系统测试（评审）

尽早、不断的进行测试

程序员避免测试自己设计的成勋

你要选择有效、合理的数据，也要选择无效不合理的数据

修改后英进行回归测试

尚未发现的错误与该成勋发现的错误数成正比

测试类型

动态测试

黑盒测试-功能测试-数据驱动

白盒测试-结构测试-逻辑驱动

灰盒测试-介于两者之间

静态测试

桌前检查-程序员自查

代码走查-程序员与架构师共同检查

代码审查-正规审查/轻量级审查（非正式）/结对编程

黑盒测试用例

等价类划分

边界值分析：把参数范围内的参数取边界值，测试极端参数的情况

错误推测：凭经验、直觉测试

因果图：由结果反推原因

白盒测试用例

基本路径测试

循环覆盖测试

逻辑覆盖测试

语句覆盖SC

判定覆盖DC

条件覆盖CC

路径覆盖

测试阶段

单元测试

集成测试

系统测试

确认测试

内部确认测试

Alpha测试

Beta测试

验收测试

非功能性测试

负载测试

强度测试

容量测试

恢复测试

安全性测试

压力测试

性能测试

可靠性测试

可用性测试

可维护性测试

安装测试

测试策略

自顶向下:

自底向下

混合式

面向队形的测试

算法层()

类层

模板层/类数层

系统层

测试自动化

优点

效率上：

成本上：

软件调试:调试是找出错误的代码和原因

软件调试方法

曼蛮力法

回溯法

原因排除法

软件调试与测试的区别

验证与确认

软件度量

软件的两种属性

外部属性：

内部属性：

软件质量

质量保证与质量控制（QC/QC）(关注过程/关注结果)（事中控制/事后检查）

软件过程改进（CMMI）

|  |  |
| --- | --- |
| 成熟度等级 | 过程域 |
|  |  |
| 已管理级 |  |
| 已定义级 |  |
| 定量管理级 |  |
| 优化级 |  |

配置管理

开发库

受控库

产品库

软件开发环境与工具

系统运行与评价

系统转换计划

新旧系统的转换策略

直接转换策略-成本低，风险高-小系统

并行转换策略-成本高，风险低-一般系统

分段转换策略-区域：试点-推广；大型系统：子系统-逐步转换

数据转换与迁移

旧数据库-抽取-转换-装载-新数据库（ETL）

三种方式：

系统切换前通过

系统运行和维护（硬件维护、软件维护、数据维护）

系统的可维护性：

系统评价指标：易分析性、易改变行、易测试性、稳定性

软件维护类型：

改正性维护（25%）：正确性，发现bug而修改

适应性维护（20%）：由于外部环境，被动惊醒软件的修改和升级

完善性维护（50%）：

预防性维护（5%）

系统审计

系统评价