****

**软件测试技术第一次实验报告**



**学 院 软件学院**

**专 业 软件工程**

**年 级 大三**

**姓 名 张聪**

**2019年 3 月 12 日**

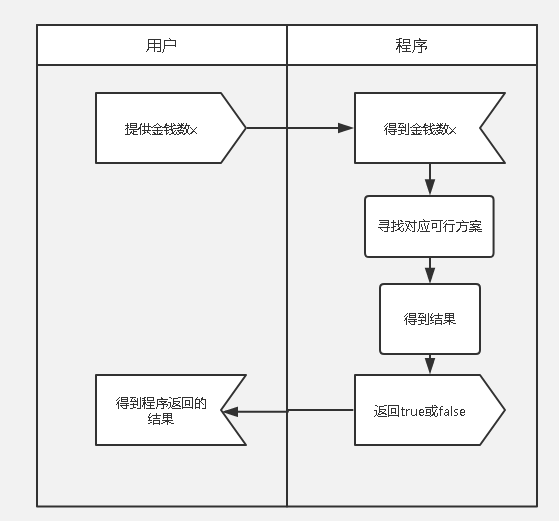
# 软件测试技术第一次实验报告

1. 需求分析（描述具体需求）

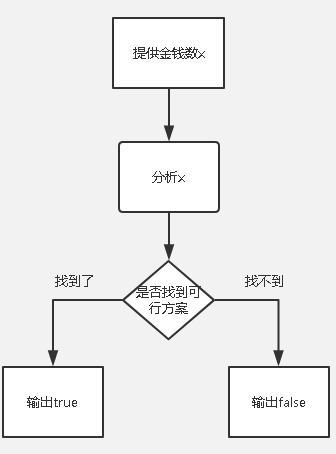
口袋里有一张50元，一张20元，两张5元的钞票和三个1元的硬币。给定一个特定的金额（X）元，看看能不能在不被找钱的情况下用口袋中的钱付出。

1. 概要设计（简单描述设计思路，配合UML图）

uml活动图



流程图



先进行数学分析，一共7张纸币，每张纸币有两种情况，使用或者不使用，因此共有2的7次方128种情况，情况数较少。对于此类问题，可利用二进制数的特点进行穷举，即每个情况对应一个7位二进制数，对应位上的数是0代表对应的纸币未被使用，对应位上的数是1代表对应纸币被使用。这样，一个大小为128的数组就可以存储所有情况信息，其中的元素编号既用于遍历、检索，又代表着钱币使用信息，数组元素的值为对应情况的钱币总金额。之后在128个数中对x进行匹配即可。

1. 详细设计（详细描述具体如何实现，附代码及说明）

**package** Lab1;

**public** **class** TakeMoney {

//1\*50,1\*20,2\*5,3\*1

**public** **int**[] notes={50,20,5,5,1,1,1};

**public** **int**[] state=**new** **int**[128];

TakeMoney(){

//二进制数字上的各位倒序表示各张纸币是否使用，1代表使用，0代表不使用

//例如1000100表示使用一张5元和一张1元

**for**(**int** i=0;i<128;i++){

**this**.state[i]=0;

**int** t=i;

**for**(**int** j=0;j<7;j++){

**int** n=t%2;

**this**.state[i]=**this**.state[i]+n\*notes[j];

t=t>>1;

}

}

}

**public** **int** getResult(**int** x){//匹配x,成功匹配返回对应状态值,即数组下标,否则返回-1

**for**(**int** i=0;i<128;i++){

**if**(state[i]==x)**return** i;

}

**return** -1;

}

**public** **boolean** getBool(**int** x){//通过调用getResult返回匹配成功与否的信息

**int** result=getResult(x);

**if**(result==-1)**return** **false**;

**else** **return** **true**;

}

}

代码如上，说明见代码中注释

1. 调试分析（在实验过程中遇到的问题以及如何解决）

问题：循环时数组越界

解决：重新改正循环条件

1. 测试结果（描述输入和输出）

输入输出如下，左为输入，右为输出

{-12,**false**},

{0,**true**},

{2,**true**},

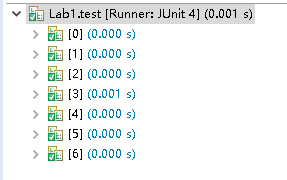
{40,**false**},

{65,**false**},

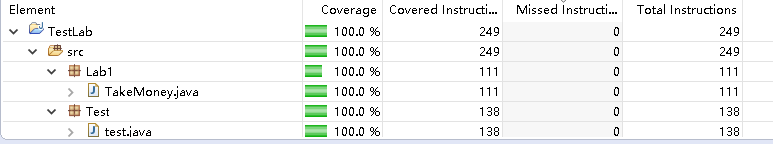
{73,**true**},

{85,**false**}

测试结果如图，全部成功：



代码覆盖率为100% ：



1. 总结

经过本次实验,我初步了解了如何使用junit, hamcrest 和 eclemma对java程序进行测试,并认识到这些工具对软件开发中的测试活动带来的便利.并对软件测试流程有了更深入的了解和掌握,且对课堂上老师讲的代码覆盖等知识点有了更深刻的了解,受益颇丰。

1. 附录

（一）安装junit, hamcrest 和 eclemma的简要描述

下载老师提供的安装包，右键项目，选择properties->java build path->add ecternal JARS 选择junit,和hamcrest的jar包

在eclipse的菜单栏点击 Help -> Eclipse Marketspace -> add Name:eclemma

    点击Name右侧的Local，选择下载好的本地eclemma-2.3.2文件夹（解压后），勾选读取到的文件，开始安装。

（二） url

https://github.com/likeAnEgg/SoftwareTest/tree/master/lab