# 程序分析工具的设计与实现

## 研究问题

本次实验能够根据给定的java代码生成其控制流程图，并能够根据控制流程图来对路径进行约束求解，从而判断该路径是否可满足。

## 实验步骤

通过调用soot的api来对java代码进行分析，从而得到这个java代码的语句块之间关系的dot文件。

然后我们对这个dot文件所表示的代码的控制流图进行广度优先遍历，在得到一条路径之后将其转换成为SMT的约束语句。通过广度优先遍历我们可以得到所有路径的约束语句。

最后我们将约束语句在Z3求解器中进行求解，如果结果为SATISFIABLE，则说明约束满足，并取出模型的解进行存储。

## 运行环境

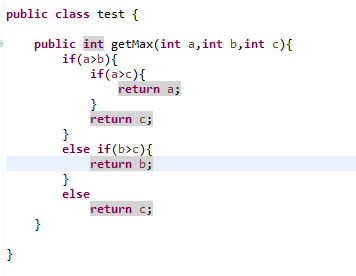
运行系统：windows 10

运行环境：Java 1.8版本，另外本次实验需要另外添加Z3和soot的依赖jar包以及可视化工具Graphviz。

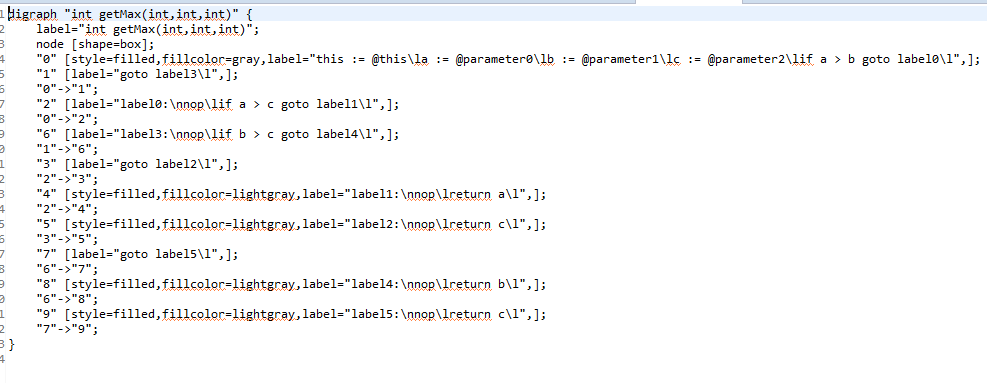
开发平台：Eclipse neon

## 实验结果

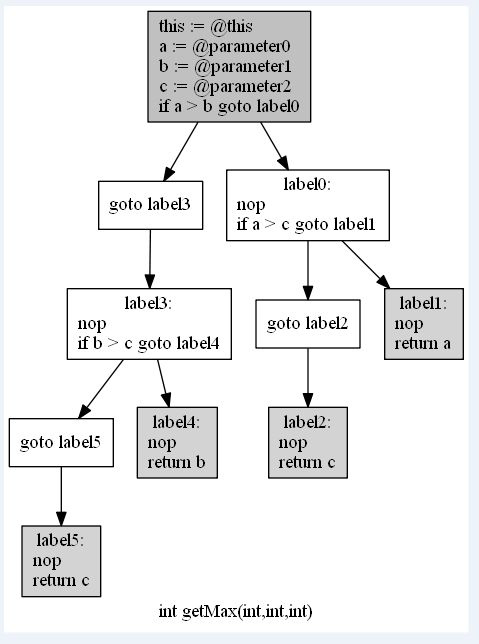
测试文件代码：



dot文件：



控制流程图：



求解结果：

|  |  |
| --- | --- |
|  |  |
|  |  |

## 总结

本工具通过对java代码进行分析能够生成相应的控制流程图，并能够根据控制流程图进行简单的路径约束求解工作。由于时间关系，本工具只对这一个测试用例进行了约束的求解内容，并且不涉及函数库的调用，比较的简单，对于较为复杂的应用场景并不能进行约束求解。

之后有时间会对其进行扩展，使得处理的场景更加丰富。