

编译原理 实验 2 实验报告

201220062 黄子睿

一、实验功能简介：

- a) 程序实现了 c—语言要求的所有基础语义分析，并通过了 OJ 平台上的所有测试用例。
- b) 选做部分完成如下：
 - i. 要求 2.1，可以识别函数声明
 - ii. 要求 2.2，支持嵌套变量定义
 - iii. 要求 2.3，支持结构等价
- c) 下面简要叙述一下语义分析的代码实现方式：
 - i. 先通过实验 1 的代码，由词法与语法分析生成语法树，通过函数 `traverseTree` 对语法树进行遍历。
 - ii. 语义分析的符号表由开散列哈希实现，通过十字链表实现符号表对嵌套定义的处理。在符号表中使用双向列表，以方便快速插入与删除。
 - iii. 对类型的定义由 `struct` 与 `union` 实现，为了实现识别函数声明，在 `type` 对象中增加对函数状态的记录，包括 `declared` 与 `defined`；为了实现嵌套变量定义，在符号表中增加 `stack` 数组记录每次嵌套情况；为了实现结构等价，按序比较 `struct` 类型中的 `fieldList` 域。

二、编译方式介绍

- a) 提交的文件结构为：
 - i. Code
(相关代码文件与 Makefile 及 python 脚本 `autorun`)
 - ii. Test
(测试输入文件)
 - iii. Result
(测试输出文件)
 - iv. Report.pdf
- b) 代码的运行需要在 Code 文件夹下。输入 `make` 指令可以编译得到 `parser`，通过输入 `./parser xxx` 可以在控制台上输出结果。
- c) 在 Code 文件夹下输入 `make clean` 将删除所有输出文件与 Test 文件夹下的所有结果文件。
- d) 在 Code 文件夹下输入 `make run` 可以执行 `autorun.py` 脚本，批量执行 Test 文件夹下的输入文件，结果存放在 Result 文件夹下。