

05 Herhalingen

In computerprogramma's is vaak sprake van een herhaling van (bijna) dezelfde reeks van opdrachten.

repeat ... until;

Een voorbeeld bij (Lazarus) Free Pascal is:

```
program herhaling_met_repeat_until;

var teller : integer;

begin
teller:=0;
repeat
  teller:=teller+1;
  <opdracht 1>;
  <opdracht 2>
  ..
  <opdracht n>;
until teller=10;
readln()
end.
```

De opdrachten tussen `repeat` en `until` worden 10 keer herhaald.

`repeat ... until` werkt als volgt

1. de opdrachten tussen `repeat` en `until` worden uitgevoerd.
2. Vervolgens wordt nagegaan of aan de voorwaarde na `until` (in ons voorbeeld `teller = 10`) wordt voldaan.
3. Zo nee, dan worden teruggegaan naar stap 1
4. Zo ja, dan wordt doorgedaan met de opdracht na `until`.

Let op! Wordt nooit aan de voorwaarden voldaan, dan zal het programma steeds weer de opdrachten tussen `repeat` en `until` blijven uitvoeren. Het programma eindigt nooit, tenzij het geheugen van de computer zo vol raakt dat het besturingssysteem ingrijpt en het programma beëindigt.

TRUE of FALSE

`teller = 10;` (uitgesproken als teller is 10)

en

`teller:= 10;` (uitgesproken als teller wordt 10)

verschillen. De laatste is een toekeningsopdracht: de variabele `teller` krijgt de waarde 10. De eerste heeft een waarde: "waar" of "niet waar". Of in (Lazarus) Free Pascal `TRUE` of `FALSE`.

Andere voorbeelden van TRUE of FALSE uitdrukkingen zijn:

```
teller > 10;
```

(als de waarde van teller groter is dan 10 dan is de uitdrukking TRUE, en anders FALSE)

```
teller >=10;
```

(als de waarde van teller groter dan of gelijk is aan 10, dan is de uitdrukking TRUE, en anders FALSE)

```
teller <=10;
```

(als de waarde van teller kleiner dan of gelijk is aan 10, dan is de uitdrukking TRUE, en anders FALSE)

TRUE and FALSE noemen we Booleans (naar de wiskundige Boole).

We kunnen ook een variabele declareren van het type Boolean.

Bijvoorbeeld:

```
Var  
  teller : integer;  
  trueorfalse : boolean;
```

en verder in het programma kan voorkomen:

```
trueorfalse:=TRUE;
```

of

```
trueorfalse:=FALSE;
```

Een variabele van het type Boolean kan maar twee waarden hebben: TRUE of FALSE.

Aangezien: `teller>10` TRUE of FALSE is, kan men ook de opdracht geven:

```
trueorfalse:=teller>10;
```

Booleans kunnen ook samengesteld zijn:

```
teller>=2 AND teller<=10
```

for ... to ... do ...;

Er is nog een opdracht in (Lazarus) Free Pascal waarmee het mogelijk is één of meerdere opdrachten te herhalen.

Voorbeeld:

```
for teller:=1 to 5 do writeln(teller);
```

Deze opdracht heeft als output:

```
1
2
3
4
5
```

of algemener:

```
for teller:=1 to 5 do <opdracht>;
```

of nog algemener:

```
for variabele:= expressie-1 to expressie-2 do <opdracht>;
```

Er is een kleine aanpassing nodig om meerdere opdrachten te herhalen:

Voorbeeld:

```
for teller:=1 to 5 do
begin
write('teller = ');
writeln(teller);
end;
```

of algemener:

```
for teller:=1 to 5 do
begin
<opdracht 1>;
<opdracht 2>;
...
<opdracht n>;
end;
```

```

while <voorwaarde> do
  begin
    <opdracht 1>;
    <opdracht 2>;
    ...
    <opdracht n>;
  end;

```

Er is nog een derde constructie die men kan gebruiken om één of een rijtje opdrachten een aantal keren uit te voeren.

Een voorbeeld:

```

program herhaling_met_while;

var teller : integer;

begin
  teller:=0;
  while teller <= 10 do
    begin
      writeln (teller);
      teller:=teller+1;
    end;
  readln()
end.

```

output:

```

1
2
3
4
5
6
7
8
9
10

```

Opmerking:

In dit programma wordt begonnen met de voorwaarde. Het kan zijn dat de voorwaarde (gelijk aan) FALSE is. In dat geval wordt de rij opdrachten tussen begin en end niet één keer uitgevoerd.

Bij repeat ... until... wordt de opdracht (of de rij met opdrachten tenminste één keer uitgevoerd.