

# TINY+ 语义分析程序及中间代码生成实验

- 运行环境: Linux(Ubuntu) g++ make
- 可执行程序路径: ./tiny/bin/main
- 源程序文件夹: ./tiny
- 报告: REPORT.pdf
- 运行方法:

```
1 cd tiny # 进入tiny文件夹
2 make # Makefile
3 #输出符号表和三地址中间代码到屏幕(控制台)
4 ./bin/main test.tny
5 #输出三地址中间代码到code文件
6 ./bin/main test.tny code
7 #输出三地址中间代码到code文件,输出错误报告信息到errors文件,输出优化过的代码
8 ./bin/main test.tny code errors optimize
9 #输出语法树到屏幕(控制台)
10 ./bin/main test.tny tree
11 #输出TOKEN序列到屏幕(控制台)
12 ./bin/main test.tny tokens
13 #输出TOKEN序列到tokens文件
14 ./bin/main test.tny tokens tokens
```

- 文件结构

```
1 tiny
2 |— .vscode
3 |— bin
4 |   |— main //可执行程序
5 |— include // .h文件夹
6 |   |— errors.h //声明枚举错误类型
7 |   |— global.h //声明全局变量等
8 |   |— scan.h //声明getToken获取序列函数
9 |   |— print.h //声明实现printToken函数
10 |   |— parser.h //声明语法分析器头文件
11 |   |— generate.h //生成中间代码
12 |— src // .cpp文件夹
```

```

13 |   |   └─ main.cpp  //主函数
14 |   |   └─ scan.cpp  //实现getToken获取序列
15 |   |   └─ print.cpp  //实现输出Token的函数
16 |   |   └─ parser.cpp  //实现语法分析，添加语义分析及符号表
17 |   |   └─ generate.cpp  //实现生成中间代码
18 |   └─ Makedfile  //Makefile文件
19 |   └─ test.tny  //测试tiny源程序
20 |   └─ tokens  //输出文件

```

- 输出示例

以下为部分示例，更多测试详见REPORT.pdf。

```

1  root@DESKTOP-6L638NB:/mnt/e/code/Compiler/tiny# ./bin/main
   test.tny
2
3  TINY COMPILATION: test.tny
4  Variable  Type  ValType
5  -----
6  A          value  Int
7  B          value  Int
8  C          value  Int
9  D          value  Int
10 fact       value  Int
11 x          value  Int
12 y          value  Bool
13 z          value  Str
14
15 1) read x
16 2) Label L1
17 3) t0:=A*2
18 4) A:=t0
19 5) t1:=A+C
20 6) t2:=B+D
21 7) Label L2
22 8) if t1<t2 goto L3
23 9) goto L1
24 10) Label L3
25 11) t3:=B+C
26 12) t4:=A+t3
27 13) Label L4
28 14) if t4<10 goto L4
29 15) goto L6
30 16) Label L5
31 17) t5:=B+3
32 18) B:=t5

```

```
33 19) goto L4
34 20) Label L6
35 21) Label L7
36 22) if x<10 goto L8
37 23) goto L9
38 24) Label L8
39 25) if x>5 goto L10
40 26) goto L9
41 27) Label L9
42 28) if x<9 goto L10
43 29) goto L11
44 30) Label L10
45 31) fact:=4
46 32) goto L12
47 33) Label L11
48 34) fact:=6
49 35) Label L12
50 DONE
```