SMDP Exercises 6

Model Driven Development, IT-University of Copenhagen
Theresa Brandt von Fackh (tebr@itu.dk) & Niels Søholm (nmar@itu.dk)
2014-03-12

We hand in this exercise as a group, because we were working on one computer due to the xtext issues with Theresa's MAC Computer. The grammar still could not be recognised after several attempts on the MAC (Snow Leopard with Java 1.6). After several hours we decided to only use Niels Computer.

Task 3 - Grammar

```
Model returns Model:
    {Model}
    'Model'
    '{'
        (assertMethods+=AssertMethod ( assertMethods+=AssertMethod)* )?
    '}';

Exp returns Exp:
    UOp | NULL | Id0 | BOp_expr | BOpMethod | FunCall;

AssertMethod returns AssertMethod:
    name=EString
    '(' params+=Parameter ( "," params+=Parameter)* ')'
    '{'
        bodyExpr=Exp
    '}';

EString returns ecore::EString:
    STRING | ID;

Parameter returns Parameter:
    name=EString ':' type=SimpleTypeEnum;

enum SimpleTypeEnum:
    DOUBLE='double' |
    BOOLEAN='boolean' |
    LONG='long' |
    CHAR='char' |
    INT='int' |
    FLOAT='float'|
    SHORT='short' |
    OBJECT='Object'
;

UOp returns UOp:
    'UOp'
    '{'
        'operator' operator=EString
        'expr' expr=Exp
    '}';
```

(continued on next page..)

```
NULL returns NULL:
    {NULL}
    'NULL'
    ;

Id0 returns Id:
    {Id}
    name=EString;

BOp_expr returns Exp:
    BOp_primary({BOp.lexpr=current} operator=EString rexpr=BOp_primary)
;

BOp_primary returns Exp:
    Id0 | NULL | '(' Exp ')'
;

BOpMethod returns BOpMethod:
    'BOpMethod'
    '{'
        'operator' operator=EString
        'lexpr' lexpr=Exp
        'rexpr' rexpr=Exp
    '}';

FunCall returns FunCall:
    {FunCall}
    name=EString
    '('
        (arg+=Exp ( "," arg+=Exp)*)?
    ')';
```

Comment: We implemented left recursion elimination, but somehow we could not make it apply to function calls. The compiler would throw a bunch of exceptions if function calls without parenthesis were allowed in a term. The result is that function calls can only be used if parenthesis are added around them as seen in the assertEquals example.

Task 4 - Example model

```
Model {
    assertEquals(expected:double, actual:double, delta:double) {
        ("Math.abs" ("expected" "-" "actual")) "<=" "delta"
    }
    assertTrue(condition:boolean) {
         "condition" "+" "true"
    }
    assertNotNull(object:Object) {
        "object" "!=" "Null"
    }
}
```