

编 号 :
版 次 :
状 态 :
密 级 :
分 发 号 :
阶段标注 :

XXX

单元测试写作模板

编制:	日期:
审核:	日期:
会签:	日期:
标审:	日期:
批准:	日期:

XXXXXX信息科技有限公司

年 月

文件更改审批记录

序号	版本	*状态	作者	审核者	完成日期	修改内容

*状态： C - 创建 A - 增加 M - 修改 D - 删除

目 录

第 1 章 概述.....	1
1.1 单元测试的目的.....	1
1.2 测试的背景.....	1
1.3 单元测试所需文档.....	1
第 2 章 主要步骤.....	1
2.1 程序语法检查.....	1
2.2 程序逻辑检查.....	1
2.3 桩模块检查.....	1
第 3 章 单元测试项目.....	2
3.1 模块接口测试.....	2
3.2 局部数据结果测试.....	3
3.3 路经测试.....	3
3.4 边界条件测试.....	3
3.5 错误处理测试.....	3
3.6 代码书写规范测试.....	4
第 4 章 单元测试报告.....	4
4.1 单元测试报告的写作目的.....	4
4.2 单元测试报告内容.....	4
第 5 章 单元测试小结.....	6

第 1 章 概述

单元测试有成模块测试，是从内部结构来测试，可多个模块平行独立完成测试。单元测试主要来检验软件设计中最小的单位——模块。模块内聚程度高，每一个模块只能完成一种功能，因此模块测试的程序规模小，易检查出错误，并且善于确定错误的位置。

1.1 单元测试的目的

- 1) 对单元测试结果进行整理和汇总，形成正式的测试文档；
- 2) 检查出各模块内部的错误，为软件单元的评审验收提供依据；
- 3) 纳入软件产品配置管理库。

1.2 测试的背景

单元测试是以程序设计说明书和之前的测试数据(正常的和错误的)为指导，测试模块内重要的路径，以检查出错误。

1.3 单元测试所需文档

需要所有的测试数据，其中包括：正常数据，错误数据以及不同的数据参考。

第 2 章 主要步骤

2.1 程序语法检查

检查程序中语法错误

2.2 程序逻辑检查

- 1) 数据满足设计上要求的上下限及循环次数；
- 2) 数据满足程序中的各种检验要求的错误数据；
- 3) 数据可以适应于人工对程序的检查工作。

2.3 桩模块检查

桩模块用以代替被测模块调用的子模块。桩模块、桩模块、被测模块共同构成了一个“测试环境”，如图 2-1。

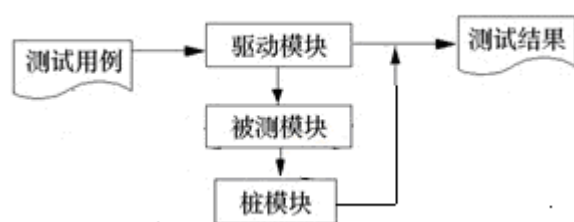


图 2-1 单元测试桩模块的环境

第 3 章 单元测试项目

3.1 模块接口测试

模块接口测试是单元测试的基础，当模块通过外部进行输入/输出操作时，必须数据能够正确流入、流出模块，模块才能完成它的功能。

1.调用其他模块时，查看所给的输入参数与模块的形式参数在个数、属性、顺序上是否匹配；

2.调用其他模块时，查看所给的实际参数的格式是否与被调用模块格式相同；

3.调用其他模块时，查看所给实际参数的属性是否与被调模块属性匹配；

4.调用定义函数时所用参数的个数、属性和次序是否正确；

5.输入的实际参数与形式参数的个数是否相同；

6.输入的实际参数与形式参数的属性是否匹配；

7.输入的实际参数与形式参数的范围是否一致；

8.是否修改了只做输入用的形式参数；

9.是否存在与当前入口点无关的参数引用；

10.是否修改了只读型参数；

11.各模块对全程变量的定义是否一致；

12.是否把某些约束作为参数传递；

13.输出给标准函数的参数在个数、属性、顺序上是否正确；

14.限制是否通过形式参数来传送；

15.文件属性是否正确；

16.OPEN/CLOSE 语句是否正确；

17.格式说明与输入输出语句是否匹配；

18.缓冲区大小与记录长度是否匹配；

19.文件使用前是否已经打开；

20.是否处理了输入/输出错误；

21.在结束文件处理时是否关闭了文件。

3.2 局部数据结果测试

局部数据结构是为了保证临时存储在模块内的数据在程序执行过程中完整及正确。模块错误的根源往往是局部数据结构，局部数据结构测试最常见的积累错误：

- 1.不适合或者不相容的类型说明；
- 2.变量无初值；
- 3.变量初始化或者缺省值有错；
- 4.不正确的变量名或不正确的截断；
- 5.出线上溢下溢和地址异常。

3.3 路经测试

对模块中重要的执行路径进行测试，路径错误主要是由错误的计算，不正确的比较或者不正常的控制流导致。路径错误主要因素是：

- 1.运算的优先次序不正确或误解了运算的优先次序；
- 2.运算方式错误，运算的对象彼此在类型上不相容；
- 3.算法错误；
- 4.初始化错误；
- 5.由于浮点数运算精度问题而造成的两数值比较不等；
- 6.关系表达始终不正确的变量和比较符号表示不正确；
- 7.不正确的多循环一次或少循环一次；
- 8.错误的或不可能的循环终止条件；
- 9.当遇到发散的迭代时不能种植的循环；
- 10.不适当的修改循环变量。

3.4 边界条件测试

边界条件测试要注意以下出错因素：

- 1.程序内有一个 n 次循环，这个 n 次循环应该是 $1\sim n$ ，而不是 $0\sim n$ ；
- 2.由小于、小于等于、等于、大于、大于等于、不等于确定的比较值出错；
- 3.出现上溢、下溢和地址异常问题。

3.5 错误处理测试

错误处理检查以下几点：

- 1.出错的描述不易理解；
- 2.出错的描述不足以对错误定位，也不能够正确的叙述出错的原因；

- 3.现实的错误与实际的错误不相符；
- 4.对错误条件的处理不正确；
- 5.异常处理不当。

3.6 代码书写规范测试

检查代码书写规范应注意以下几点：

- 1.模块设计程序框架流程图是否清晰；
- 2.代码书写以及对齐方式是否规范；
- 3.代码的注释；
- 4.参数类型、数据长度、指针、数组长度、大小；
- 5.输入/输出参数和结果。

代码审查如表 3-1 所示。

表 3-1 代码审查表

BUG	ID	审查人员	审查日期	问题描述	重审人员

第 4 章 单元测试报告

4.1 单元测试报告的写作目的

- 1) 对单元测试结果进行整理，写成正式的文档；
- 2) 为软件的单元评审验收提供依据。

4.2 单元测试报告内容

1) 软件单元描述

简单的描述被测试软件的单元项目名称，所属系统及此单元需要完成的功能。

2) 单元结构

画出本单元的结构及此结构的属性、方法、输入/输出等。

3) 单元控制

对本单元的控制及操作画出大概过程。

4) 测试过程

简要描述单元测试的过程。

5) 测试

主要包括的测试：

①代码审查

②测试用例

③测试特性（功能测试、性能测试、与两侧是、容错性测试）

④用例描述（对测试的功能进行简单描述）

⑤测试结论（说明测试的本单元是否通过）

⑥提交软件 BUG 清单，软件 BUG 清单如表 4-1 所示。

表 4-1 软件 BUG 清单

测试项	测试用例号	测试特性	用例描述	测试结论	BUG	ID	重审人员

⑦质量评估

对本单元的模块进行功能，性能，容错性，人机交互界面，可靠性，可维护性等进行评估

⑧填写单元测试表格

具体的单元测试表格可以参考下表 4-2 所示。

表 4-2 要填写的单元测试表格

开发项目名称		开发项目编号		第一责任人		
单元名称		责任人		单元所属子系统	开发周期	
代码测试						
代码测试内容	测试人员	测试结果			备注	
路经测试						
循环测试						
边界测试						
接口测试						
界面测试						
数据确认测试						

代码走查						
功能测试						
序号	功能名称	操作方法	结果	建议	测试人员	备注
测试结论				项目第一责任人		
责任人						
审核						
项目经理		测试经理		总工程师		

第 5 章 单元测试小结

- ★ 单元测试可将每一项都进行测试，保证它的正确性；
- ★ 单元测试具有回归性，它避免了代码出现回归，编写完成后可以随时随地快速运行测试；
- ★ 单元测试具有保证性。它能够保证代码质量和代码可维护性及可扩展性；
- ★ 测试之后，要对每一个程序写一份程序测试说明书，以备今后修改维护。