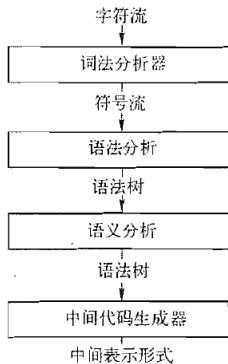


(1.)

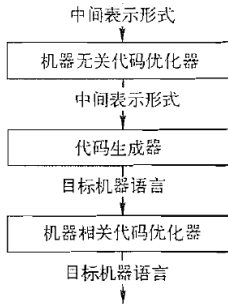
hfwei@nju.edu.cn

20230412





符号表



Definition ((Symbol Table))

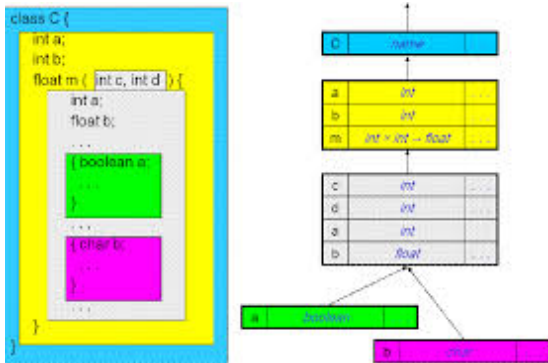
Definition ((Symbol Table))

Name	Type	Size	Dimension	Line of Declaration	Line of Usage	Address	...
<i>count</i>	int	4	0
<i>str</i>	char[]	5	1

“” (DSL) ()

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

“” (GPL)

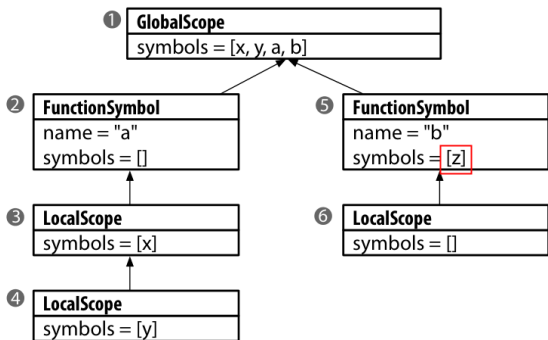


```
1 int x;  
  int y;  
2 void a()  
3 {  
    int x;  
    x = 1;  
    y = 2;  
4    { int y = x; }  
    }  
5 void b(int z)  
6 { }
```

```

1 int x;
  int y;
2 void a()
3 {
    int x;
    x = 1;
    y = 2;
4     { int y = x; }
5 }
6 void b(int z)
7 { }

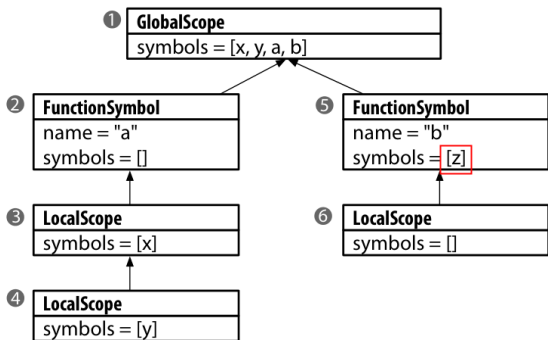
```




```

1 int x;
  int y;
2 void a()
3 {
    int x;
    x = 1;
    y = 2;
4     { int y = x; }
5 }
6 void b(int z)
7 { }

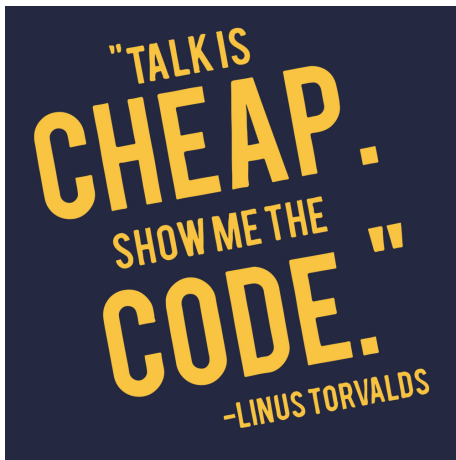
```



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  
}
```

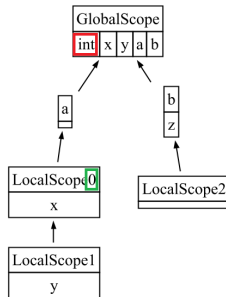
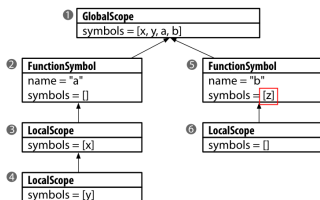
/

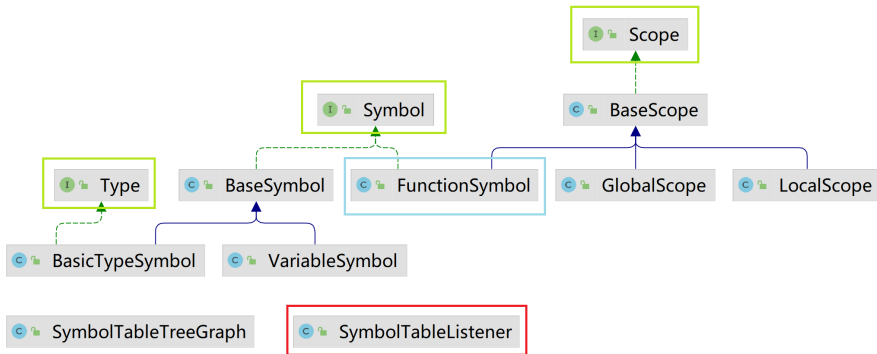


```

1 int x;
  int y;
2 void a()
3 {
    int x;
    x = 1;
    y = 2;
4     { int y = x; }
5 }
6 void b(int z)
7 { }



```







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

  SymbolTableListener

SymbolTableListener

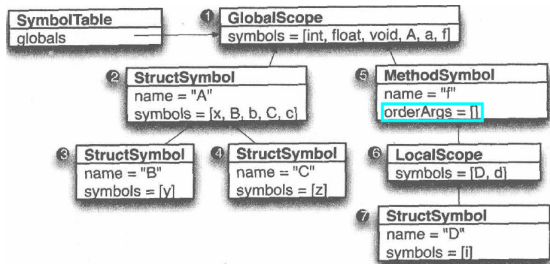
SymbolTableListener	
f	currentScope Scope
f	globalScope GlobalScope
f	graph SymbolTableTreeGraph
f	localScopeCounter int

SymbolTableListener

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

struct/class:

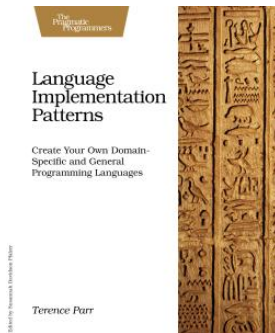
```
1 struct A {  
2   int x;  
3   struct B { int y; };  
4   B b;  
5   struct C {int z; };  
6   C c;  
7 };  
8 A a;  
9  
10 void f()  
11 {  
12   struct D {  
13     int i;  
14   };  
15   D d;  
16   d.i = a.b.y;  
17 }
```



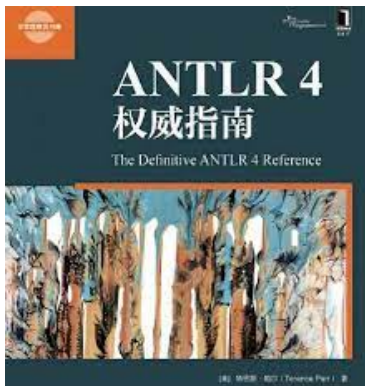
d.i

a.b.y

6 :



7 :

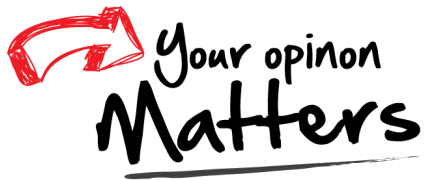


8.4 :

symtab @ antlr by parrt

symtab @ cs652 by parrt

Thank
You!



Office 926

hfwei@nju.edu.cn