

# Grammar for the Exp programming language

Version 9

program  $\rightarrow$  ( function )<sup>\*</sup> main

function  $\rightarrow$  **def name ( ) { ( statement )<sup>+</sup> }**

main  $\rightarrow$  ( statement )<sup>+</sup>

statement  $\rightarrow$  st\_print | st\_attrb | st\_if | st\_while | st\_break | st\_continue  
| st\_array\_new | st\_array\_push | st\_array\_set | st\_call

st\_call  $\rightarrow$  **name ( )**

st\_print  $\rightarrow$  **print ( expression ( , expression )<sup>\*</sup> )**

st\_attrb  $\rightarrow$  **name = expression**

st\_if  $\rightarrow$  **if** comparison **{ ( statement )<sup>+</sup> ( } else { ( statement )<sup>+</sup> } )<sup>?</sup> }**

st\_while  $\rightarrow$  **while** comparison **{ ( statement )<sup>+</sup> }**

st\_break  $\rightarrow$  **break**

st\_continue  $\rightarrow$  **continue**

st\_array\_new  $\rightarrow$  **name = [ ]**

st\_array\_push  $\rightarrow$  **name . push ( expression )**

st\_array\_set  $\rightarrow$  **name [ expression ] = expression**

comparison  $\rightarrow$  expression ( **=** | **!=** | **<** | **<=** | **>** | **>=** ) expression

expression  $\rightarrow$  term ( ( **+** | **-** ) term )<sup>\*</sup>

term  $\rightarrow$  factor ( ( **\*** | **/** | **%** ) factor )<sup>\*</sup>

factor  $\rightarrow$  **number** | **string** | ( expression ) | **name** | **read\_int ( )** | **read\_str ( )**  
| **name . length** | **name [ expression ]**

---

tokens: + - \* / % ( ) = , { } == != < <= > >= [ ] .

number string name read\_int read\_str

print if else while break continue push length **def**

bytecode instructions:

+1 **ldc** *integer or "string"*

-1 **iadd**

-1 **isub**

-1 **imul**

-1 **idiv**

-1 **irem**

-1 **istore** *index*

+1 **iload** *index*

-1 **astore** *index*

+1 **aload** *index*

+1 **getstatic**

-2 **invokevirtual** .../print(I)V

-2 **invokevirtual** .../print(Ljava/lang/String;)V

-1 **invokevirtual** .../println()V

+1 **invokestatic** .../readInt()I

-2 **if\_icmp??** *label*                      ?? → **eq** | **ne** | **lt** | **le** | **gt** | **ge**

0 **goto** *label*

+1 **new** Array

+1 **dup**

-1 **invokespecial** Array/<init>()V

-2 **invokevirtual** Array/push(I)V

0 **invokevirtual** Array/length()I

0 **invokevirtual** Array/string()Ljava/lang/String;

-3 **invokevirtual** Array/set(II)V

-1 **invokevirtual** Array/get(I)I

0 **invokestatic** Test/function()V