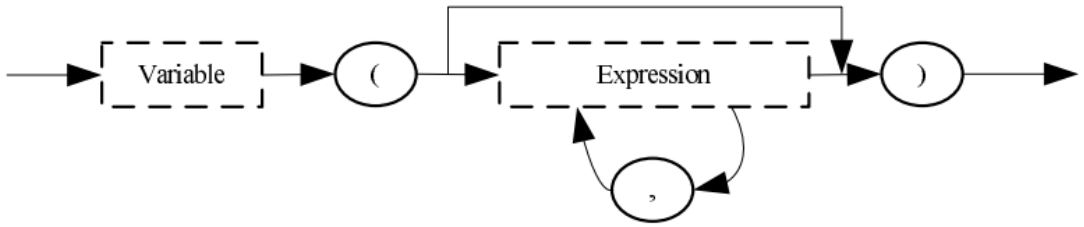
Ziyan Wang

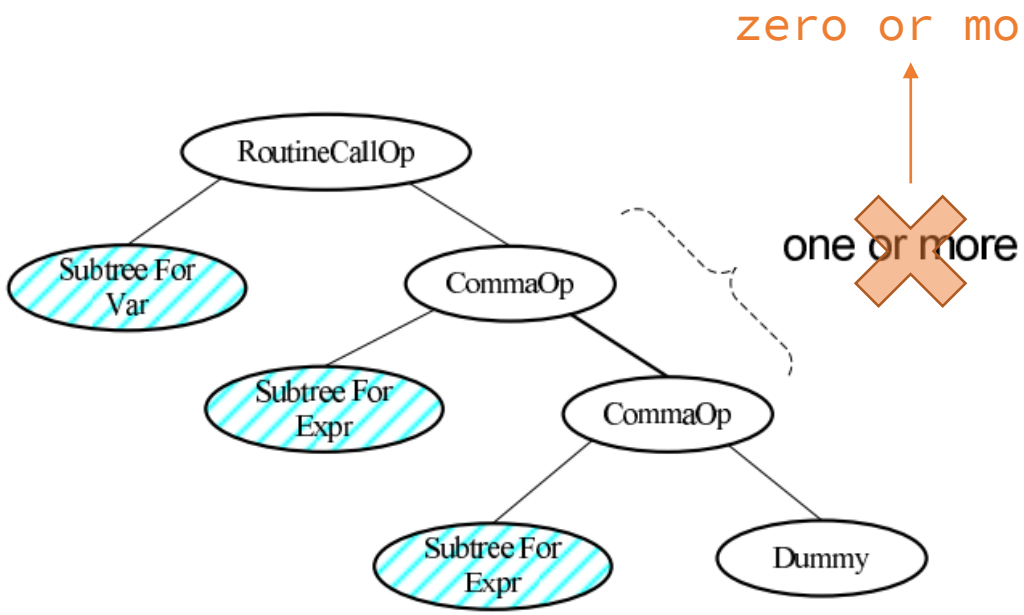
Project 2
Q&A

修正一处错误（附录B-4）

MethodCallStatement



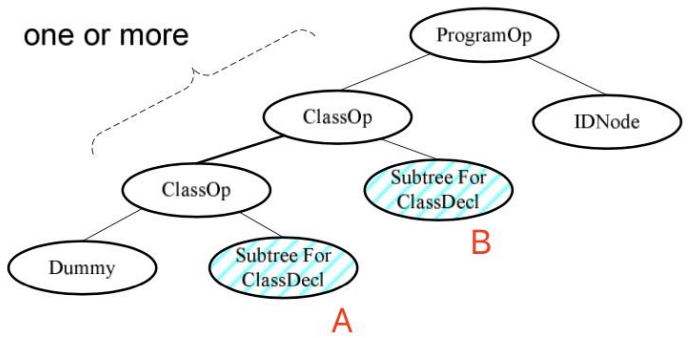

```
foo(bar, baz);
foo(bar);
foo();
```

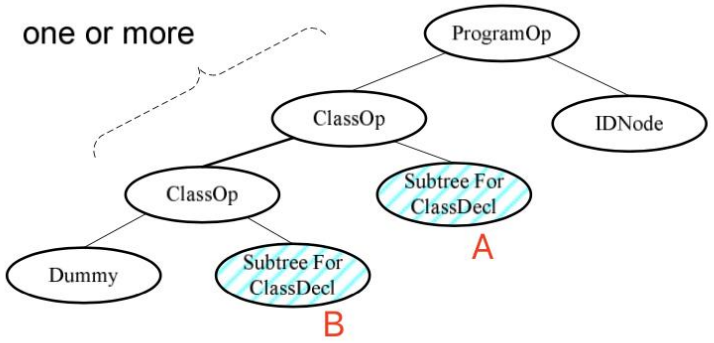


- Q: proj2为什么不同人写的语法分析器，输出会不同？

- A: 文档没有规定递归的方向，可以写成左递归，也可以写成右递归。

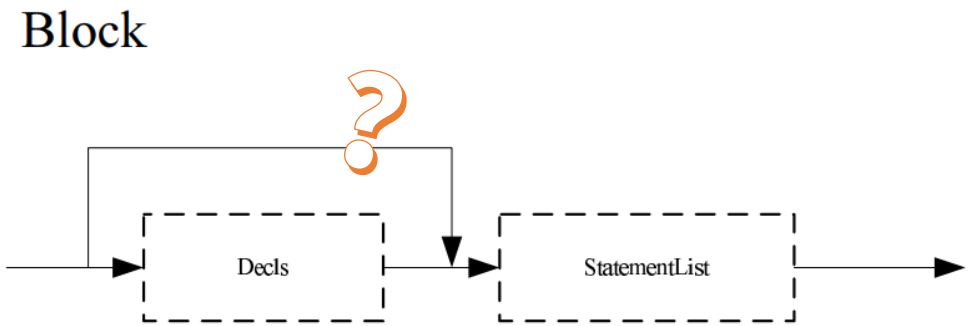
```
program xyz;
class A {}
class B {}
```

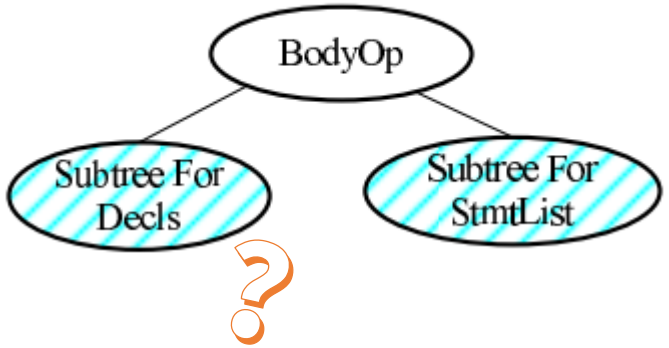




- 可以自己选择左递归或右递归，但整个语法树应保持一致，都用左递归或右递归。

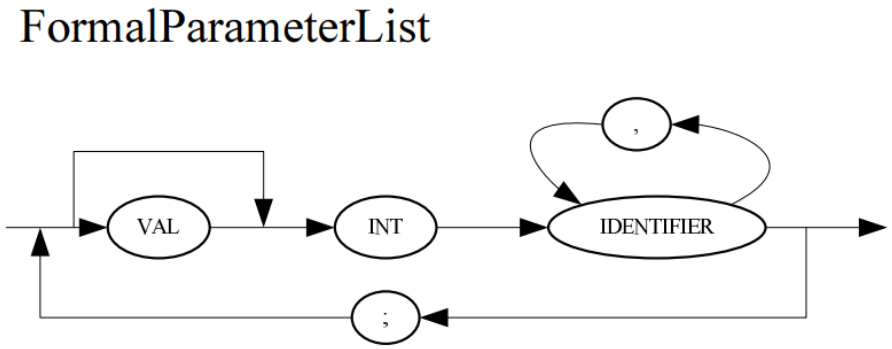
- Q: 如何处理规则中可以跳过的部分?

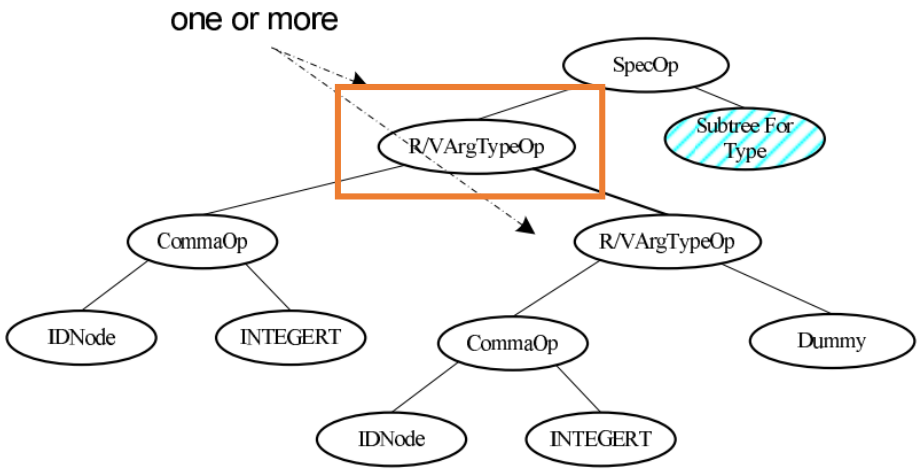




- A: 如果代码中没有Decls，则BodyOp的左子树设为Dummy

- Q: 什么是R/VArgTypeOp?





- A: 表示函数参数的传值类型
- RArgTypeOp: Reference Argument Type 引用传递
- VArgTypeOp: Value Argument Type 值传递

```
method void main (val int x; int y) {}
```

值传递　　引用传递

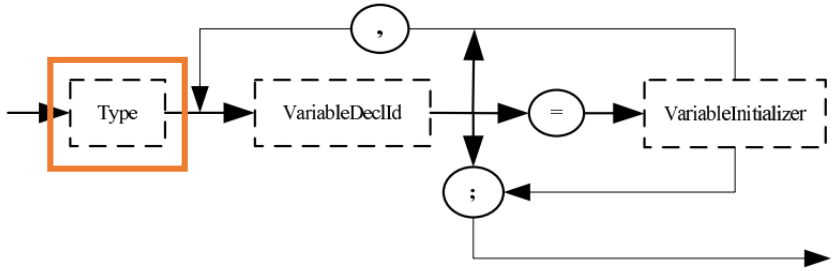
- Q: Type should be stored in a separate pointer (global variable) such that it may be used in building the *VariableInitializer* subtree.

这是什么意思？

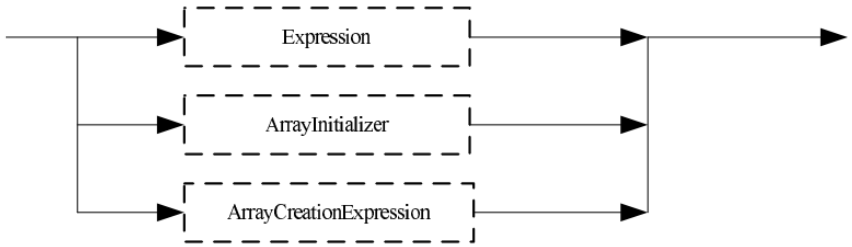
- A:

FieldDecl

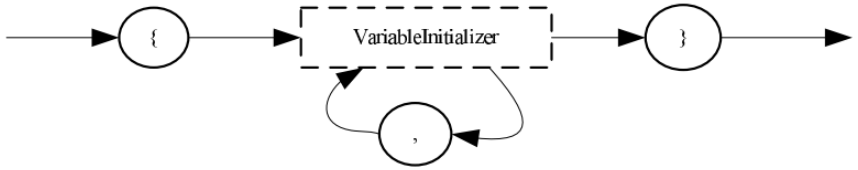
```
int[] array = {2, 3, 4, 5};
```

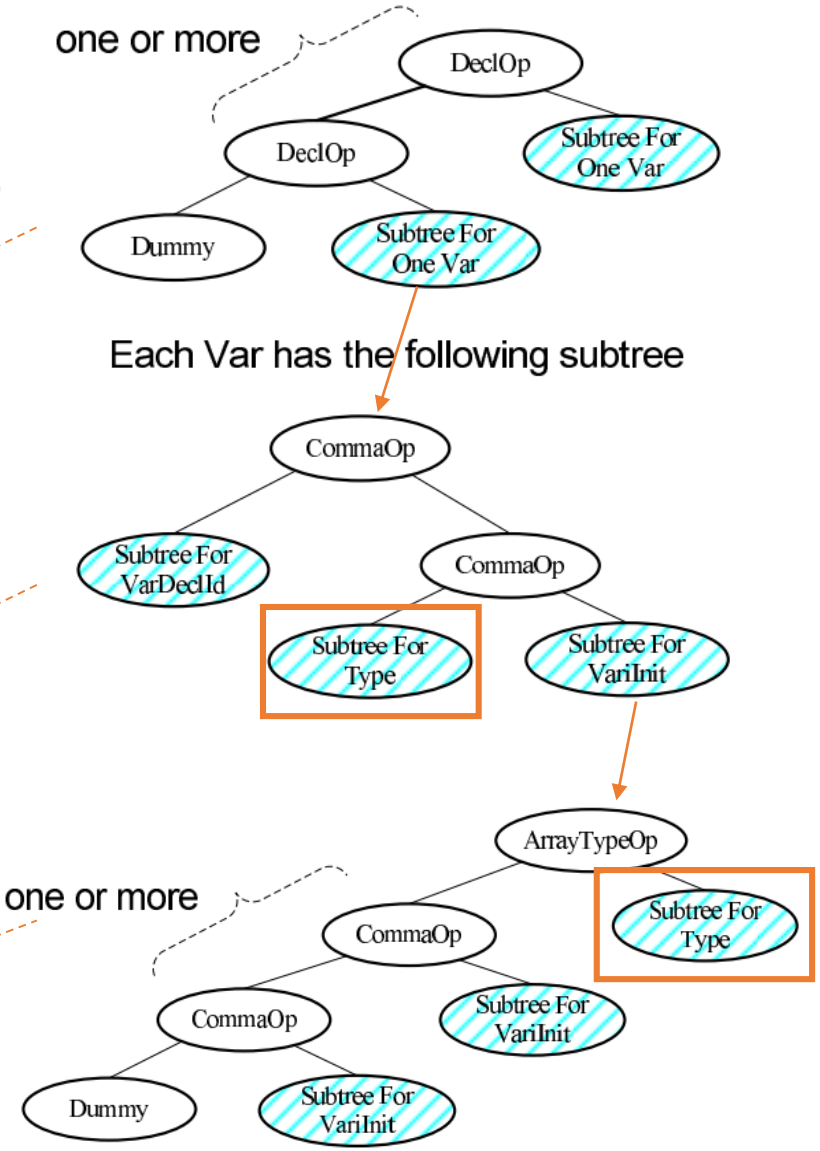


VariableInitializer



ArrayInitializer





- Q: Type should be stored in a separate pointer (global variable) such that it may be used in building the *VariableInitializer* subtree.

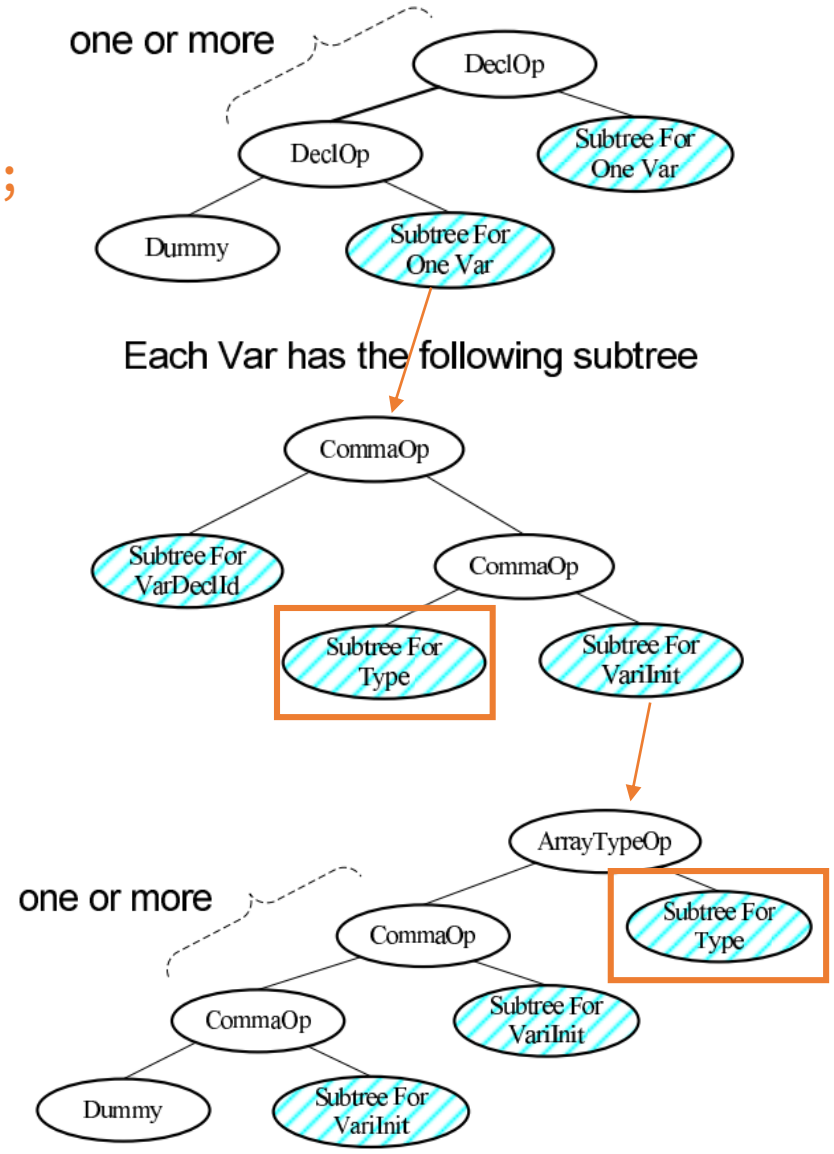
这是什么意思?

```
int[] array = {2, 3, 4, 5};
```

- A:

```
FieldDecl : Type {
  fieldType = $1;
};

FieldDeclSub :
VariableDeclId EQUALnum VariableInitializer {
  $$ =
    MakeTree(DeclOp, NullExp(),
      MakeTree(CommaOp, $1,
        MakeTree(CommaOp, fieldType, $3)));
};
```



其他注意事项

- 评测环境
  - `flex 2.6.4`
  - `bison (GNU Bison) 3.5.1`
- 如果你使用的flex和bison版本与上述版本不相同，必须在Readme文件中注明你的版本号

Thanks