

Generating Software from Specifications WS 2013/14 - Solution 3

Solution for Exercise 6

The specification file `TryScanSol.fw` contains the commented solution of this exercise. Here are a few remarks on it:

The number of given test cases is to be considered as an absolute minimum! Some students are observed to check an unacceptable small number of test cases.

Be aware that canned token descriptions include the association of a useful coding function, which is implemented in a library of Eli.

Be aware that comments are not considered to be tokens. They may occur before or after any token in the program. Hence, they do not occur in the concrete syntax, and they are not named in the token specification.

The description of line comments of Java is easily obtained by copying that of Ada line comments, and replacing the `--` by `//`, where the `/` characters have to be escaped.

Solution for Exercise 7

The directory `blatt3/solution` contains the commented solution of this exercise in the file `CalendarSol7.fw`.

Solution for Exercise 8

The solution in File `CalendarSol8.fw` continues that of the previous exercise.

1. Define `DayNames` using an EBNF construct.

The production

```
DayNames:      DayName / DayNames ' , ' DayName .
```

is substituted by

```
DayNames:      DayName // ' , ' .
```

2. Modify the structure of `Entry` such that the time or the time span is not combined with the `Description`, but with the `Date`.

The productions

```
Entry:         Date Event .
Event:         When Description / Description .
```

are substituted by

```
Entry:         DateAndTime Description .
DateAndTime:   Date When .
```

3. Modify the structure of `Entry` such that `Date`, time or time span, and `Description` are direct components of `Entry`.

The production for `Entry` is changed into

```
Entry:         Date When Description .
When:          Time / Time '-' Time / .
```

The productions for `Event` and `DateAndTime` are eliminated.

4. Simplify the productions for patterns of dates: Specify `Modifier` to be optional; then remove unnecessary productions.

The productions for patterns are substituted by

```
GeneralPattern:      SimplePattern Modifier.  
SimplePattern:      'Weekday' / 'Weekend'.  
Modifier:           '+' DayNames / '-' DayNames / .
```

5. Finally extend the language such that also spans of dates can be described, like "20.8. - 5.9." or "Tue-Thu" but not "Tue-4.5."

Two more alternatives are inserted for Date:

```
Date: DayNum '.' MonNum '.' '-' DayNum '.' MonNum '.' /  
      DayName '-' DayName.
```