

实验选题

可执行程序及使用说明请见[README.md](#)

将 `c` 语言转为 `llvm` 中间表示

功能实现

使用 `ANTLR` 对 `c` 语言进行词法分析及语法分析，转为语法树后使用 `visitor` 模式由上至下遍历进行语义分析，并转为 `llvm` 中间表示代码

测试程序

- 回文检测：

```
python parse.py tests/palindrome.c out.ll  
lli out.ll
```

- 排序：

```
python parse.py tests/sort.c out.ll  
lli out.ll
```

- KMP字符串匹配：

```
python parse.py tests/kmp.c out.ll  
lli out.ll
```

- 四则运算计算：

```
python parse.py tests/calc.c out.ll  
lli out.ll
```

- 深度优先前序遍历：

```
python parse.py tests/dfs.c out.ll  
lli out.ll
```

该程序为自选程序，使用到结构体语法，样例输入：“1,2,3,4,5”，程序会尝试按照输入顺序构建完全二叉树，并输出该树的深度优先前序遍历结果，样例输出：“1 2 4 5 3”

语法支持

- ✓ `include` 头文件
- ✓ 基础类型支持（包括 `int`，`double`，`char`，`void`）
- ✓ 函数声明定义
- ✓ 结构体声明定义
- ✓ 赋值语句
- ✓ 变量定义语句（包括基础类型，数组，结构体等）
- ✓ 条件语句（包括 `if`，`else if`，`else`）
- ✓ 循环语句（包括 `for`，`while`）
- ✓ 表达式（包括各种运算，变量及函数调用等）

创新点

词法分析及语法分析结构借鉴了[SpiderMonkey的AST结构](#)，更加清晰明确，易于扩展