点	缺点
静态测试	发现缺陷早 降低返工成本 覆盖关键代码 发现缺陷概率高	非常耗费时间 需要知识和经验积累 技术能力要求高 准备工作多
动态测试	较为简单易行	发现缺陷迟 没有代码覆盖的针对性

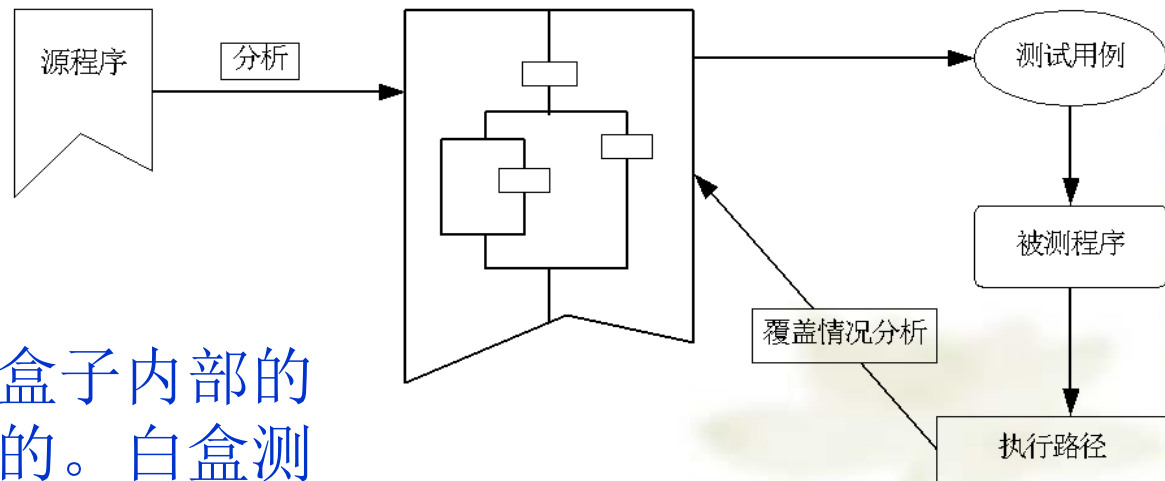
黑盒测试

又称功能测试、数据驱动测试或基于规格说明的测试。被测程序被当作一个黑盒，不考虑程序内部结构和特性，测试者只知道该程序输入和输出之间的关系或程序的功能，依靠能够反映这一关系和程序功能的需求规格说明书确定测试用例，然后执行程序，检查输出结果的正确性。



白盒测试

又称结构测试、逻辑驱动测试或基于程序的测试。它把程序看成是一个可以透视的盒子，能看清楚盒子内部的结构以及是如何运作的。白盒测试依赖于对程序内部结构的分析，针对特定条件设计测试用例，对软件的逻辑路经进行测试。白盒测试可以在程序的不同位置检验“程序的状态”以判定其实际情况是否和预期的状态相一致。



相互关系

黑盒测试 ——> 动态测试

白盒测试 { 静态测试
动态测试

动态测试 { 白盒测试
黑盒测试

静态测试 ——> 白盒测试

人工测试



人工测试

是指由测试人员手工执行测试活动，并记录测试结果，观察分析结果是否正确或者符合要求。当测试任务很重，需要执行非常多的测试数据时，人工测试是难以满足实际需要的。

自动化测试

自动化测试是指通过开发和使用软件分析和测试工具、测试脚本等来实现软件分析和测试过程的自动化，具有良好的可操作性、可重复性和高效率等特点。



自动化测试

软件测试的基本策略

1. 软件测试应当和软件开发同步进行。
2. 应对软件需求、软件设计等进行验证和确认。
3. 可按单元测试、集成测试、系统测试、验收测试分步实施。
4. 多种软件测试方法和技术应当合理的综合运用。
5. 应运用自动化测试技术，采用软件测试工具，提高软件测试的效率。
6. 软件测试项目可按照测试需求分析、测试计划、测试设计、测试开发、测试执行、测试总结这样的环节来组织实施。