

```

LangA.sdf3
1 module LangA
2
3 imports Common
4
5 context-free start-symbols Program Expr
6
7 sorts Program Expr Class Field Method Param
8
9 context-free syntax
10
11 Program.Program = [
12   program [[ID]
13
14     [{Class "\n\n"}*]
15
16     run
17       [Expr]
18   ]
19
20 Class.Class = [
21   class [ClassName] [Extends] {
22     [{Field "\n"}*]
23     [{Method "\n\n"}*]
24   }

```

```

expr-arith.spt
24
25 test multiply and add (1) [[
26   1 + 2 * 3
27 ]] parse to Add(., Mul(., .))
28
29 test multiply and add (2) [[
30   1 + 2 * 3
31 ]] parse to [[
32   1 + (2 * 3)
33 ]]
34

```

```

class01.la
1 program class01
2
3 class A {
4   Num x;
5   Num y;
6 }
7
8 class B extends A {
9   Bool x;
10 }
11

```