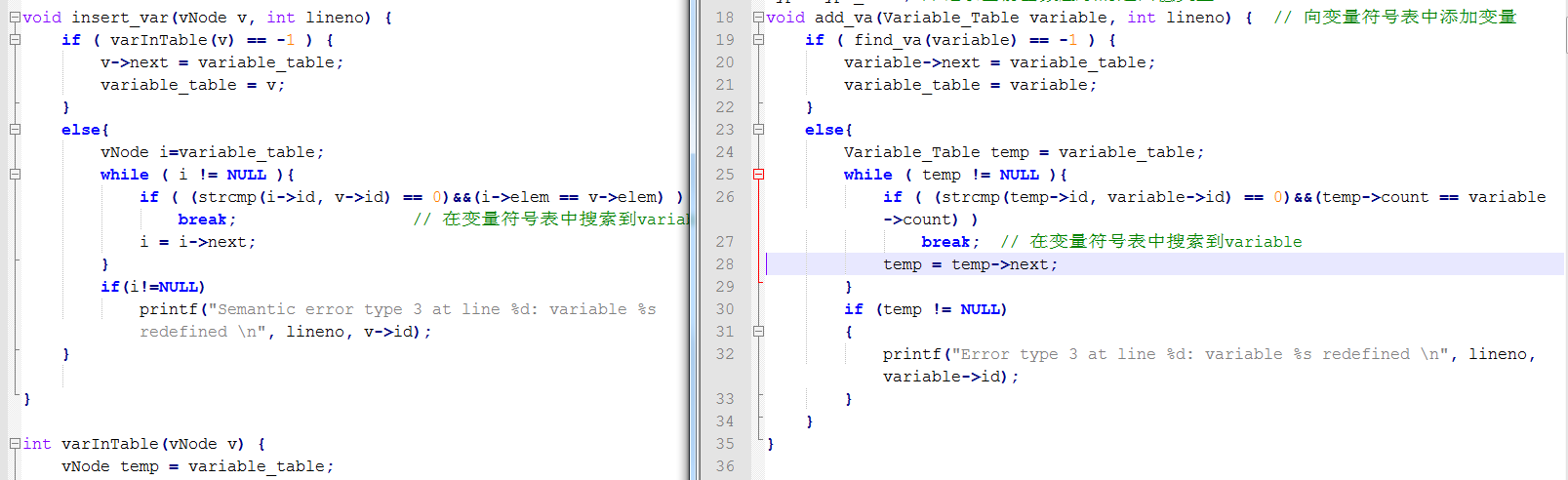
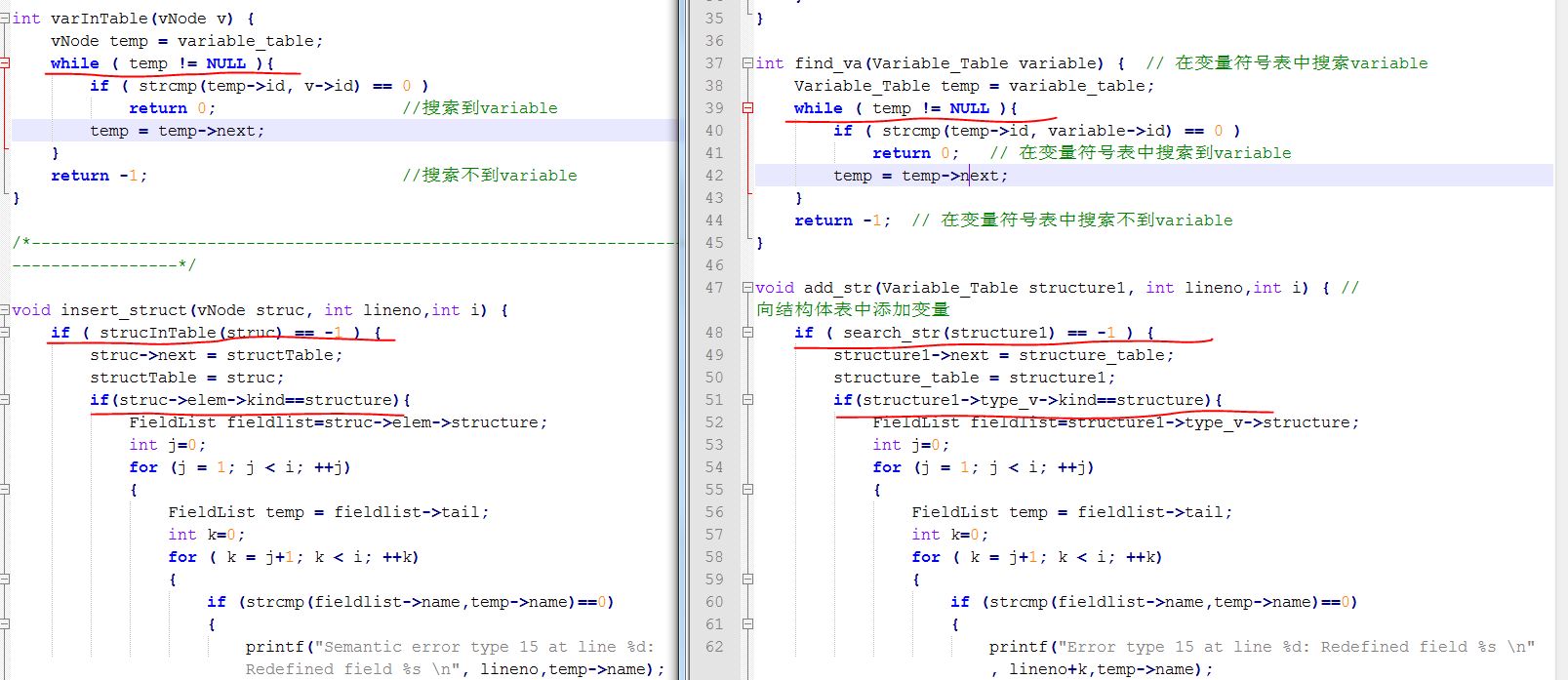
|  |  |  |
| --- | --- | --- |
| **嫌疑同学** | **相似度** | **鉴定结论** |
| 111220027->111220160 | 90.02% | 代码结构区别较大，抄袭痕迹不明显 |
| 111220160->111220120\_111220124 | 90.36% | 代码结构区别较大，抄袭痕迹不明显 |
| 111220054\_111220055->  111220132\_111220131 | 90.82% | 代码结构区别较大，抄袭痕迹不明显 |
| 111220027->111220077 | 91.28% | 代码结构区别较大，抄袭痕迹不明显 |
| 111220027->111220097 | 91.73% | 代码结构区别较大，抄袭痕迹不明显 |
| 111220039->111220029\_111220031 | 92.94% | 代码结构区别较大，抄袭痕迹不明显 |
| 111220097->111220160 | 93.02% | 确认抄袭（A） |
| 111220088->111220063 | 94.52% | 确认抄袭（B） |
| 111220160->111220097 | 96.14% | 确认抄袭（A） |

111220097->111220160

Trace.c（0097）和symbol.c（0160）高度雷同，例如下面这段代码，一开始所有的括号都是同一种空格风格，（前后都有空格），到最后的if都发生了突变。

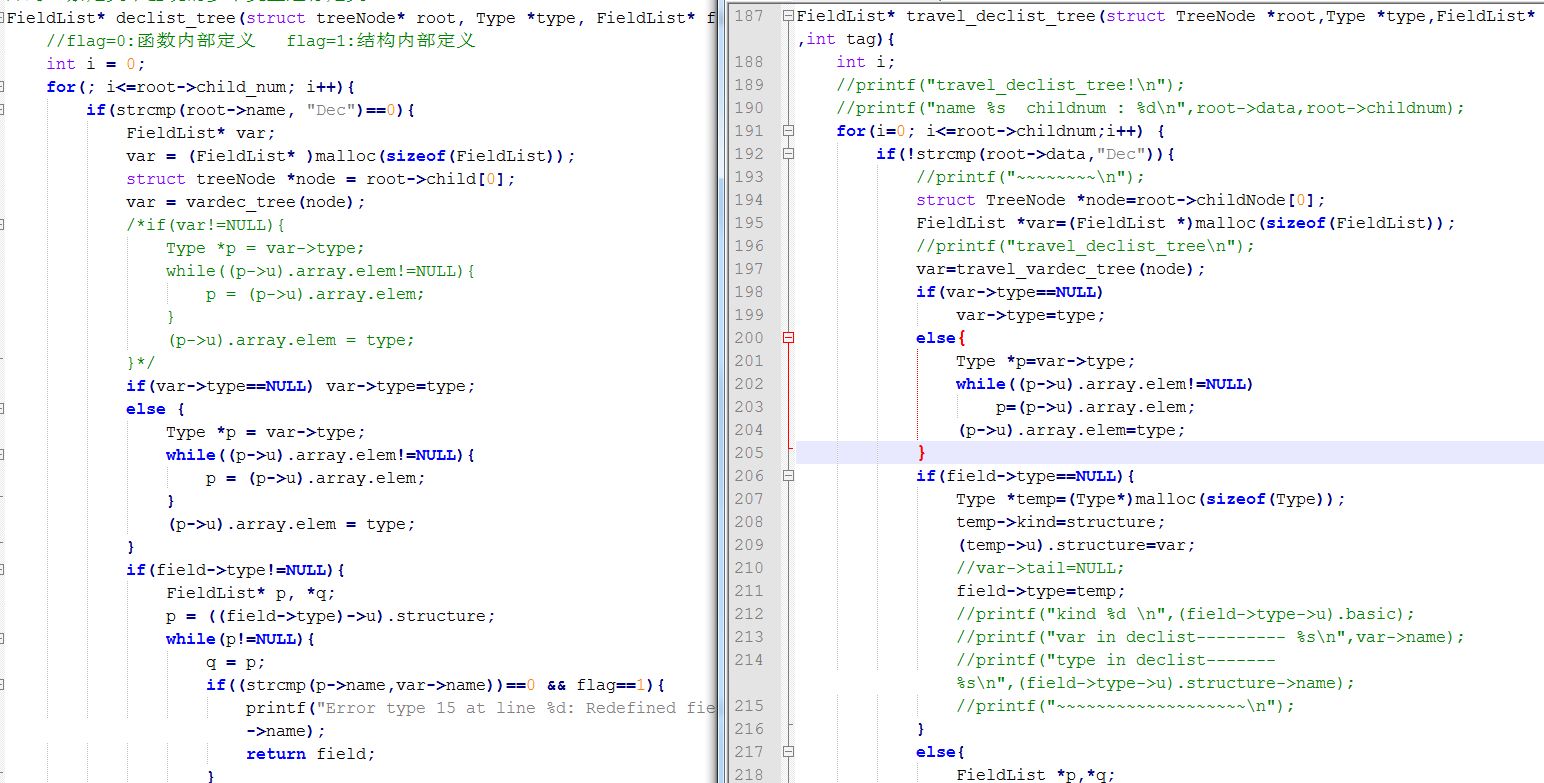


再比如下面这一段，小括号后面跟大括号的风格发生了两次变化，非常同步，如图中划出的三句。



111220088->111220063

semantic\_analysis.c（0063）和semantic.c(0088)代码大量雷同，但是两位同学是两种不同的代码风格，而且各自都是基本一致的，而且采用的完全不同的变量名风格。抄袭痕迹比较明显的有下面这个函数，他们各自的第一个for循环的分号后面跟空格的风格有一致的突变



再比如下面这一段，判断是否是左值的代码，正常来说，肯定不会是完全一样的。

