



课程项目1

- 感谢唐浩、朱琪豪对开发包和评测平台的贡献
- 实现一个Java上的指针分析系统
- 排行榜给分：
 - 无法在测试程序上正常运行的不合格
 - 如：超时（3分钟），崩溃
 - 在测试程序上能输出结果，但结果不健壮(unsound)，1分
 - 结果健壮，根据精度分数在1-2之间
 - 得分为在所有测试用例上得分之和（数量由助教公布）
 - 代码提交作为评分参考
 - 提交一个自己编写的测试样例，包括代码和标准输出
- 最终给分：根据所有测试样例（含预定义的和大家提交的）上运行结果精度评分。
 - 公开的两个测试用例为60分，剩余测试用例为40分。
- 组队完成：
 - 2-3名同学一个小组
 - 组内贡献不均等的，请在提交的时候说明



程序样例

输入程序:

```
public static void main(String[] args) {  
    BenchmarkN.alloc(1); //标记分配点, 没有标记的默认编号为0  
    A a = new A();  
    BenchmarkN.alloc(2);  
    A b = new A();  
    BenchmarkN.alloc(3);  
    A c = new A();  
    if (args.length>1) a=b;  
    BenchmarkN.test(1, a); //标记测试点编号和被测变量  
    BenchmarkN.test(2, c);  
}
```

输出:

```
1: 1 2  
2: 3
```

每行一个测试点, 以测试点编号开头。
冒号后面是可能的分配点, 多个分配点以空格分割



开发平台

- Java上常见静态分析平台（自学）：
 - Soot（推荐）
 - WALA
 - Chord
- 部分平台已经自带指针分析，要求
 - 不能直接调用平台的指针分析模块
 - 可以使用平台提供的其他支撑，比如数据流分析框架，控制流图构建，Java语言化简等



时间节点和提交内容

- 组队报给助教（10月18日）
- 代码提交（11月12日）
 - Readme.pdf: A4两页以内，描述算法的主要设计思想，小组成员姓名、学号和分工
 - Code目录：项目源代码
 - analyzer.jar: 编译好的jar文件
- 现场报告（11月15日）
 - 各组交流所采用的算法，预计每组5分钟左右



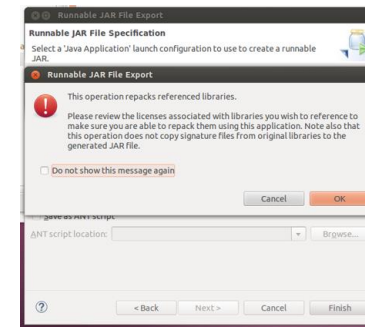
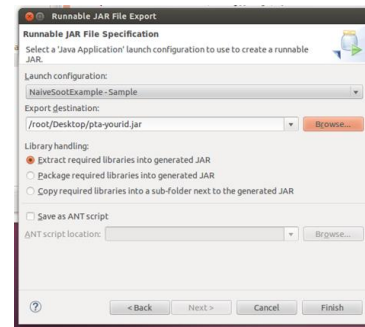
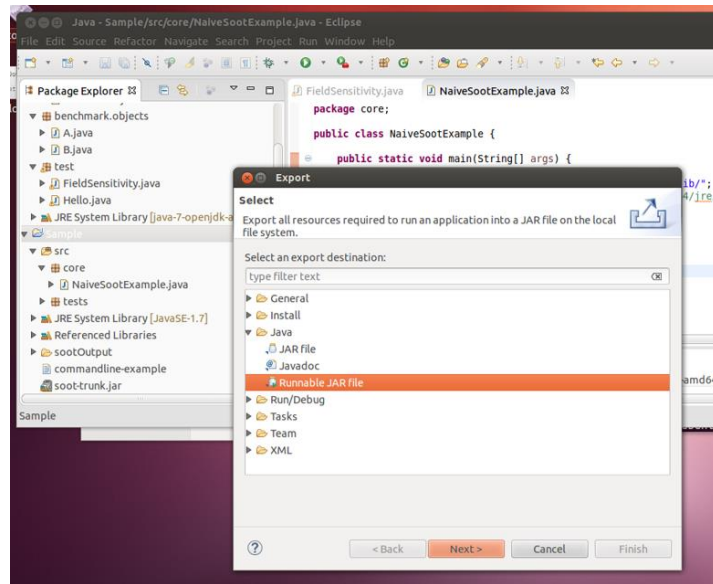
测试方法

- `java -jar analyzer.jar [src] [SomePackage.Main]`
 - [src]: 程序源码的根目录，同时包括java文件和class文件，并且包括JDK1.7版本的rt.jar和jce.jar（SOOT需要，开发包已经包括）
 - [SomePackage.Main]: 包含main函数的类名
- 输出写到result.txt



导出可执行的jar包（以eclipse为例）

- Jar包需要包括所有dependency





测试程序

- Ubuntu16.04 **JDK1.8** 双核CPU 实时给出程序运行结果
- 请使用队长的学号提交
- 测试一次程序所消耗的时间大概为3-4分钟，请不要一直连续提交评测任务。
- 提交网站：<http://162.105.88.145:8082>
- 10月18日 – 10月31日 简单的测试样例
- 11月1日 – 11月12日 复杂的测试样例
- 一个账号一天只能提交5次，组队完成后队长私信助教获得账号密码
- 时间：ddl提交样例测试程序之后，会更新排行榜上的对应排名，加入新编写的测试样例。



助教的忠告

- Soot 古老工具，几乎没有DOC，可以早点熟悉一下API。
- 保证sound、不崩溃以及不超时是得分高的关键
- Corner Case是更进一步的关键
- 限制使用的计算资源，并行不是可行的优化策略



一些透露了头像的前辈的忠告



啊我感觉没有什么特别特殊的建议🤔，就是一些比较平常的：如果以卷分数为目标的话，可以考虑一些面向测试用例特点的、比较ad hoc的方法；小组合作的时候最好组员之间的分工清晰明确一些，尽量并行地做；如果项目1还是pta的话，多熟悉了解soot的API可以避免一些造轮子



那就 project 半天是做不完的 至少得一天



一些透露了头像的前辈的忠告



保证soundness, 不用写太多优化



看了一下, 感觉我只能复读去年的忠告了, 保证 sound, 尽早开始, 不用写太复杂但是要测试边界情况



不过真的不用写太多



因为无法保证自己的程序没有bug



我们去年最后疯狂增加各种情况

hhh说明去年的忠告很不错



发现分数基本没变化



一些透露了头像的前辈的忠告



我就记得流敏感不是那么重要，域敏感比较重要



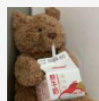
保证分析结果sound比啥都重要!



还有吐槽一点..soot的javadoc约等于没有，需要靠函数名猜功能 (x



建议先把算法要实现的规则写出来再写代码



不然debug会哭