Exercise VI

Generalization of the repetition

The language MINI has to be extended with a generalization of the repetition. Example:

```
FOR i UPTO 10
WHILE invoer i geldig
REP
   doe iets met i
ENDREP
```

The syntax of the MINI language is therefore changed and augmented by the following syntax rules.

The repetition

Syntax

a) `repetition:`
```
    for part option, from part option,
       direction part option, while part option,
       rep token, paragraph, until part option,
       endrep token.
```

b) `for part:`
```
    for token, loop variable name.
```

c) `loop variable name:`
```
    small name.
```

d) `from part:`
```
    from token, expression.
```

e) `direction part:`
```
    upto token, expresion;
    downto token, expression.
```

f) `while part:`
```
    while token, condition.
```

g) `until part:`
```
    until token, condition.
```

Furthermore, the rules for `while clause` and `loop body` are deleted from the grammar.

Context conditions

a) The repetition always has type void.

b) The loop variable must be declared. It must have type INT and access VAR.

d) The expression must have type INT and access CONST.

e) The expressions must have type INT and access CONST.

f) The condition must have type BOOL and access CONST.

g) The condition must have type BOOL and access CONST.

Semantics

If the for part is absent, an anonymous variable of type INT and access VAR is used as the loop variable. Initially the from part, if present, is evaluated and assigned to the loop variable. If the from part is absent the value 1 is assigned to the loop variable. Next the direction part, if present, is evaluated.

The evaluation of a repetition then consists of continuously iterating the following steps:

- If the direction part is present as its first alternative, the value of the loop variable is compared with the value evaluated for the direction part. If it is greater, the evaluation of the repetition is finished.

  If the direction part is present as its second alternative, the value of the loop variable is compared with the value evaluated for the direction part. If it is less, the evaluation of the repetition is finished.

- If present, the while part, is evaluated. If it yields the value FALSE, the evaluation of the repetition is finished. If it yields the value TRUE or if the while part is absent, the evaluation of the repetition continues.

- The paragraph is evaluated.

- If present, the until part, is evaluated. If it yields the value TRUE, the evaluation of the repetition is finished. If it yields the value FALSE or if the until part is absent, the evaluation of the repetition continues.

- If the direction part is absent or if the direction part is present as its first alternative, the value of the loop variable is incremented.

  If the direction part is present as its second alternative, the value of the loop variable is decremented.

Symbols

The following symbols are added to the language

`for symbol`	`FOR`
`from symbol`	`FROM`
`upto symbol`	`UPTO`
`downto symbol`	`DOWNTO`
`until symbol`	`UNTIL`

A small set of test programs can be obtained by unpacking the file `ftp.cs.kun.nl://pub/cdl3/ex6.tgz` in your MINI working directory.