

活跃变量分析

1. 活跃变量

Live variable (about variable): A variable v is live at point p if the value of v is used along some path in the flow graph starting at p

For variable v and program point p , if the value of v at p can still be used along some path starting at p we say v is live at p .

- for each basic block, determine if each variable is live in this block.

For statement $s(d: x = y + z)$

- $Use[s] = \{y, z\}$
- $Def[s] = \{x\}$

2. 算法步骤

对于每个程序节点 n , 找到如下定义:

- $pred[n]$: 当前节点的前驱
- $succ[n]$: 当前节点的后继
- $def[n]$: 在当前节点定义的变量
- $use[n]$: 在当前节点使用的变量

输出:

- $in[n]$: 在当前节点属于live-in的变量
- $out[n]$: 在当前节点属于live-out的变量

算法:

- 数据流方程
 - $in[n] = use[n] \cup (out[n] - def[n])$
 - $out[n] = \bigcup in[s]$ (s 是 n 的所有后继)

```
for each node n in CFG
    in[n] = {};
    out[n] = {};
do {
    for each node n in CFG (reverse order) {
        in'[n] = in[n];
        out'[n] = out[n];
        out[n] =  $\bigcup in[s]$  ( $s$ 是 $n$ 的所有后继);
```

```
        in[n] = use[n] U (out[n] - def[n]);
    }
} until (in'[n] == in[n] && out'[n] == out[n]) // 收敛
```

3. 实现步骤

1. ILiveVariableRecorder.java
// 接口
2. LiveVariableDefinition.java
// 包含活跃变量名
3. LiveVariableRecorder.java
// 活跃变量中间结果, Def[n], Use[n], LiveIn[n], LiveOut[n]
4. LiveVariableAnalyzer.java
// 具体实现

4. 使用说明

1. 把JAnalyzer文件夹下.classpath文件中所有jar包的路径删除, 在eclipse中重新添加jar包
2. 修改文件路径为本地路径
 - TestCFGCreator.java

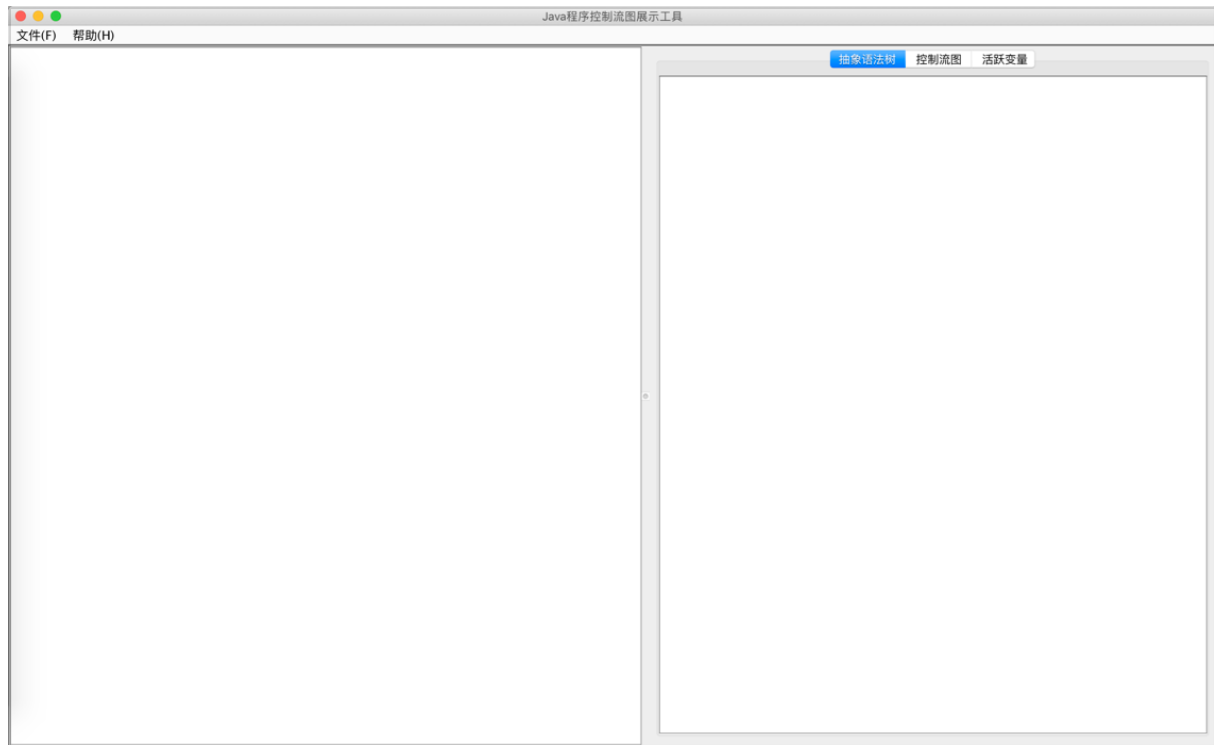
```
41      String rootPath = "/Users/merlyn/Study/java_code/JAnalyzer/";
42
43      String path = rootPath + "src/Test.java";
44      String result = rootPath + "result.txt";
```

- TestASTViewer.java (template.java是中间文件, 必须保存)

```
298 String path = "/Users/merlyn/Study/java_code/JAnalyzer/template.java";
```

3. 运行

打开 TestASTViewer.java, 在eclipse中点击运行



然后点文件->打开，打开文件后点击活跃变量分析，输出即为分析结果。（可以在TestASTViewer.java里面修改

```
329         if (controlFlowGraph != null)
330             output += LiveVariableAnalyzer.outPutDefAndUseVariable(
把 LiveVariableAnalyzer.outPutDefAndUseVariable 修改为
LiveVariableAnalyzer.outPutLiveInAndOutVariable，查看不同的输出
)
```

4. 输出说明

每一个可执行点包括两个变量集合，一个是live in，一个是live out