

Codesim 实验报告

MG20330050 侍林天

2020 年 11 月 23 日

1 引言

Codesim需要我们做的工作简单来说就是给出两份代码之间的相似程度，而这份工作的应用场景往往是对代码进行查重，找出OJ竞赛或学生作业中的作弊行为。所以我们需要去思考什么的两份代码是相似的，尤其是在代码查重的应用场景下。首先，两份相似或者疑似作弊的代码相似度绝不可能出现在两份代码的书写上，改变代码的变量名和函数名是一种常见的作弊方式，并且有自动化的代码模糊工具，能够一键将代码重写成面目全非但仍保持原功能的样子。所以我们必须剥开代码比如.cpp文件是由ASCII码组成的文本文件的表象，而是以更抽象更深入地运用静态分析的方法去挖掘代码所表示的更抽象的信息。

2 代码的抽象表示

上下文无关文法（context-free grammar）是形式语言的一种，几乎所有程序设计语言都是通过上下文无关文法来定义的。

在本Codesim工具中，我们先使用Clang根据C++的语法规则将代码解析为一个抽象语法树（Abstract Syntax Tree）。