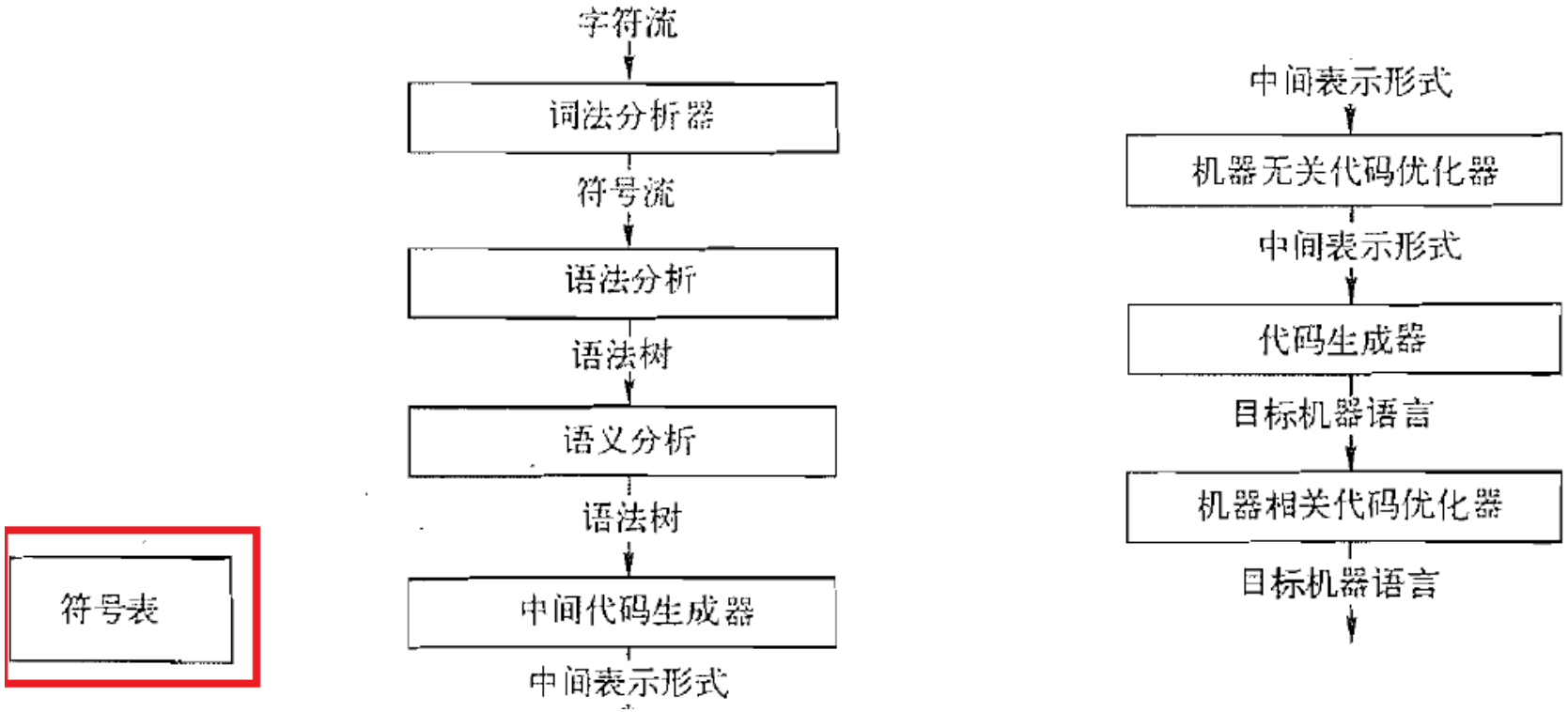
(1.)

hfwei@nju.edu.cn

20230412


字符流
词法分析器
符号流
语法分析
语法树
语义分析
语法树
中间代码生成器
中间表示形式
符号表
中间表示形式
机器无关代码优化器
中间表示形式
代码生成器
目标机器语言
机器相关代码优化器
目标机器语言

Definition ((Symbol Table))

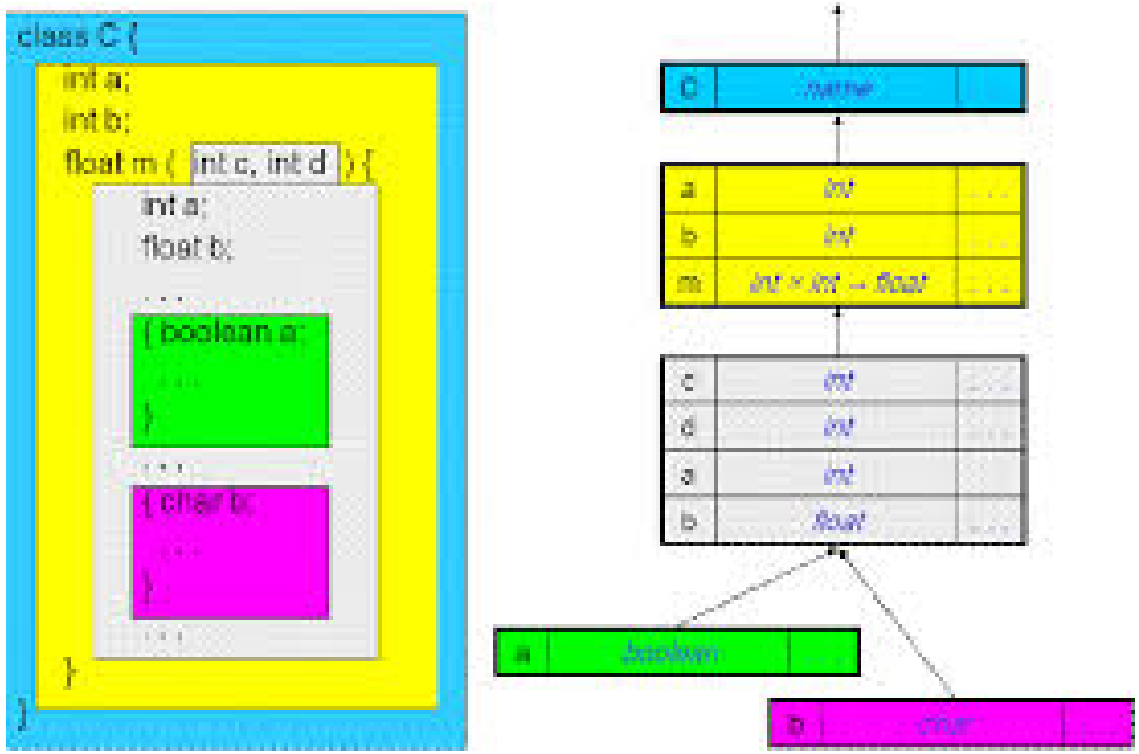
Definition ((Symbol Table))

Name	**Type**	**Size**	**Dimension**	**Line of Declaration**	**Line of Usage**	**Address**	$\cdots$
count	int	4	0	$\cdots$	$\cdots$	$\cdots$	$\cdots$
str	char[]	5	1	$\cdots$	$\cdots$	$\cdots$	$\cdots$

“” (DSL) ()

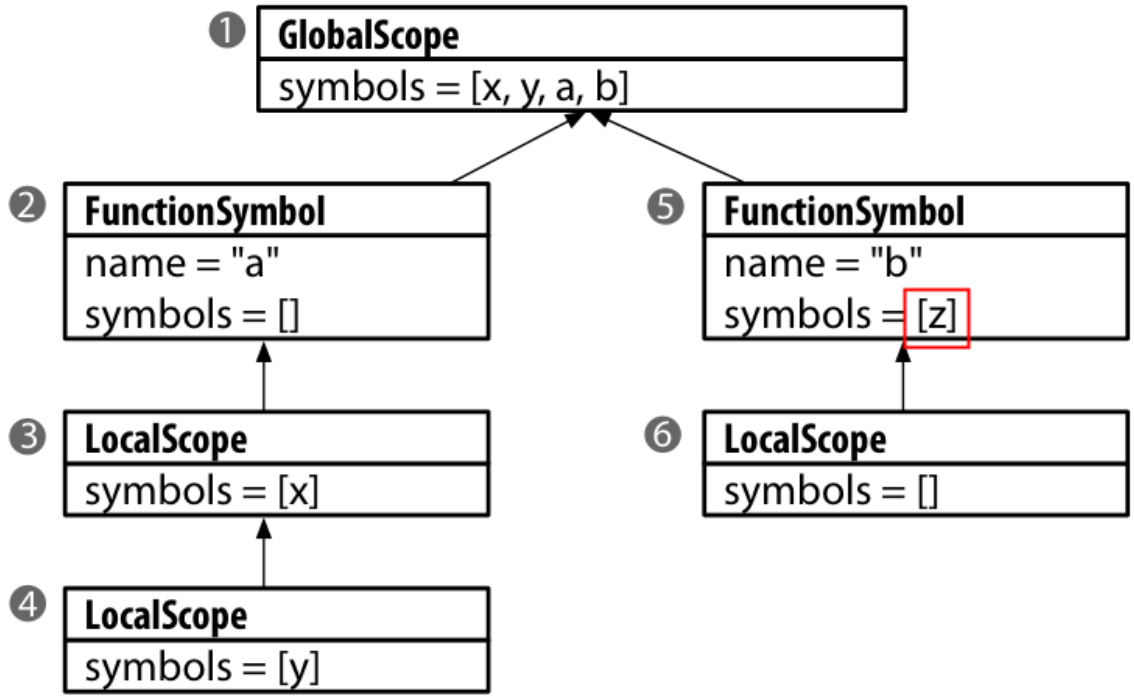
```
host=antlr.org
port=80
webmaster=parrt@antlr.org
```

“” (GPL)

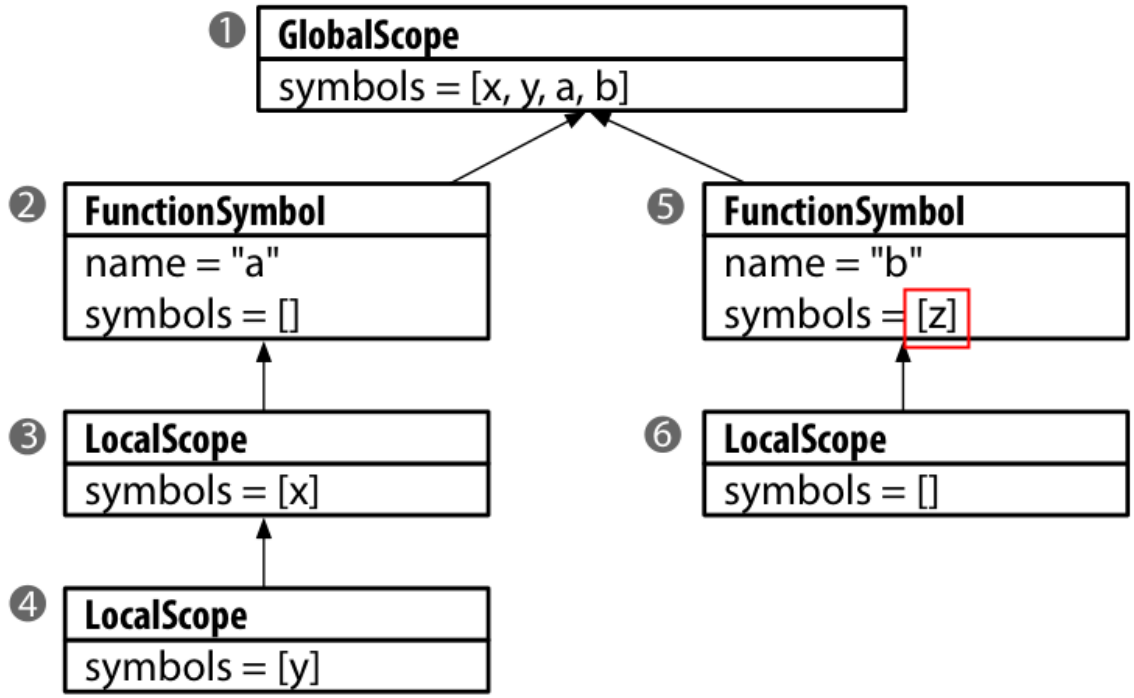

class C {
int a;
int b;
float m (int c, int d) {
int a;
float b;
{ boolean a;
}
{ char b;
}
}
}
C
name
a
int
b
int
m
int × int → float
c
int
d
int
a
int
b
float
a
boolean
b
char

```
1 int x;
  int y;
2 void a()
3 {
      int x;
      x = 1;
      y = 2;
4     { int y = x; }
  }
5 void b(int z)
6 { }
```

```
1 int x;
  int y;
2 void a()
3 {
      int x;
      x = 1;
      y = 2;
4     { int y = x; }
  }
5 void b(int z)
6 { }
```

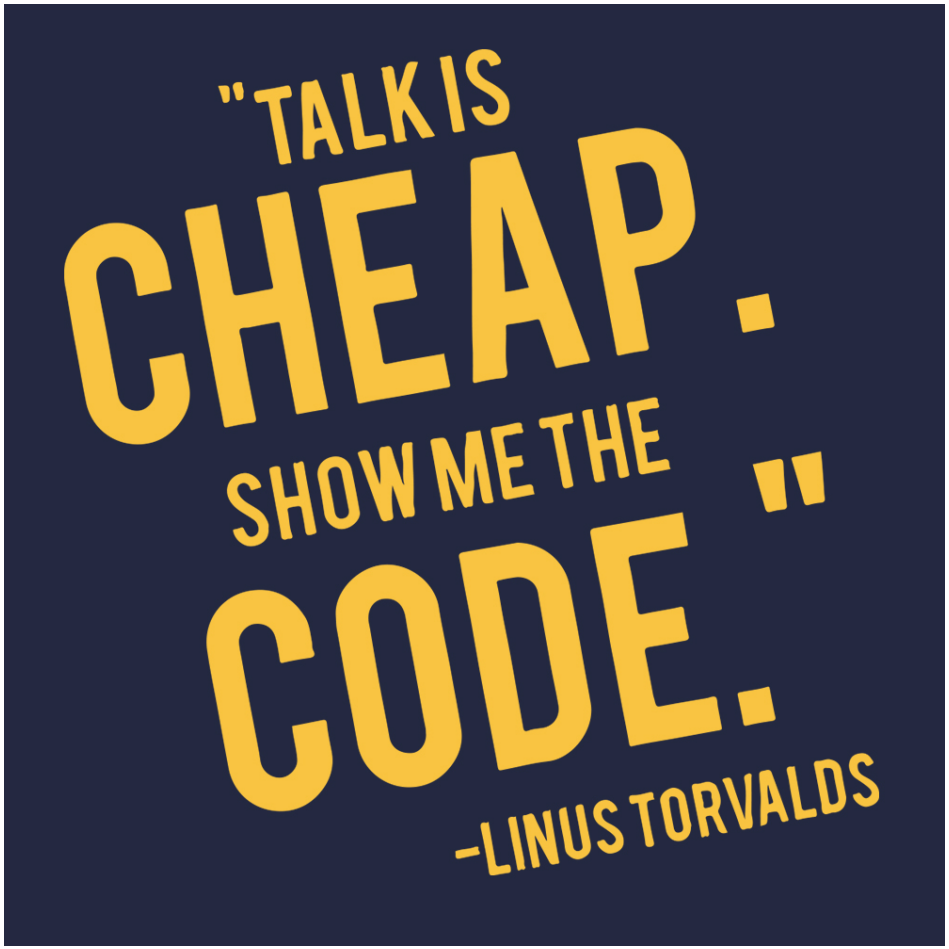


```
1 int x;
  int y;
2 void a()
3 {
      int x;
      x = 1;
      y = 2;
4     { int y = x; }
  }
5 void b(int z)
6 { }
```

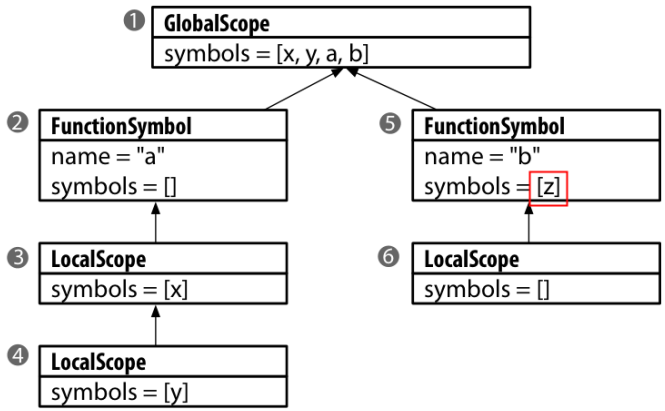


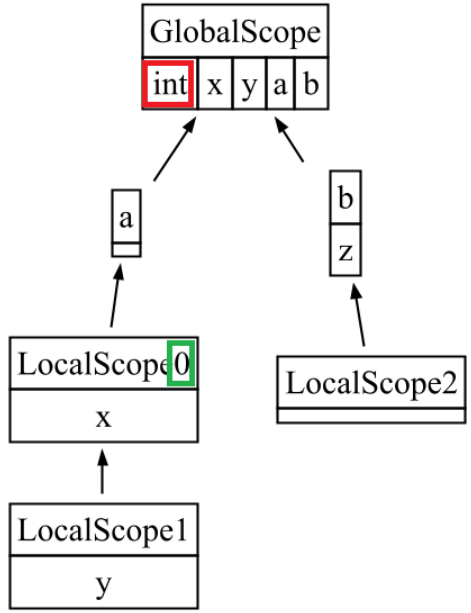
We take a **WRONG** assumption here about `FunctionSymbol`'s scope.

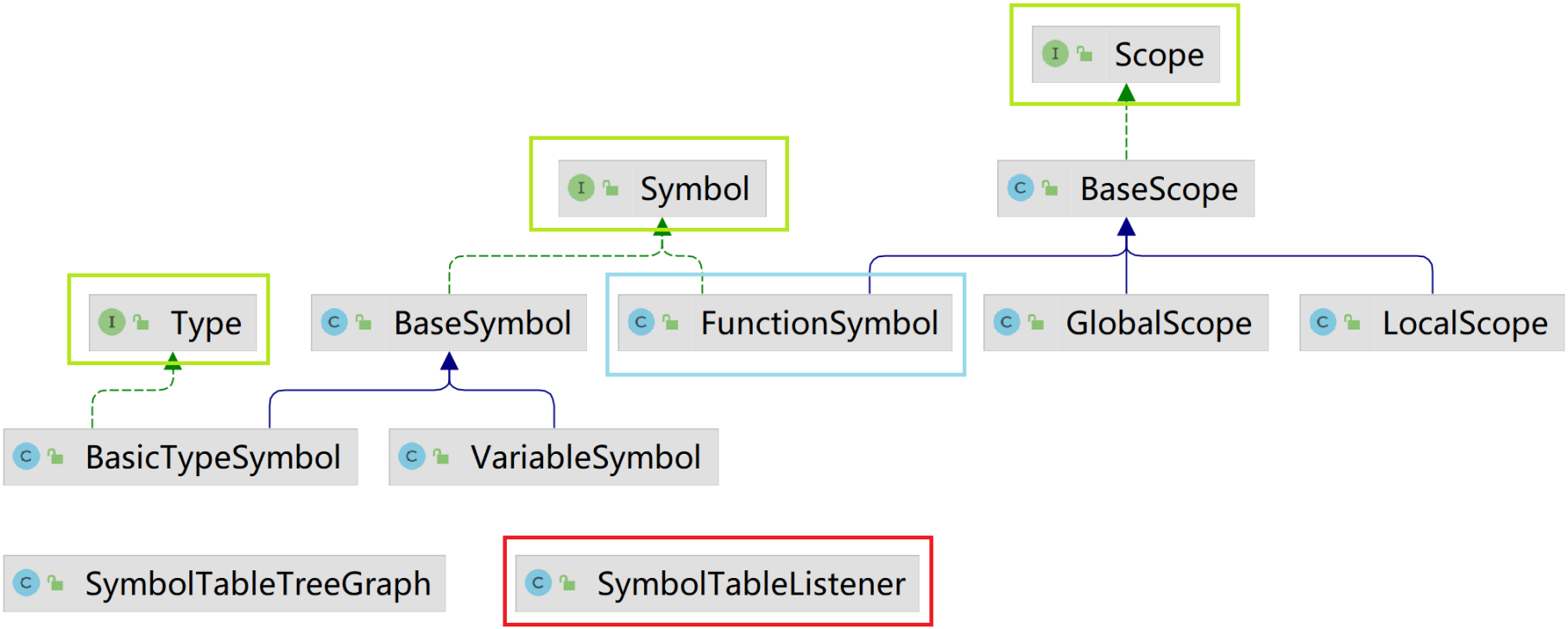
```
public interface Scope {
    public String getScopeName();          // 有名称吗?
    public Scope getEnclosingScope();      // 有外部作用域吗?
    public void define(Symbol sym);        // 在作用域中定义符号
    public Symbol resolve(String name);    // 根据名称查找
}
```

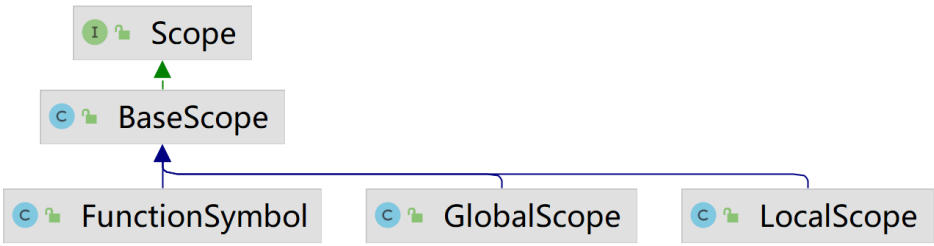

"TALK IS
CHEAP.
SHOW ME THE
CODE."
-LINUS TORVALDS

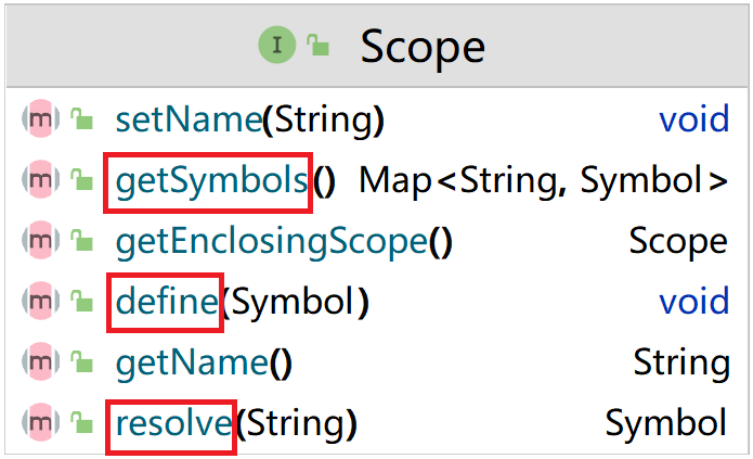
```
1 int x;
  int y;
2 void a()
3 {
      int x;
      x = 1;
      y = 2;
4     { int y = x; }
  }
5 void b(int z)
6 { }
```





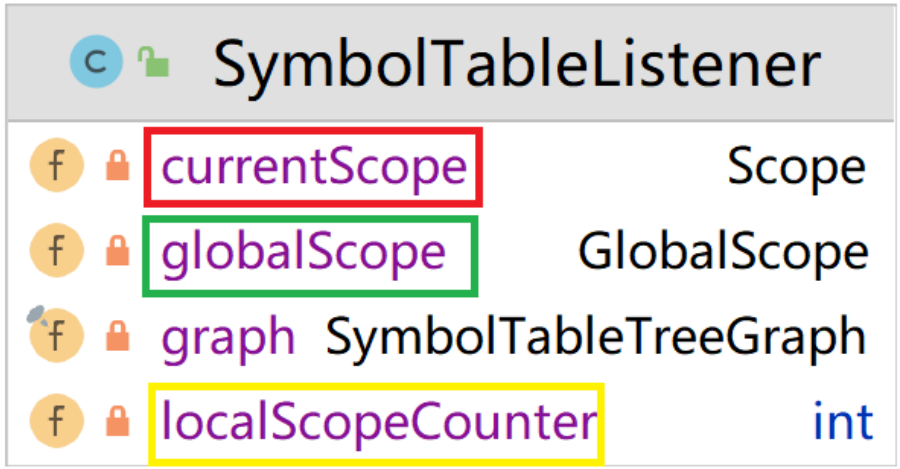

Scope
Symbol
BaseScope
Type
BaseSymbol
FunctionSymbol
GlobalScope
LocalScope
BasicTypeSymbol
VariableSymbol
SymbolTableTreeGraph
SymbolTableListener

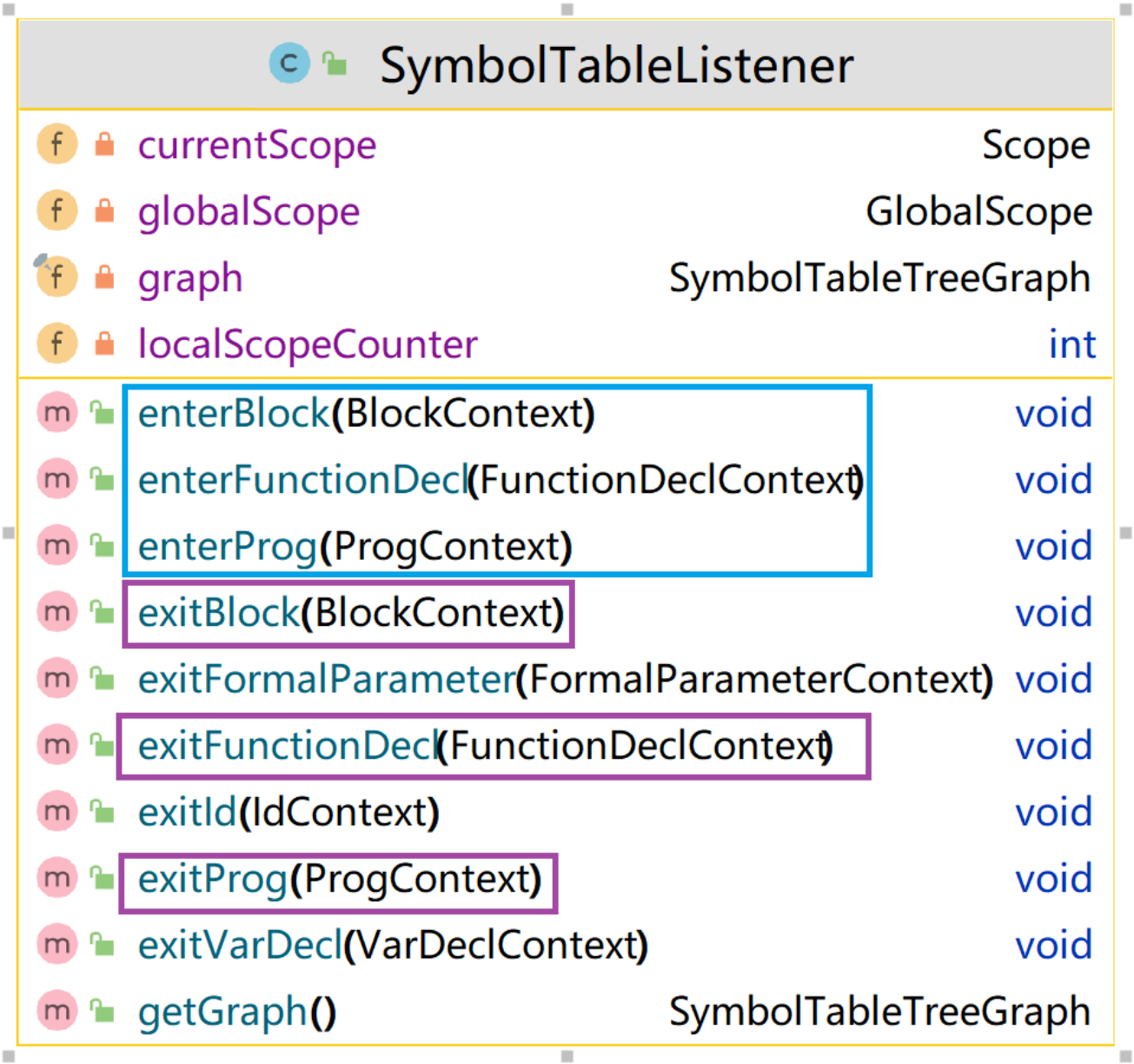

Scope
BaseScope
FunctionSymbol
GlobalScope
LocalScope


Scope
setName(String) void
getSymbols() Map<String, Symbol>
getEnclosingScope() Scope
define(Symbol) void
getName() String
resolve(String) Symbol


SymbolTableListener


SymbolTableListener

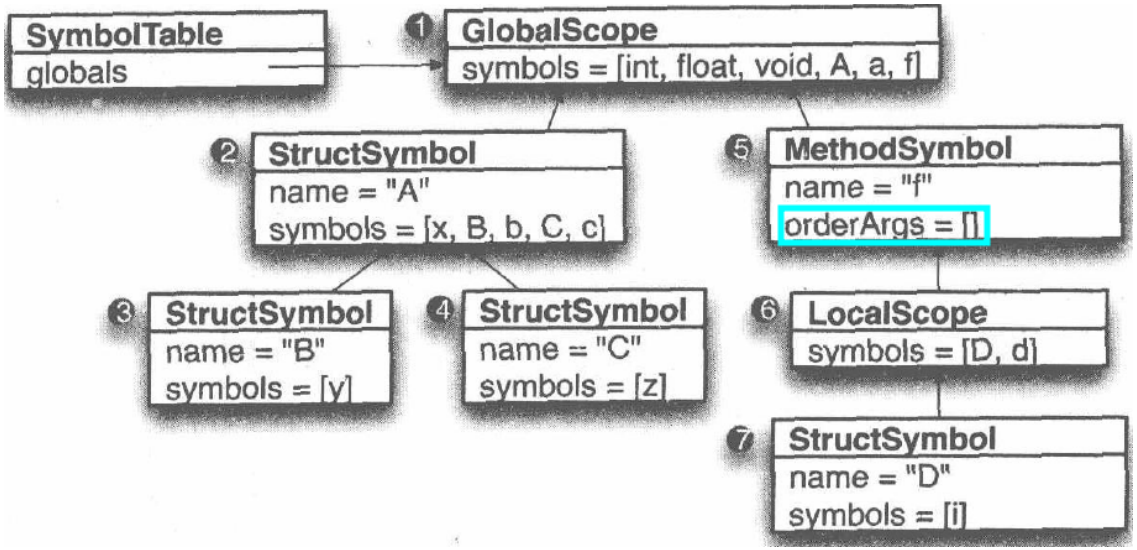

SymbolTableListener
currentScope
Scope
globalScope
GlobalScope
graph
SymbolTableTreeGraph
localScopeCounter
int


SymbolTableListener
currentScope Scope
globalScope GlobalScope
graph SymbolTableTreeGraph
localScopeCounter int
enterBlock(BlockContext) void
enterFunctionDecl(FunctionDeclContext) void
enterProg(ProgContext) void
exitBlock(BlockContext) void
exitFormalParameter(FormalParameterContext) void
exitFunctionDecl(FunctionDeclContext) void
exitId(IdContext) void
exitProg(ProgContext) void
exitVarDecl(VarDeclContext) void
getGraph() SymbolTableTreeGraph

struct/class:

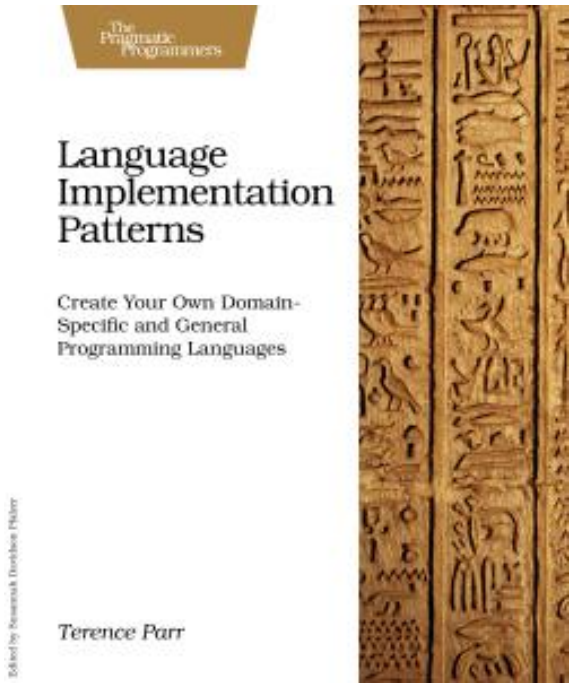
```
❶
❷  struct A {
     int x;
❸    struct B { int y; };
     B b;
❹    struct C {int z; };
     C c;
   };
   A a;

❺  void f()
❻  {
❼    struct D {
       int i;
     };
     D d;
     d.i = a.b.y;
   }
```

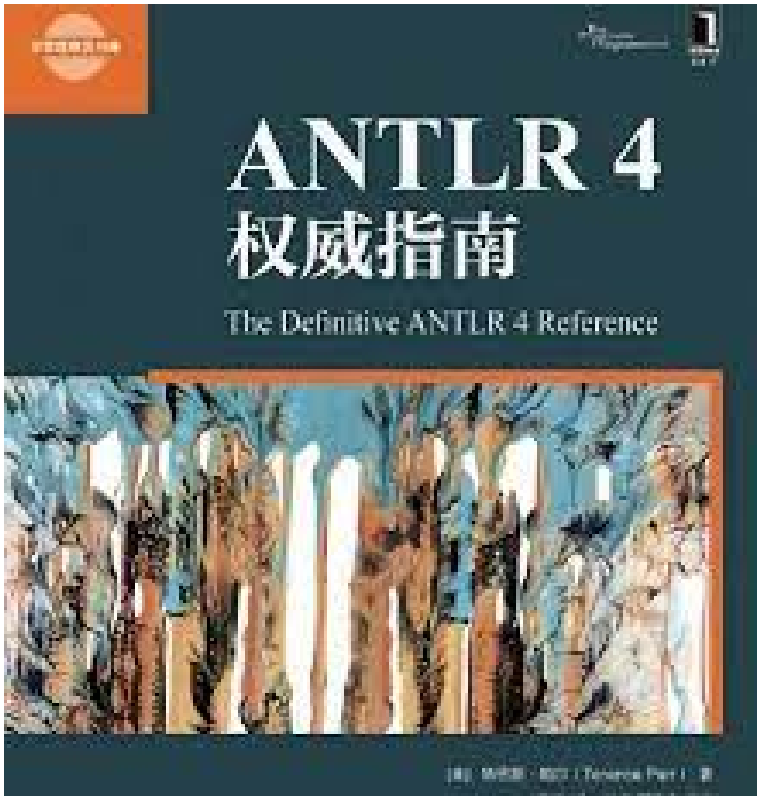


$d.i$ $a.b.y$

6 :


The Pragmatic Programmers
Language Implementation Patterns
Create Your Own Domain-Specific and General Programming Languages
Terence Parr

7 :

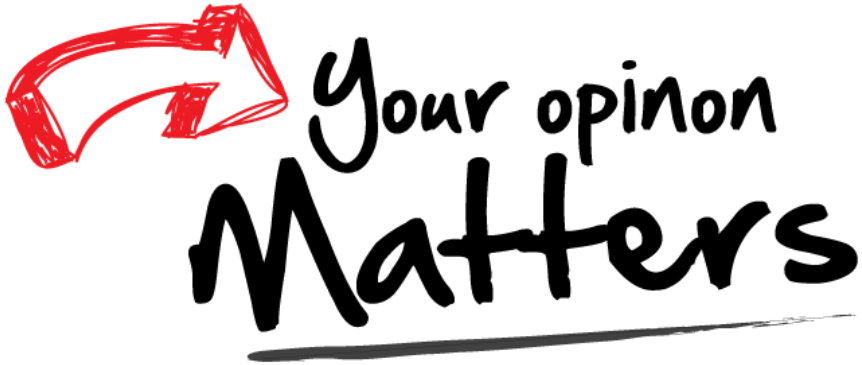


8.4 :

symtab @ antlr by parrt

symtab @ cs652 by parrt


Thank
You!



Office 926

hfwei@nju.edu.cn