

## 2. Spelen met het scherm

### 2.1 Schermindeling

In dit boek werken we veel met tekstschermen. Daarbij wordt het scherm van de monitor vaak ingedeeld in tachtig kolommen en vijftientig rijen. Een andere veelgebruikte instelling is tachtig kolommen en vijftig rijen.

```
+---+---+---+---+---+---+---+---+---+
X/Y=  |1/1|2|3|4|5|6|7|8|9|10|..
+---+---+---+---+---+---+---+---+---+
|2| | | | | |
+---+---+---+---+---+---+---+---+
|3| | | | |
+---+---+---+---+---+---+
|4| | |
+---+---+---+---+---+
|5| |
+---+---+
|6
|..
```

Bovenstaande tekening geeft de indeling van de linker bovenhoek van het scherm weer. Alle kolommen zijn in vakjes verdeeld en genummerd van 1 tot en met 80. In elk vakje past één letter- of leesteken. Iedere regel loopt van 1 tot en met 25.

Er wordt in dit geval ook wel gesproken van schermcoördinaten. Hierbij wordt de nummering van de horizontale vakjes op de regel de X-coördinaat genoemd en de verticale nummering van de regels de Y-coördinaat. Als we met behulp van X en Y een positie op het scherm bepalen, staat "20,10" voor de twintigste positie op de tiende regel.

### 2.2 Een eerste programma

Het wordt tijd dat we een eerste programma gaan maken. Het programma is heel eenvoudig. We starten eerst Free Pascal zoals is uitgelegd in hoofdstuk 1. We komen nu in de ontwikkelomgeving van Free Pascal terecht, ook wel de IDE genoemd. Deze IDE ziet eruit als een werkblad met bovenaan een menu en onderaan een statusbalk. We kunnen via een druk op de functietoets F10 de keuzebalk in het menu activeren. Met de pijltjestoetsen kunnen we deze keuzebalk heen en weer bewegen. Als we op de Enter-toets drukken, gaat er een menuvenstertje open. Binnen dit venstertje kunnen we de keuzebalk weer met de pijltjestoetsen op en neer bewegen.

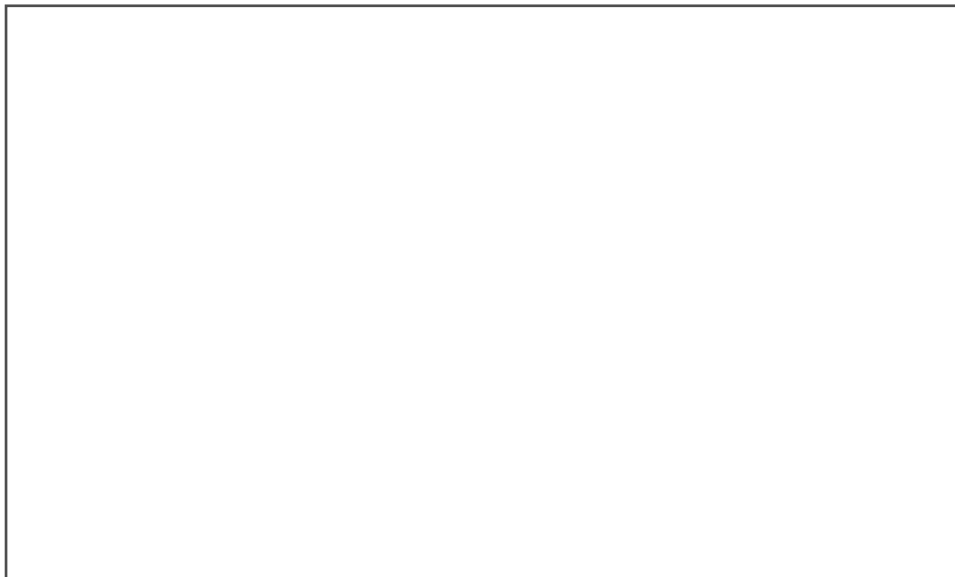
De verschillende mogelijkheden van het menu hebben ook nog een anders gekleurde letter in hun tekst. Bij monochrome schermen is dit een oplichtende letter. Met de toetscombinatie Alt-letter activeer je deze keuzemogelijkheid rechtstreeks. Als je over een muis beschikt, kun je de keuzemogelijkheid activeren door de muiscursor op het gekozen onderdeel uit het menu te zetten en de linker muisknop in te drukken.

Laten we voordat we beginnen met programmeren eerst even wat anders doen, de IDE aanpassen aan je wensen. Veel mensen willen grootte van de venster van de IDE bijvoorbeeld wat anders hebben.



*Fout! Bladwijzer niet gedefinieerd.***Afbeelding 3**

Met de toetscombinatie Alt-O kiezen we nu voor «*Options*». Vervolgens kiezen we voor «*Environment*» en dan «*Preferences*» en krijgen dan een scherm te zien waarmee we de IDE kunnen aanpassen aan onze wensen. Klik op «*Video mode*». Er klapt dan een lijstje uit waar je kunt kiezen. Kies bijvoorbeeld 80×25 als je liever een klein scherm hebt of 80×50 als je liever met een groot venster werkt.



*Fout! Bladwijzer niet gedefinieerd.***Afbeelding 4**

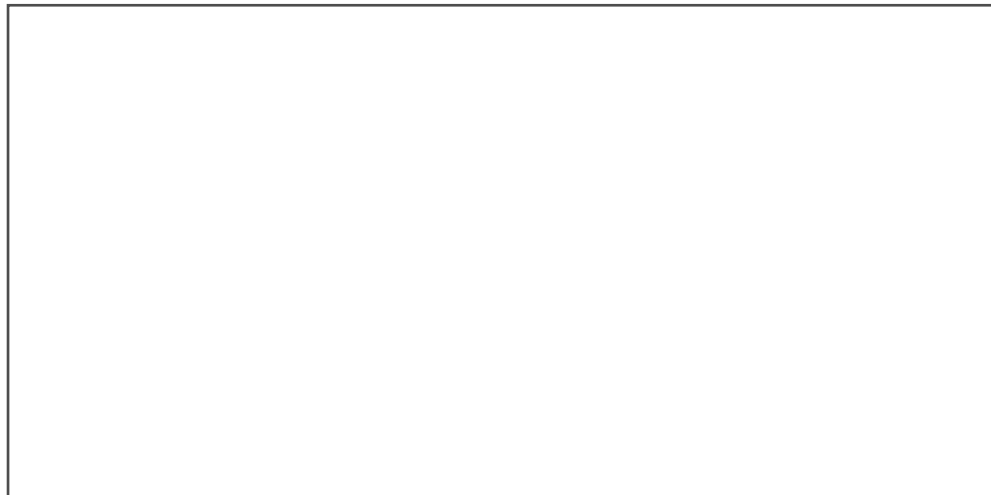
Na op de button OK gedrukt te hebben (of door net zolang de Tab-toets in te drukken tot OK actief is en dan op de Enter-toets te drukken), kiezen we in hetzelfde Options-menuvenster voor de optie «*Save*». Nu worden de zojuist gemaakte instellingen in het bestand FP.INI bewaard.

Als we een volgende keer de IDE gebruiken, hoeven we deze instellingen niet opnieuw in te tikken, omdat Free Pascal bij het opstarten de benodigde gegevens uit het bestand FP.INI haalt. Als je voor de keuzemogelijkheid «*SaveAs*» kiest, kun je een bestand met instellingen onder een andere naam opslaan. Bij het starten van Free Pascal zoekt het programma altijd naar een bestand FP.INI. Door achtereenvolgens te

kiezen voor «Options» en «Open» kun je instellingen inlezen die je een andere naam gegeven hebt. Het voordeel van deze mogelijkheid is, dat je voor verschillende programmeerklussen zonodig ook verschillende instellingen kunt hanteren.

We kiezen nu in het menu voor «File» en «New». Hierdoor wordt de Free Pascal-editor actief. Hier tikken we onze programma's in. De manier waarop de editor bediend wordt, is in de bijlagen van dit boek opgenomen. We typen het volgende programma in:

```
PROGRAM SCHERM_1;  
USES CRT;  
  
BEGIN  
  ClrScr;  
  GotoXY(20,10);  
  Write('Ik ben bezig Pascal te leren');  
  Readln  
END.
```



*Fout! Bladwijzer niet gedefinieerd.***Afbeelding 4**

Als we nu in het menu voor «Run» en nogmaals «Run» kiezen, zal het programma uitgevoerd worden en zal er op het scherm verschijnen:

In een Pascal-programma staat altijd op de eerste regel PROGRAM met daarachter de naam die je aan het programma geeft, gevolgd door een puntkomma.

Op de volgende regel(s) geef je op welke unit(s) uit de programmabibliotheek je gebruikt. In het programma SCHERM\_1 stelt de regel "USES CRT;" de unit CRT open voor gebruik. In deze unit staan de procedures en functies voor de schermafwikkeling.

Het hoofdprogramma staat tussen BEGIN en END. De procedure ClrScr die hier aangeroepen wordt is afkomstig uit de unit CRT. ClrScr zorgt ervoor dat het scherm schoongeveegd wordt. Ook de procedure GotoXY is afkomstig uit de unit CRT. De procedure GotoXY plaatst de cursor op positie X van regel Y.

De procedure Write is niet opgenomen in de unit CRT, maar in de unit SYSTEM. Deze unit hoeft je nooit in de USES-regel op te nemen, omdat hij altijd automatisch gebruikt wordt door een Free Pascal-programma. De procedure Write zet tekst op het scherm, beginnend op de positie van de cursor.

Als laatste opdracht wordt een "Readln" gegeven. Dit is een afkorting van "Read line", ofwel "Lees regel". De procedure Readln, die ook opgenomen is in de unit SYSTEM, is ondermeer bedoeld om gegevens vanaf het toetsenbord in te lezen. Gegevens die je invoert via het toetsenbord, worden afgesloten met een druk op de Enter-toets. In dit geval wordt Readln als een pauze gebruikt. Zou je deze opdracht niet opnemen, dan zou de tekst maar een fractie van een seconde op het beeldscherm te zien zijn. Haal de opdracht maar eens weg, dan kun je zien wat er gebeurt.

Het inspringen na BEGIN met drie spaties en het vóór END weer terugspringen, is algemeen gebruikelijk in de vormgeving van een Pascal-programma. De compiler trekt zich van de vorm niets aan, maar deze manier van schrijven bevordert de leesbaarheid van een programma. Ook het gebruik van hoofdletters dient ertoe een programma overzichtelijk te houden. De compiler maakt geen onderscheid tussen "clrscr" en "ClrScr", maar door de laatste manier van schrijven geven we duidelijk aan dat ClrScr kennelijk een afkorting is van twee woorden: "Clear" en "Screen".

Wil je dit programma bewaren, dan kies je in het menu van de IDE voor: «File» en vervolgens voor «Save». Je krijgt dan de mogelijkheid om het programma een naam te geven. Denk eraan dat de naam niet meer dan acht lettertekens mag bevatten. Geef je toch een langere naam op, dan vallen de letters die teveel zijn weg. Als je het programma later weer nodig hebt, kies je voor «File» en voor «Open». Je kunt dan de naam intikken op de invoerregel en op Enter drukken, of op de Tab-toets drukken, de keuzebalk op het gewenste programma zetten en op de Enter-toets drukken.

Je kunt in de IDE verschillende tekstbestanden tegelijk inlezen. Met de F6-toets kun je door de verschillende schermen bladeren. Het scherm van de monitor fungeert als het ware als een bureaublad (desktop) waar een stapeltje papier op ligt. Met de keuzen «File» en «Save all» worden alle bestanden die zich op de desktop bevinden, weggeschreven naar schijf.

## 2.3 Gebruik van kleuren

Als je de beschikking hebt over een computer met een kleurenmonitor, dan kun je ook kleuren in je programma's gebruiken. De beschikbare kleuren hebben alle een nummer:

Nummer:	Kleur:	Nummer:	Kleur:
0	zwart	8	donkergrijs
1	blauw	9	lichtblauw
2	groen	10	lichtgroen
3	cyaan	11	lichtcyaan
4	rood	12	lichtrood
5	magenta	13	lichtmagenta
6	bruin	14	geel
7	lichtgrijs	15	wit

Als je deze kleurentabel goed bekijkt, zie je dat er eigenlijk van maar acht kleuren sprake is. Deze kleuren zijn genummerd van nul tot en met zeven. Als je acht optelt bij een van deze kleuren, wordt dezelfde kleur in een lichtere tint op het scherm weergegeven. Als je zestien optelt bij een kleur, wordt de betreffende kleur knipperend weergegeven als het de voorgrondkleur betreft. Wil je bijvoorbeeld een knipperende groene letter, dan neem je kleur 2 en tel je er 16 bij op. Als kleur geef je dan het nummer 18 op.

We gaan er nu voor zorgen dat het programma SCHERM\_1 de tekst geel weergeeft op een blauwe achtergrond. Het nieuwe programma noemen we SCHERM\_2:

```
PROGRAM SCHERM_2;
USES CRT;
```

```

BEGIN
    TextBackground(1);
    TextColor(14);
    ClrScr;
    GotoXY(20,10);
    Write('Ik ben bezig Pascal te leren');
    Readln
END.

```

Aan ons nieuwe programmaatje hebben we slechts twee opdrachten toegevoegd: TextBackground en TextColor. Tussen de haakjes staat het nummer van de gewenste kleur. TextBackground(1) zet de achtergrondkleur op blauw (nummer 1 volgens de tabel). De opdracht TextColor(14) levert volgens de tabel een tekst in geel op.

## 2.4 Windows

We hebben gezien dat het tekstschermbereik verdeeld is in 25 x 80 posities en dat we elke positie met de opdracht GotoXY kunnen bereiken. De unit CRT kent ook nog een procedure Window. Met deze procedure kun je binnen het scherm een kleiner scherm definiëren. Het programma SCHERM\_3 laat zien hoe zo'n window gemaakt wordt:

```

PROGRAM SCHERM_3;
USES CRT;

BEGIN
    TextBackground(1);
    TextColor(14);
    ClrScr;
    Window(15,7,65,13);
    TextBackground(4);
    ClrScr;
    GotoXY(5,3);
    Write('Ik ben bezig Pascal te leren');
    Readln
END.

```

Na het definiëren van de kleuren van de achtergrond en de tekst, en het aanroepen van de procedure ClrScr, wordt de procedure Window aangeroepen. Tussen de haakjes worden schermcoördinaten meegegeven in de vorm: X1, Y1, X2, Y2. De coördinaten X1 en Y1 wijzen naar de linker bovenhoek van het scherm (positie 15,7). De coördinaten X2 en Y2 wijzen naar de rechter benedenhoek (positie 65,13). Samen vormen X1, Y1, X2, Y2 een rechthoek. Deze rechthoek kan met TextBackground en ClrScr ingekleurd worden. In dit geval kiezen we als invulkleur rood (nummer 4). We hebben nu een rood venster op een blauwe achtergrond, met daarin een gele tekst. Als de procedure Window aangeroepen is, heb je dus niet meer de beschikking over een scherm van 80 x 25 posities. Het nieuwe scherm bevat zes regels van vijftig posities. Dit is berekend door X1 van X2 af te trekken en Y1 van Y2 af te trekken.

## 2.5 Video-instellingen

Om de tekst in heldere of doffe kleuren op het scherm te krijgen, beschik je in Free Pascal over een drietal instellingen. Het programma SCHERM\_4 laat zien hoe je deze gebruikt:

```

PROGRAM SCHERM_4;
USES CRT;

BEGIN
    TextBackground(1);
    TextColor(7);
    ClrScr;
    HighVideo;
    Writeln('Dit is een regel met HighVideo');
    NormVideo;
    Writeln('Deze regel wordt weer gewoon afgedrukt');
    Write('                Druk op Enter');
    GotoXY(WhereX,WhereY);
    InsLine;
    Readln;
    TextColor(14);
    LowVideo;
    DelLine;
    Writeln('Nu is LowVideo ingesteld');
    NormVideo;
    Write
    ('Alles is nu als bij start. Druk op Enter');
    Readln
END.

```

In het programma SCHERM\_4 wordt de achtergrondkleur op blauw gezet en de tekstkleur op lichtgrijs. Daarna wordt met de procedure ClrScr het scherm schoongemaakt.

Op de volgende regel wordt de procedure HighVideo aangeroepen. Voor alle schrijfoopdrachten naar het scherm wordt nu bij de kleurwaarde van de tekst 8 opgeteld. Op die manier verandert lichtgrijs in helderwit. Als de procedure NormVideo aangeroepen wordt, verandert de tekstkleur naar de kleurencombinatie van voor- en achtergrond zoals die bij de start van het programma is ingesteld. Zoals we de procedure HighVideo kunnen gebruiken om van doffe kleuren heldere kleuren te maken, zo kunnen we de procedure LowVideo gebruiken om van heldere kleuren doffe kleuren te maken.

In het programma SCHERM\_4 zijn nog een paar extra faciliteiten uit de unit CRT opgenomen. Met een aanroep van de procedure WhereX krijgen we de X-coördinaat van de plaats waar de cursor staat terug. De procedure WhereY geeft de Y-coördinaat van de cursor terug. De procedure InsLine zet op de regel waar de cursor zich bevindt een nieuwe, lege regel en de procedure DelLine verwijdert een hele regel op de plaats waar de cursor zich op dat moment bevindt.

## 2.6 Opgaven

1. Maak een lichtgrijs scherm met in het midden een blauw venster van 40×10 posities. Zet midden in dit blauwe venster de tekst: "Een eerste programma".

-----

36

36

36