
“软件工程”作业四：

白盒测试

一 作业说明

学习掌握基于控制流的测试用例设计方法，设计测试用例，满足特定的覆盖率要求。

1. PyUnit

学生采用 PyUnit 编写单元测试用例，其提交的单元测试代码对教师给出的函数或者类的覆盖率将会作为此作业评分的依据。

2. Coverage 工具

系统使用 Coverage 工具来测试学生所上传的单元测试对教师提供模块的覆盖率。Coverage 工具的主页为 <http://nedbatchelder.com/code/coverage/>。在 Linux 和 OS X 下可以通过 `pip install coverage` 来安装。

Windows 下安装 Coverage 工具比较繁琐。如果你的单元测试代码没有问题，可以直接上传到系统中，由系统计量它的覆盖率。

二 评分标准

总评分=(语句覆盖率*50%+分支的完全覆盖率*50%+分支的部分覆盖率*25%)*90%+代码风格得分*10%。

1. 语句覆盖率

单元测试代码所执行的语句条数，占有效语句（不包括空行及一些语义结构语句）的比例，为语句覆盖率。

2. 分支覆盖率

分支覆盖有两种，完全覆盖及部分覆盖。所有出口都被执行过的分支，被称为完全覆盖；若是只执行过一部分出口的分支，则称为部分覆盖。比如图 1 中的 if 语句一共有两个出口：条件为真的出口，与条件为假的出口。

```
if a > b:
    return a
return b
```

图 1：由两个出口的分支

3. 代码风格得分

对于学生提交上来的文件，系统会对所有 `.py` 文件进行代码风格评测。在给定的一个 `.py` 文件中，只要有任何一条规范没有满足，则记该文件为不通过。

最终的得分为： $100 * \text{通过的文件数} / \text{总文件数}$ 。

4. 作业迟交评判

超过第一个截止日期提交的作业，会被乘以一个特定的惩罚因子（如图 2）。

截止日期	
这个作业将于 2天 之内截止，且最终得分将被乘以 0.5。	
截止日期	得分比例
2014年9月10日 下午11:59:59	1.0
2014年9月20日 下午11:59:59	0.5
2014年9月30日 下午11:59:59	0.0

图 2：不同截止日期的不同得分比例

三 提交方式

在 Railgun 在线评测系统上提交：<http://railgun.secoder.net/>。
最终代码同时提交到网络学堂。

- 1) 进入作业“白盒测试”。
- 2) 下载给定的程序代码
- 3) 针对题目要求编写 PyUnit 代码，满足语句覆盖和分支覆盖的要求。**注意，为了构造测试用例，你可能需要构造一些压缩包文件，与代码一起提交到系统中。**
- 4) 代码需遵照 PEP8 规范要求。
- 5) 上载修改后的代码
- 6) 查看作业检查结果
- 7) 可依据步骤 3-6，重复提交多次，按最好成绩计算。

四 提交日期

2014 年 12 月 13 日午夜 12:00