

04 Variabelen

Variabelen voor gehele en variabelen voor gebroken getallen

In Pascal programma's kan gebruik gemaakt worden van variabelen. Een variabele is op te vatten als een plek in het geheugen van de computer, waarin een getal of een zin kan worden opgeslagen. Het maakt voor de grootte van die plek uit wat erin wordt opgeslagen. Daarom moet eerst in het programma worden aangegeven wat de naam van de plek zal zijn en wat erin wordt opgeslagen.

Voorbeeld:

Met

```
var a : integer;
```

bedoelen we dat er een plek is vastgelegd met de naam a, en dat daarin een geheel getal zal kunnen worden opgeslagen. Dit heet het declareren van a als een integer (een geheel getal).

Met

```
a := 3;
```

(uit te spreken als "a wordt 3")

bedoelen we dat op de plek met de naam a, (die eerder is vastgelegd) het getal 3 wordt opgeslagen. Dit heet "a heeft de waarde 3". Je kunt met a rekenen als met 3.

Je kunt de waarde van een variabele aanpassen:

```
a := 5;
```

```
a := 7;
```

```
a := a + 8;
```

(a wordt 5, a wordt 7 en a wordt wat a was vermeerderd met 8) (In dit geval, was a gelijk aan 8 en wordt nu 15).

In een Pascal programma moet het declareren van een variabele vooraf gaan aan het

begin

end.

blok.

Dus:

```
program <naam van het programma>;  
  
var a: integer;  
  
begin  
a:=5;  
writeln('a = ', a);  
a:=7;  
writeln('a = ', a);  
a:=a+8;  
writeln('a = ', a);  
readln();  
end.
```

Er kunnen meerdere variabelen in één regel worden gedeclareerd.

```
var a,b,c :integer;
```

betekent dat a, b en c variabelen zijn met geheel getal als waarde.

Je kunt ook gebroken getallen declareren:

```
var x,y,z : real;
```

en tegelijkertijd beide soorten:

```
var a, b, c : integer;  
    x, y, z : real;
```

Namen voor variabelen

Variabelen kunnen ook langere namen hebben zoals `teller` en `huisnummer` of `aantal_kinderen`.

Maar altijd één woord, zonder bijzondere tekens zoals `*`, `/`, `-` en `+`. En de namen mogen ook niet overeenkomen met namen van bijzondere woorden in Pascal zoals `program`, `begin`, `writeln` en `end`.

Variabelen voor woorden en zinnen

Er is ook een variabel type dat geschikt is om er woorden en zinnen in op te slaan. Die declareer je door:

```
var <naam> : string;
```

Voorbeeld:

```
program een_zin;
  var s, t : string;

begin
  s:= "mijn eerste zin";
  t:= "mijn tweede zin";
  writeln (s, 'en ', t);
  readln()
end.
```

Opdracht 1

Schrijf een programma dat de tafel van 23 afdrukt.

(1 * 23 = 23

2* 23 = 46

...

10* 23= 230)

Gebruik daarbij een variabele teller van het soort integer, de opdracht

```
teller := 1
```

en de opdracht

```
teller := teller + 1;
```

Opdracht 2

Je weet dat de omtrek van een cirkel benaderd wordt door de formule $2\pi r$ waarbij $\pi = 3,1415962$ en r de straal van de cirkel is.

Schrijf een programma met de naam omtrek en variabele r van het soort integer.

Zorg ervoor dat wordt afgedrukt:

Bij $r = 1$ is de omtrek ... (waarbij in plaats van de ... de bijbehorende omtrek wordt ingevuld)

Bij $r = 2$ is de omtrek ... (idem)

Bij $r = 3$ is de omtrek ... (idem)

Bij $r = 4$ is de omtrek ... (idem)

Opdracht 3

Schrijf een programma dat je naam op slaat in de variabele `mijn_naam`, je adres op slaat in `mijn_adres`, en je postcode op slaat in `mijn_postcode` en je woonplaats op slaat in `mijn_woonplaats`.

Vervolgens laat je door de variabelen te gebruiken je naam, adres, postcode en woonplaats op drie regels afdrukken (postcode en woonplaats op één regel en tussen postcode en woonplaats twee spaties).