D.B.Baarda / A.van Londen

Leren omgaan met MSX-computers

BASIC voor jonge computergebruikers



Kluwer Technische Boeken

D.B.Baarda/A.van Londen
Leren omgaan met MSX-computers

Leren omgaan
met
MSX-computers

Leren omgaan met **MSX-computers**

BASIC voor jonge computergebruikers

Drs. D. B. Baarda
en
Drs. A. van Londen

SONY
PHILIPS
GOLDSTAR
SPECTRAVIDEO
PIONEER
NATIONAL
JVC
PANASONIC
CANON
FUJITSU
HITACHI
MITSUBISHI
SANYO
TOSHIBA
YAMAHA
YASHICA
DAEWOO
en andere MSX-computers



Kluwer Technische Boeken B.V. - Deventer - Antwerpen

Omslag en illustraties: W. Niessink

ISBN 90 201 1875 7
D/1986/0108/176

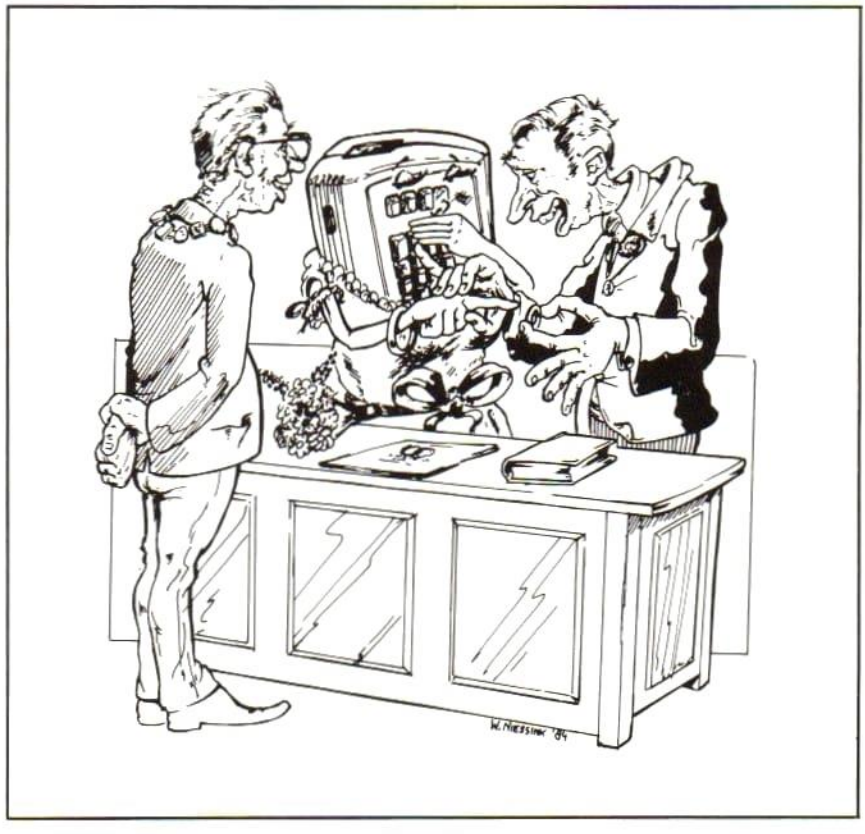
© 1986 Kluwer Technische Boeken B.V. - Deventer

1e druk 1986

Niets uit deze uitgave mag worden verveelvuldigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze ook, zonder voorafgaande schriftelijke toestemming van de uitgever.

No part of this book may be reproduced in any form, by print, photoprint, microfilm or any other means without written permission from the publisher.

Ondanks alle aan de samenstelling van de tekst bestede zorg, kan noch de redactie noch de uitgever aansprakelijkheid aanvaarden voor eventuele schade, die zou kunnen voortvloeien uit enige fout, die in deze uitgave zou kunnen voorkomen.



Met dank aan:
Arleen
Andre
Christian
Jonathan
Mariska
Oscar
Raf
Ruth

Programma's zijn geschikt voor MSX BASIC
en getest op de SONY HIT BIT

INHOUD

Waarover gaat dit boek?

Deel 1
WAT KAN EEN COMPUTER?
MSX-standaard 13
Tekenen met de computer 14
Schrijven met de computer. . . . 19
Rekenen met de computer 21
Nog meer tekenmogelijkheden. . 27
Kleuren en muziek maken met de computer 30
Kleuren met de computer 30
Muziek. 33

Deel 2
PROGRAMMEREN IN BASIC
Klaar om te programmeren? . . . 39
Regelnummer. 39
RETURN 40
END. 41
Opdrachten 41
NEW. 42
LIST. 42
RUN. 43
STOP 44
CTRL en STOP 44
RESET 44

LEREN PROGRAMMEREN
Inleiding. 49
PRINT. 51
INPUT. 57
GOTO 61
GOSUB en RETURN. 65
REM. 71
IF ... THEN 74
FOR ... NEXT 78
READ ... DATA 83
RND. 89
Fouten in je programma's 93

Deel 3
PROGRAMMA'S
Rekenen met de computer 99
Toetsenbord. 104
Kleurendemonstratie 105
Rising Sun 105
Toonladders. 105
Old Mac Donald. 106
Happy Birthday. 106
Old Mac 106
De computer is verliefd. 107
Grootste Gemene Deler. 107
Uitrollende dobbelsteen 108
Twee dobbelstenen 108
Getallen omkeren. 109
Namen omkeren 109
Gedachtenlezen. 110
Priemgetallen 111
Nonsens Generator 111
Huiswerkhulp. 112
Telefoonboek 113
Rekenkundige tafels 113
Overhoren tafels 114

Bijlage 1
DE CASSETTERECORDER
Bewaren. 117
Laden 118

Bijlage 2
ANTWOORDEN OP DE VRAGEN

WAAROVER GAAT DIT BOEK?

Het boek dat je nu begint te lezen gaat over het omgaan met computers en het is speciaal voor kinderen geschreven. (Dit betekent niet dat een volwassene er niks uit kan leren!) We leren je om zelf de MSX-computers te gebruiken. Als het goed gaat heb je er geen volwassene bij nodig.
In dit boek kun je BASIC leren. Dit is een computertaal. Daarover gaat het tweede deel van dit boek. Als je al kunt omgaan met het toetsenbord, kun je beginnen met het hoofdstuk KLAAR OM TE PROGRAMMEREN? Anders kun je beter vooraan beginnen en in ieder geval TEKENEN en SCHRIJVEN MET DE COMPUTER lezen, voordat je aan het tweede deel begint.
Het eerste deel van het boek behandelt hoe je kunt tekenen en schrijven, kleuren en geluid maken met de computer. Zo gebruik je de computer als een soort schrijf- of tekenpapier, als kleurdoos of als muziekinstrument.
We leren je welke toetsen je hiervoor moet indrukken. Het verschil met papier is alleen dat je tekening of verhaal weg is als je de computer uitzet en dat is soms jammer.
In het tweede deel kun je leren hoe je de computer ook voor andere zaken kunt gebruiken. Daarbij kun je denken aan het onthouden en bewaren van dingen (bijvoorbeeld het opstel dat je wilt gaan schrijven of de tekeningen die je gaat maken, of het muziekstukje).
En hij kan ook snel karweitjes voor je uitvoeren waar je zelf veel meer tijd voor nodig hebt, zoals een moeilijke som uitrekenen, duizend strafregels schrijven, je huiswerk-hulp zijn en spelletjes ontwerpen.
Maar dit kan de computer niet zonder jouw hulp. Jij moet hem vertellen wat je wilt dat hij doet. Je schrijft de computer daarvoor een soort 'brief' via het

toetsenbord. Zo'n 'brief' noem je een 'programma'. Je kunt niet beginnen met 'Beste computer, luister eens...', want dat snapt hij niet. Je moet het programma in zijn eigen taal schrijven, een computertaal. De computertaal die wij je leren heet 'BASIC'.
BASIC is een schrijftaal. Er is niemand die BASIC spreekt. Er worden alleen computerprogramma's in geschreven. Dit leer je in het tweede deel:
PROGRAMMEREN IN BASIC.
In het derde deel vind je voorbeelden van programma's die al gemaakt zijn. Je kunt ze overtypen. Als je daarna RUN typt en vervolgens op 

drukt, doet de computer wat er in het programma staat. Sommige van deze programma's worden in deel 1 beschreven.

DEEL 1

WAT KAN EEN COMPUTER?

MSX-standaard

Een aantal computerfabrikanten hebben (gelukkig) afspraken gemaakt over hoe hun computers er uit zullen zien en welke computertaal ze zullen gebruiken. Ze noemen dat de MSX-standaard en de letters MSX staan voor MicroSoft eXtended. Microsoft is een hele grote firma die onder andere de computertaal BASIC maakt.

Waarom een standaard?
Als je een computer gekocht hebt, zul je snel merken dat je moeilijk spelletjes kunt gebruiken van je vriend, omdat die een andere computer heeft. Het werkt gewoon vaak niet.
Dat is jammer, want er bestaan heel veel schitterende programma's voor computers. Prachtige spelletjes, uitstekende tekstverwerkers, mooi rekentuig en ga zo maar door. Maar deze programma's zijn meestal voor één soort computer geschikt en niet voor andere computers. Dit behoort nu tot het verleden. De MSX-standaard houdt een aantal afspraken in waar computers aan moeten voldoen. De afspraken over hoe de computers en de spelletjes er uit moeten zien zijn voor alle MSX-computers hetzelfde. Hierdoor zijn de benodigde onderdelen, zoals de programma's, uitwisselbaar geworden (compatible heet dat).
Wanneer je nu zelf met een mooi programma bezig bent en je vraagt een vriend om iets voor je uit te zoeken, kan hij het op zijn eigen computer rustig bekijken.
Toch bestaan er per computer nog kleine verschillen die meestal niet belangrijk zijn. Wij hebben alles getest op de SONY HIT BIT. Alle voorbeelden zijn van de HIT BIT afkomstig.

TEKENEN MET DE COMPUTER

Als je de computer aanzet zie je als het goed is:



Bij sommige MSX-computers moet je wel eerst duidelijk maken dat je met BASIC wilt werken.

- Je ziet niet zoals bij een gewoon TV-toestel een testbeeld. Wat je ziet is een TV-scherm waar de computer op kan tekenen of schrijven en waar jij op kunt tekenen of schrijven via het toetsenbord. Het lijkt dus gewoon op een blanco velletje papier. Er staat wel een blokje op. Daarmee geeft de computer aan dat jij aan de beurt bent om iets op het scherm te zetten. Als de computer zelf iets gaat schrijven, verdwijnt dat blokje (in het Engels heet die de cursor). In dit deel van het boek zullen we vooral kijken wat jij op het scherm kunt tekenen en schrijven. Hoe je er voor zorgt dat de computer iets gaat tekenen of schrijven leer je in het tweede deel. Maar het blokje staat er, dus jij kunt iets typen op de plek waar het blokje staat.
  Druk maar eens op x.
  Heb je het gedaan? Dan staat er als het goed is een x op het TV-scherm.
  Druk nog eens op x. Je ziet dat er nu nog een x naast staat. Het blokje schuift iedere keer een plekje naar rechts als je iets intypt.
  Het blokje is een soort stuur. Druk eens 34 keer op x.

Als het goed is, staat het blokje nu aan de rand en heb je een hele rij x-tekens. Klopt het? Nee, zet dan de computer uit en weer aan en begin nog eens te lezen bij •. Heb je een hele rij x-tekens en staat het blokje aan de rand, typ dan nog eens x. Het blokje staat nu weer helemaal aan de linkerkant, onder wat je net getypt hebt. Als een regel vol is, gaat hij namelijk naar het begin van de volgende regel.

•• Je kunt het blokje ook besturen met de knoppen ◀ ▶ ▲ en ▼. Kun je ze vinden?
Ze zitten allemaal aan de rechterkant.

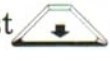


Druk nu op . Je ziet dat het blokje naar rechts schuift. Breng het maar eens naar de zijkant van het scherm. Je zult merken dat als je de toets ingedrukt houdt, de cursor heel snel gaat. Probeer maar eens.
Druk nu eens op . Nu gaat het blokje naar beneden. Probeer eens in de onderste hoek te komen. Nu moet je weer omhoog. Daarvoor moet je  indrukken. En wanneer je naar links wilt gebruik je . Kijk nu eens of je in alle hoeken van het TV-scherm kunt komen. Lukt het niet, lees dan nog eens vanaf • •.
We gaan nu tekenen, waarbij het blokje het stuur voor het potlood is.

••• Als je op [HOME] drukt zul je zien dat de cursor naar linksboven springt. Op sommige computers is dat de

CLS-toets. Als je op SHIFT en HOME drukt krijg je een heel schoon scherm. SHIFT ingedrukt houden en dan op HOME drukken.
Kun je de toetsen vinden?



Onthoud het maar goed. Dus wanneer je op SHIFT en HOME drukt, is het TV-scherm schoon.
Wij beginnen met een vierkant in de hoek van het TV-scherm. Typ eerst vijf keer x. Nu moet je proberen onder de laatste x te komen.
Dus eerst  . Je gaat nu één regel naar beneden.

Dan ga je met  één stapje terug.
Als het goed is sta je nu onder de laatste x.

Staat het blokje onder de laatste x?
Nee, begin dan nog eens bij • • •.
Probeer nu dit vierkant te maken.

```
* * * * *
*       *
*       *
*       *
* * * * *
```

• • • • **UITVEGEN**

Het is erg moeilijk en je hebt misschien wel een fout gemaakt die je wilt uitvegen. Ga dan met het blokje naar wat je wilt uitvegen en druk op die hele lange toets onderaan. De spatiebalk noemt men die.



Veeg nu eens de onderste lijn van het vierkant uit. Gelukt? Nee, lees dan nog eens • • • •. Je kunt nu van alles tekenen. Wij hebben een aantal voorbeeldjes gemaakt. Maar je kunt ook allerlei dingen zelf bedenken. Veeg eerst het TV-scherm maar schoon. Weet je niet meer hoe dat moet, kijk dan bij • • •.

opgaven

1 Maak eerst het scherm schoon. En teken dan van x'jes het volgende huisje.

```
    *
  *   *
*       *
*       *
*       *
* * * * *
```

2 Maak het scherm weer schoon. En teken dan weer van x'jes het beest dat je op blz. 18 ziet staan. Het is een soort computerhond.



3 Maak het scherm schoon. En teken een boom, boot, auto of iets wat je zelf bedenkt.

SCHRIJVEN MET DE COMPUTER

Het zal je niet verbazen dat je ook kunt schrijven met de computer. Hij lijkt immers erg veel op een typemachine. Je kunt hem dan ook gebruiken om te typen. Op de MSX-computers zitten alle lettertoetsen op dezelfde plaats als bij een gewone typemachine. Je kunt er bijvoorbeeld je naam op typen.

- Typ maar eens oscar of jouw naam.
  Je ziet, het zijn alleen maar kleine letters. De computer kent ook wel hoofdletters.
  Druk maar eens op [CAP] en typ dan OSCAR. Als het goed is, dan staat er nu OSCAR. Als je [CAP] hebt ingedrukt kun je hoofdletters typen. Je zult zeggen ik heb hoofdletters en kleine letters door elkaar nodig. En dat kan ook. Als je [SHIFT] en bijvoorbeeld [A] tegelijk indrukt krijg je de hoofdletter A. Druk je alleen op [A] dan krijg je a op het TV-scherm. Je moet wel eerst de [CAP]-toets uitzetten.
  Druk eens vijf keer op [A]. Krijg je nu aaaaa? Als je dit niet hebt maar wel AAAAA, druk dan nog eens op [CAP] en druk vijf keer op [A]. Nu moet het kloppen.
  Druk eens vijf keer op [SHIFT] en [A] tegelijk. Nu krijg je AAAA. Lukt het niet ga dan nog eens terug naar •.
  Vooral op de cijfertoetsen zie je telkens boven het cijfer nog iets anders staan. Boven de 8 staat bijvoorbeeld * en boven 9 staat (. In programma's die je gaat maken kunnen deze tekens erg belangrijk zijn. Zoek ze maar eens op. Je kunt * krijgen door [SHIFT] en [8] tegelijk in te drukken.
  Ook het vraagteken kun je alleen door [SHIFT] en [/] samen in te drukken krijgen. Als je [SHIFT] vergeet krijg je /.
  Telkens wel de [SHIFT]-toets ingedrukt houden en dan pas op de andere drukken.
  Onderaan het toetsenbord vind je een hele lange toets, dit is de spatiebalk. Daar kun je spaties mee maken, druk er maar eens op. Zoals je ziet schuift de cursor (het

blokje ■) één plaats op, hij typt niets. Er wordt een soort van opening gemaakt, men noemt dit een spatie. Je vindt die bijvoorbeeld tussen twee woorden in.

Fouten verbeteren kun je met de cursorbesturing zoals wij hiervoor hebben uitgelegd. Het kan echter ook met de toets waar BS op staat. Druk er maar eens op. De cursor gaat nu één plekje terug. Typ eens FIETZ; ja met een Z.
Zoals je weet moet het FIETS zijn. Ga nu eens met BS terug en maak van de Z een S. Je hoeft alleen maar op de S te drukken, de Z veegt hij vanzelf uit.
Het valt waarschijnlijk niet mee om snel de verschillende letters te vinden. Wij hebben een programmaatje gemaakt dat je daarbij kan helpen. Het staat achterin en het heet 'TOETSENBORD'. Misschien kan iemand anders het even voor je intypen. Wij hebben het met opzet kort gehouden. Het programma geeft telkens een letter op en jij moet die dan zo snel mogelijk intypen. Aan het einde krijg je te zien hoeveel je er goed had.

opgaven

1 Typ eens: Ik leer al aardig typen. Goed zo!

Je moet zowel de hoofdletters als de kleine letters en ook het uitroepteken (!) typen.

2 Typ eens:

4 " 8 $ ( ) ?

3 Oefen met het programma 'TOETSENBORD'.

4 Maak eens een gedicht op de computer.

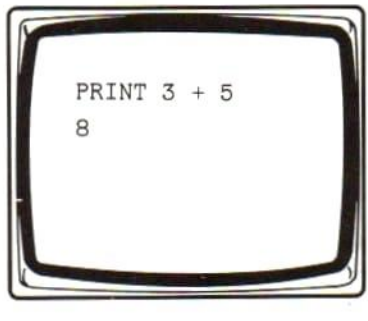
REKENEN MET DE COMPUTER

Computers werden vroeger alleen maar als rekenmachines gebruikt: rekenen kunnen ze veel sneller en nauwkeuriger dan mensen! Daarvoor zijn ze ook uitgevonden. Je zult zien hoe snel de computer je rekenhuiswerk kan maken.
Je hoeft alleen maar je som in te typen, en hij weet het antwoord al. Als je voor je som PRINT typt, dan schrijft hij het antwoord op het TV-scherm.
De computer gebruikt voor vermenigvuldigen en delen niet dezelfde tekens die jij op school leert. Hieronder komt een lijstje:

	BASIC wat de computer begrijpt:	*Op school leer je:*
Optellen	+	+
Aftrekken	–	–
Vermenigvuldigen	*	×
Delen	/	:

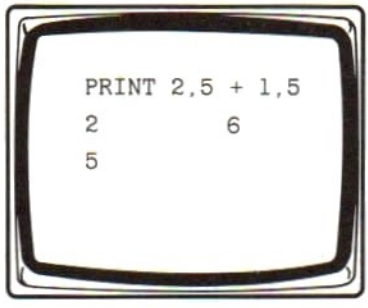
Als je de computer als rekenmachine wilt gebruiken, dan typ je PRINT en daarachter de getallen en het teken voor optellen, aftrekken of iets anders. Laten we een paar voorbeelden maken.

typ



– *druk op* RETURN

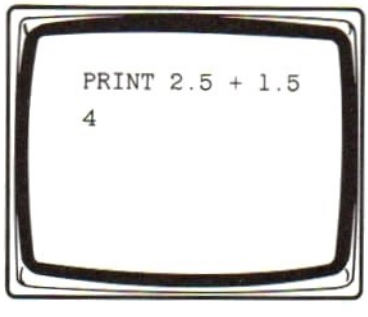
typ



– *druk op* RETURN

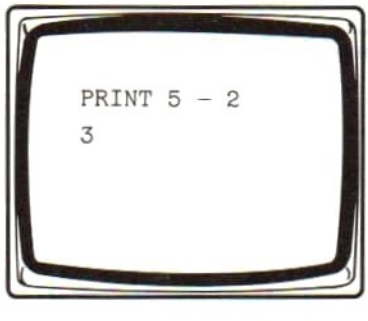
Hé, dat gaat fout. Je dacht dat er 4 zou komen en er staat nu 2 6 en 5. Hoe zou dat komen? De computer gebruikt de , (komma) voor iets anders dan waar wij hem nu voor nodig hebben. De komma betekent dat de volgende uitkomst een stukje verder op het scherm komt. Eerst de 2, dan een stukje verder 5 + 1 en weer een stukje verder 5. We hebben nu een . (punt) nodig. Dus onthouden: EEN . (PUNT) IN PLAATS VAN EEN , (KOMMA) midden in een getal.

typ



– *Let op een punt in plaats van een komma*

typ



– *Aftrekken*

typ

```
PRINT 2.5 - 1.5
1
```

– *Let op een punt in plaats van een komma*

typ

```
PRINT 3 * 5
15
```

– *Vermenigvuldigen. Let op een * in plaats van een* ×

typ

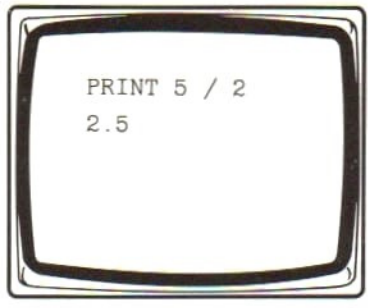
```
PRINT 2.5 * 1.5
3.75
```

typ

```
PRINT 6 / 2
3
```

– *Delen. Let op een / in plaats van :*

typ

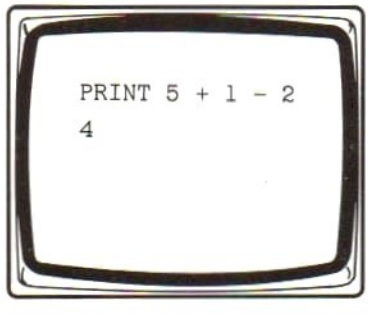


Als je deze voorbeelden gemaakt hebt, dan weet je hoe je de computer als rekenmachine kunt gebruiken.

Laten wij eens een som gaan maken:

Je hebt 5 gulden. Je krijgt er één gulden bij. Je koopt een boekje van 2 gulden. Hoeveel heb je over?
De computer kan hier snel bij helpen:

typ



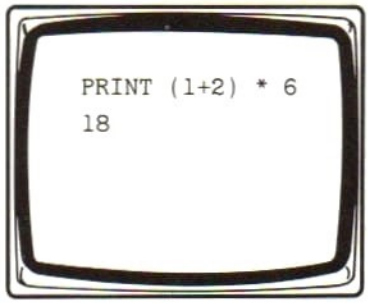
Dus 4 gulden over. Heb je in de gaten, dat je 5 + 1 − 2 gewoon achter elkaar kunt zetten?

Als je langere sommen maakt, moet je net als op school, haakjes gebruiken.

Voorbeeld: Jan heeft 6 gulden en Kees en Ans hebben ieder 6 gulden. Samen hebben ze:

(1 + 2) * 6 = 18 gulden *(goed)*
1 + 2 * 6 = 13 gulden *(fout)*

typ



De mogelijkheden voor rekenen, boekhouden, wiskunde, enz. van de computer zijn onbegrensd. Maar dat is te veel om uit te leggen.

De volgende fouten komen vaak voor als je gaat rekenen:

---

1 Een komma in plaats van een punt getypt.

```
PRINT 2,5 + 1,5                    (fout)
```

Als je dit typt krijg je drie getallen (2 6 5). Je hebt in plaats van een punt een komma gezet.

```
PRINT 2.5 + 1.5                    (goed)
```

2 Aanhalingstekens gebruikt.

```
PRINT "5 + 3"                      (fout)
5 + 3
```

Wat tussen aanhalingstekens (") wordt gezet, komt precies zo op het TV-scherm. En dat was niet de bedoeling. Je wou 5 + 3 laten uitrekenen. Dus:

```
PRINT 5 + 3
8                                  (goed)
```

---

opgaven

De oplossingen vind je achterin.

1 Hoeveel is

7 + 6
18 + 33
9 − 3
19 − 17
2 *4
12 *15
4 /2
32 /8

2 Tel 7 en 6 bij elkaar op en trek er 3 vanaf.

3 Je hebt f 2,50 zakgeld. Je geeft drie kwartjes aan je vriend. Hoeveel heb je over?

4 Drie zusjes willen samen een computerspel kopen van f 50,00. Ze krijgen ieder f 12,50 van een tante. Daarvan moet f 3,00 naar de bank van hun ouders. Hoeveel moeten ze nog sparen?

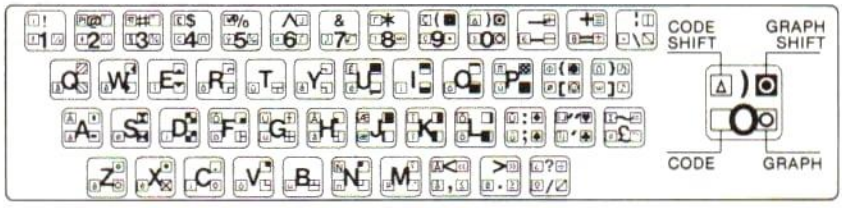
NOG MEER TEKENMOGELIJKHEDEN

Misschien ben je al eens een grappig figuurtje tegengekomen. Dat zijn de grafische tekens. Je kunt ze krijgen door op GRAPH te drukken. Houd GRAPH even vast en druk op

	U	I	O	P	[	]
je ziet nu	▬	▬	■	■	☺	♪

Er zitten nog meer van deze grappige figuurtjes op je toetsenbord en je kunt er mooi mee tekenen. Je krijgt ze door ze samen met de GRAPH -toets in te drukken.

Op deze afbeelding zie je waar ze zitten,

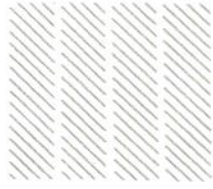


Zoek de ┼ eens op en de ▨.
Heb je ze gevonden?
Je kunt hier prachtig mee tekenen.
Probeer de volgende voorbeeldjes maar eens te maken

GRAPH + G



GRAPH + O



Er zitten nog veel meer grappige en nuttige tekens op het toetsenbord. Bij de SONY HIT BIT zit zelfs een aparte kaart, waarop al die tekens staan. Soms staan er wel vier op een toets. Kijk maar eens naar de P - of de [- toets.



Ja, je ziet het goed. Zelfs muzieknoten, kaarten, golfjes, enz.

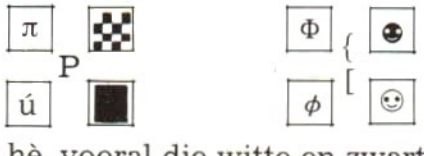
Hoe je het teken rechtsonder moet krijgen weet je al. Je moet dan de GRAPH -toets ingedrukt houden en op die letter drukken. *rechtsonder*

Als je nu op GRAPH , SHIFT en de letter drukt, krijg je het teken dat rechtsboven staat.
Probeer maar eens met de P. *rechtsboven*

Het teken linksonder krijg je door eerst op CODE en dan op de letter te drukken. Als je niet krijgt wat je verwacht dan ben je misschien vergeten dat je de tekens krijgt die op de aparte kaart staan. Ze staan niet altijd op de toetsen. *linksonder*

Met CODE , SHIFT en de letter tover je het teken dat linksboven staat, op het scherm. *linksboven*

Voor alle duidelijkheid zetten we ze nog even op een rijtje.



Leuk hè, vooral die witte en zwarte gezichtjes.

opgaven

1 Zoek de volgende tekens eens op en typ ze dan op het scherm.



2 Je kunt met deze tekens weer allerlei leuke figuren en patronen maken. Bijvoorbeeld een dambord met 
Probeer het maar eens.

KLEUREN EN MUZIEK MAKEN MET DE COMPUTER

MSX-computers hebben twee mogelijkheden die je niet op alle huiscomputers vindt: aangesloten op een kleurentelevisie kan het scherm worden 'gekleurd' en je kunt de computer ook muziek laten maken. We zullen in de volgende hoofdstukken de belangrijkste dingen leren over het gebruik van kleuren en muziek. Maar ook hiervoor geldt dat de mogelijkheden van de computer haast onbeperkt zijn en dat je daar alleen maar achter kunt komen door zelf te experimenteren. Wij zullen je laten zien dat er veel kan.

KLEUREN MET DE COMPUTER

Zoals je misschien zelf al ontdekt hebt kun je verschillende kleuren krijgen, als je de computer op een kleurentelevisie hebt aangesloten.
Je kunt dat regelen met COLOR.
Typ bijvoorbeeld maar eens `COLOR 1`.
Vergeet niet op RETURN te drukken.
Als het goed is, worden de letters nu zwart.
Je kunt maar liefst 16 verfpotten gebruiken om je letters te kleuren.
Met `COLOR 6` krijg je mooie donkerrode letters.
Hier volgen alle mogelijke kleuren.

0 TRANSPARANT
1 ZWART
2 MIDDELGROEN
3 LICHTGROEN
4 DONKERBLAUW
5 LICHTBLAUW
6 DONKERROOD
7 HEMELSBLAUW

8 MIDDELROOD
9 LICHTROOD
10 DONKERGEEL
11 LICHTGEEL
12 DONKERGROEN
13 MAGENTA
14 GRIJS
15 WIT

Ook het vel waarop je schrijft of tekent kun je één van deze kleuren geven. Typ maar eens `COLOR 6,1Ø`
Er verschijnen nu rode letters op een donkergeel vel. Je kunt nog veel meer combinaties proberen. *Bijvoorbeeld:*

`COLOR 2,7` (middelgroen op hemelsblauw)
`COLOR 15,1` (wit op zwart)
`COLOR 15,4` (wit op donkerblauw)

Als je dacht dat dit alles was heb je het mis. Zelfs de rand van het vel kan worden gekleurd. Alleen moet je even een scherm kiezen waarop een rand zit.

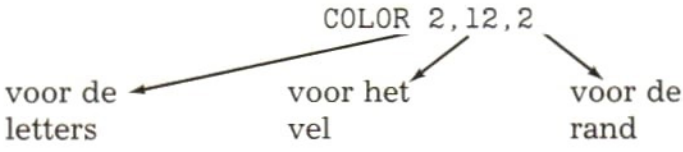
typ `SCREEN 1`

Nu zie je witte letters (15) op een donkerblauw (4) vel met een hemelsblauwe (7) rand. Er zijn nog veel mogelijkheden. *Bijvoorbeeld:*

```
COLOR 2,7,6
COLOR 15,1,5
COLOR 15,4,7
```

De laatste keuze (COLOR 15,4,7) heeft precies weer de kleuren, waar de computer mee begint als je SCREEN 1 kiest.
Je ziet dat er achter COLOR drie getallen mogelijk zijn. Het eerste getal geeft aan uit welke verfpot je de kleur voor de letters haalt. Het tweede getal geeft de kleur voor het vel. En het derde getal is voor de kleur van de rand. De rand zie je als je SCREEN 1 hebt gekozen.

Nu nog even alles op een rijtje.



Wij hebben een programmaatje gemaakt dat je alle kleuren laat zien. Typ het maar eens in.

Programma **KLEUR:**

```
 1Ø SCREEN 1
 2Ø PRINT
 3Ø PRINT "KLEUREN-DISCO"
 4Ø FOR X=0 TO 15
 5Ø FOR Y=1 TO 15
 6Ø FOR Z=2 TO 15
 7Ø COLOR X,Y,Z
 8Ø FOR PAUZE=1 TO 1Ø : NEXT PAUZE
 9Ø NEXT Z
1ØØ NEXT Y
11Ø NEXT X
12Ø COLOR 15,4,7
```

typ RUN

Als het je te snel gaat, kun je in regel 80 de 10 in 50 of 100 veranderen. Dus:

```
8Ø FOR PAUZE=1 TO 5Ø : NEXT PAUZE
```

In regel 120 zet de computer zijn normale kleuren weer op het scherm.

MUZIEK

Om muziek te maken heb je de PLAY-opdracht nodig. Typ maar eens:

typ `PLAY "CDEFGAB"` *(vergeet de aanhalingstekens niet en op* RETURN *drukken)*

Prachtig hè? Je hoort alle tonen van een toonladder (DO, RE, MI, FA, SOL, LA, SI). Men noemt die ook wel C, D, E, F, G, A, B. Voor de computer zullen wij niet DO maar C gebruiken en niet RE maar D, enz. Met de letters geef je dus aan welke toon je wilt horen.



Mensen kunnen toonladders zoals DO, RE, MI op verschillende manieren zingen. Mannen zingen deze meestal lager dan vrouwen. Ook bij de computer kun je aangeven of je een hoge of een lage toonladder wilt gebruiken. In muziektermen zeg je dan welke octaaf je wilt gebruiken. Je doet dat bij de computer met de O van octaaf. O1 is de laagste en O8 de hoogste. Probeer het volgende voorbeeld maar eens:

typ `PLAY "O2CDEFGAB"`
typ `PLAY "O6CDEFGAB"`

Met het programma 'TOONLADDER', dat achterin staat, kun je alle tonen van de verschillende octaven horen. Als je niets zegt, neemt de computer vanzelf octaaf 4. Dit is nog lang niet alles. Je kunt ook aangeven hoe lang iedere toon moet duren.

typ `PLAY "C64 D32 E16 F8 G4 A2 B1"`

Heb je het gehoord? De C, D, en E gaan heel snel en de B duurt het langst. De langste noot duurt twee tellen.

Hoe groter het getal, des te korter duurt de noot. Hieronder zie je ze op een rijtje met de noten er bij, zoals die in de muziek worden geschreven.

	C64	D32	E16	F8	G4	A2	B1
noot	*C*	*D*	*E*	*F*	*G*	*A*	*B*
maat	*vier-enzes-tigste*	*twee-ender-tigste*	*zes-tien-de*	*acht-ste*	*kwart*	*half*	*heel*
muziek-schrift							

Iedere keer als je een muziekje wilt spelen, moet je PLAY typen met daarachter de noten (tussen aanhalingstekens). Dat is soms wel eens lastig.
Je kunt de noten ook in wat men 'strings' noemt, zetten. Dit is heel eenvoudig. Typ het volgende maar eens precies na:

```
typ     10 A$="02 CDEFGAB"
typ     20 B$="03 CDEFGAB"
typ     30 PLAY A$
typ     40 PLAY B$
typ     50 PLAY A$,B$

typ     RUN
```

Je hebt nu een klein programma gemaakt, dat je net zo vaak kunt spelen als je wilt. Gewoon iedere keer RUN typen en daarna op RETURN drukken. Het programma start dan weer opnieuw, want het staat in het geheugen. Je bent het pas kwijt als je op RESET drukt of de stroom uitzet of als je NEW typt en dan op RETURN drukt.

Heb je gezien dat er in regel 50 van het programma iets nieuws staat?

```
5Ø PLAY A$,B$
```

Met deze regel geef je aan dat de computer tegelijk moet spelen wat er in A$ en in B$ staat. Dat kan, want de computer heeft drie stemmen. Hij kan dus drie tonen tegelijk zingen of spelen. Bij PLAY A$,B$ worden er twee gebruikt. Wil je er drie gebruiken dan moet je van regel 50 maken:

```
5Ø PLAY A$,B$,A$
```

De stemmen 1 en 3 spelen dan wat er in A$ staat en stem 2 wat er in B$ staat.
De meeste liedjes klinken mooier als je meer stemmen tegelijk gebruikt.

Je kunt nog veel meer dingen regelen. Het is te veel om hier allemaal uit te leggen. We zullen je nog een paar dingen kort laten zien. Hier komen ze:

typ `PLAY "O3 V15 T15Ø CDEFG R2 AB"`

Wat heb je nu gedaan?
- Met O3 geef je aan dat je de derde octaaf wilt gebruiken.
- Met V15 zet je de sterkte van het geluid (het volume) op 15.
- Met T150 heb je het tempo op 150 gezet.
- Met R2 maak je een pauze of rust van een halve tel.

Al deze dingen kun je apart regelen. Hier komen ze nog een keer:

```
PLAY   "O4    V1Ø    T12Ø   CDEFG  R2     AB"
```

	octaaf van 1 t/m 8	*volume van 1 t/m 15*	*tempo van 1 t/m 255*	*noten*	*rust 1, 2, 4, 8, 16, 32 of 64*	*noten*

Je kunt werkelijk prachtige muziek maken met zonodig drie stemmen of instrumenten tegelijk. De getallen die hierboven achter PLAY staan, zoals de 4 bij O en de 10 bij V, zijn de getallen die de computer kiest als jij niets

invult. Dus als jij geen octaaf aangeeft, dan kiest de computer vanzelf octaaf 4. Het volume komt dan op 10 en het tempo wordt vanzelf 120.
Je kunt nu, als je dat wilt muziekstukken omzetten. Daarna kun je dan eventueel wijzigingen aanbrengen om het mooier te maken. Hoe je een muziekstukje omzet, zullen we even laten zien.



d c b a g a b c

```
PLAY "D4  C8   B4   A8   G4   A8   B4   C8"
```

Achterin dit boek staan een paar demonstratieprogramma's. Je moet ze maar eens bekijken. Het is echt heel eenvoudig om met MSX-computers muziek te maken.

DEEL 2

PROGRAMMEREN IN BASIC

KLAAR OM TE PROGRAMMEREN?

Je weet nu hoe je via het toetsenbord kunt tekenen en schrijven. Wat je op het toetsenbord intypt, kun je op het TV-scherm krijgen. Ook weet je hoe je letters en getallen op het TV-scherm kunt veranderen of uitvegen. Dat zijn de belangrijkste dingen die je moet weten om de fantastische wereld van het programmeren binnen te stappen. Als je kunt programmeren, kun je de computer opdrachten gaan geven die hij voor jou moet gaan uitvoeren. Je kunt hem snel laten rekenen, je kunt hem dingen laten onthouden, zodat hij je bijvoorbeeld kan overhoren, je kunt spelletjes ontwerpen en nog veel, veel meer.

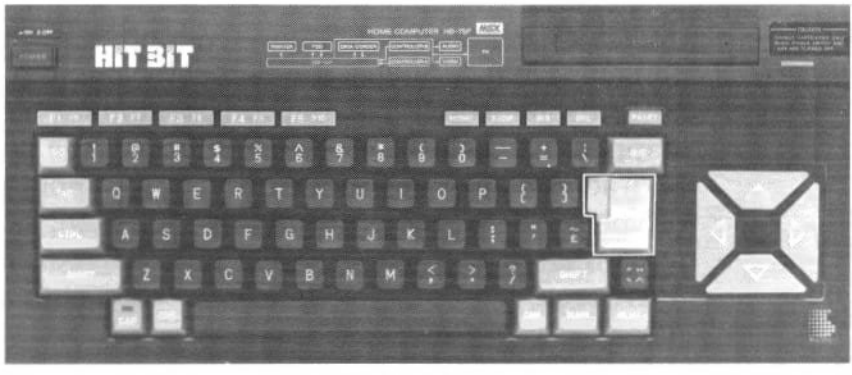
Wat is programmeren?
Programmeren is het schrijven van een programma, een soort brief, waarin precies staat wat de computer moet gaan doen. Zo'n programma bestaat uit regels die allemaal moeten beginnen met een nummer.

REGEL-NUMMER

Zo'n nummer noemen we een REGELNUMMER. De regelnummers geven aan, in welke volgorde een programma moet worden uitgevoerd. Je zou daarvoor nummer 1, 2, 3 enz. kunnen nemen. Als je de computer dan de opdracht geeft om een programma te gaan uitvoeren begint hij bij regel 1, dan regel 2, dan regel 3 enz.
Nu is programmeren een heel creatief en precies werkje. Vaak wil je tijdens het werken aan je programma tussen twee regels nog een regel zetten.
Maar wanneer je als eerste regel 1 hebt gekozen en als tweede regelnummer 2, kun je er niets meer tussen zetten. Je mag alleen hele nummers gebruiken. Daarom beginnen programma's bijna allemaal met regelnummer 10 en daarna 20, 30, 40 enz. Later kun je dan

gemakkelijk tussen de eerste regel 10 en de tweede regel 20 nog een opdracht zetten op bijvoorbeeld regel 12, 15 of 18. Net zoals je een regel moet beginnen – met een heel nummer – moet je ook iedere regel eindigen.

RETURN Je beëindigt een regel altijd, door op RETURN te drukken. Dat is de knop die rechts ongeveer in het midden zit.



De RETURN knop is heel belangrijk. Alles wat je op het TV-scherm typt, krijgt pas betekenis voor de computer als je op de RETURN knop drukt. Dan pas gaat de computer controleren of wat op die regel staat ook zo in zijn geheugen mag. Na het indrukken van de RETURN knop gaat de computer heel snel kijken of er een regelnummer voorstaat, of er begrijpelijke opdrachten tussen staan en nog veel meer. Als alles goed gaat, accepteert de computer die regel en stopt hem in zijn geheugen. Op het scherm is die controle niet te zien. Je ziet alleen maar dat het blokje (de cursor) weer helemaal links staat. Je kunt dan een nieuwe regel intypen.
Als je iets fout hebt gedaan, kun je het beste de regel nog eens opnieuw typen.
De computer kan op één regel vrij veel tekens onthouden.
Meestal stoppen wij er maar 10 of 20 tekens in. Een aantal van die regels vormen dan samen een programma. Je moet dus onthouden dat een programma in BASIC er altijd als volgt uit moet zien:

regelnummer	
bijv. 10	*iedere regel eindigen*
20	*met een druk op* 
30	
100 END	

END

Zoals je ziet eindigen we op de laatste regel (in dit geval 100) met END. END is het Engelse woordje voor einde. Als de computer END tegenkomt, weet hij dat het programma klaar is. Soms staat END niet aan het einde van een programma, maar dat is eigenlijk niet mooi. Mooie programma's eindigen altijd met END.

OP-DRACH-TEN

Je weet nu hoe een programma er uitziet. In een programma kun je opdrachten zetten, deze heten: programma-opdrachten. Verder bestaan er opdrachten die de computer kent om een programma te beginnen, te laten bekijken en uit te voeren, deze noemen we directe opdrachten. Hier volgen de opdrachten die wij je zullen leren. Er zijn er nog meer maar met deze opdrachten kun je heel ver komen.

DIRECTE opdrachten	*PROGRAMMA-opdrachten.*
NEW	*PRINT*
LIST	*INPUT*
RUN	*GOTO*
STOP	*GOSUB en RETURN*
CTRL en STOP	*REM*
RESET	*IF ... THEN*
	FOR ... NEXT
	RND
	END

De programma-opdrachten komen in de volgende hoofdstukken aan bod. Dan leren we je hoe je ze kunt gebruiken in een programma. De directe opdrachten zullen we nu uitleggen. Dan weet je hoe je ervoor kunt zorgen dat de computer je programma's uitvoert.

NEW

NEW is heel eenvoudig het Engelse woordje voor nieuw. Als je een nieuw programma wilt intypen, dan begin je met:

typ *NEW* *(RETURN indrukken)*

De computer maakt dan eerst zijn geheugen schoon, zodat jij je programma kunt intypen. Het lijkt een beetje op eerst je tafel of bureau opruimen en een nieuw stuk papier en een potlood pakken. Of een cassettebandje aan het begin zetten, zodat je een nieuwe L.P. kunt opnemen. Na NEW is het geheugen schoon en je kunt beginnen.
Typ nu het volgende kleine programmaatje eens in.

```
typ     NEW                  (druk op RETURN )
typ     10 PRINT "DAG";      (druk op RETURN )
typ     20 PRINT "ARLEEN"    (druk op RETURN )
typ     30 GOTO 10           (druk op RETURN )
typ     40 END               (druk op RETURN )
```

Zo maak nu het TV-scherm maar schoon. (Je weet dat je dan SHIFT en HOME tegelijk in moet drukken.) En whoeps, alles wat er stond is weg.
Zou je programma nu helemaal weg zijn zodat je opnieuw moet beginnen? Welnee, het programma is niet weg, je hebt het alleen maar even verstopt. Het nieuwe blaadje met je programma erop zit als het ware in de la van je bureau. Je kunt het blaadje er weer uit halen en bekijken. Je hebt daar de directe opdracht LIST voor nodig.

LIST

Als je LIST intypt, verschijnt het blaadje weer op het TV-scherm en kun je het programma weer bekijken. Probeer het maar eens:

typ *LIST* *(druk op RETURN)*

Staat er nu:

```
1Ø PRINT "DAG";
2Ø PRINT "ARLEEN"
3Ø GOTO 1Ø
4Ø END
```

Ja, dan is het goed.
Als je heel lange programma's hebt, die niet op het TV-scherm passen, dan komen alleen de laatste regels op het TV-scherm. Geen nood want je kunt de eerste regels ook bekijken. Als je bijvoorbeeld een programma hebt van 100 regels (ja die bestaan!) dan kun je die regels bekijken door te typen:

```
LIST 1Ø - 1ØØ
```

De computer laat je dan de regels met nummers tussen 10 en 100 zien.
Daarna kun je dan de volgende regels bekijken door te typen:

```
LIST 11Ø - 21Ø
```

Eenvoudig he?
Je weet nu wat je moet doen als je een nieuw programma wilt gaan maken (NEW typen).
Je kunt bekijken of dat programma goed in het geheugen staat (LIST typen). Nu moet je de computer nog de directe opdracht geven om het programma uit te voeren.

RUN

Dat doen we met de directe opdracht RUN.

typ RUN *(druk op* RETURN *)*

Zie je nu

```
DAG ARLEEN
DAG ARLEEN
DAG ARLEEN
DAG ARLEEN
DAG ARLEEN
```

over het TV-scherm gaan?

Ja? Gefeliciteerd! Je hebt nu een programma gemaakt en door de computer laten uitvoeren.
RUN is het Engels voor hardlopen. En hardlopen kan de computer. In vliegende vaart zoekt de computer de eerste regel (10) op, voert die uit, gaat snel naar de volgende regel (20), net zo lang tot de computer END tegenkomt. Ook programma's van 100 regels of meer rent de computer moeiteloos langs.

STOP

Maar hoe kun je hem nu laten stoppen? Door op [STOP] te drukken. Midden in het programma houdt de computer op met uitvoeren. Als je weer op [STOP] drukt, gaat hij verder. Met [STOP] kun je het programma dus even onderbreken.

CTRL EN STOP

Om helemaal te stoppen moet je [CTRL] en [STOP] tegelijk indrukken. De computer vertelt je waar hij was gebleven.
Hij zegt: BREAK IN 20 (bijvoorbeeld: het nummer waar hij was).

RESET

Als je op [RESET] drukt, stopt hij en maakt het scherm schoon. Je programma is dan weg. Je moet dan helemaal opnieuw beginnen. Dus kijk uit.

Zo dit waren de directe opdrachten waarmee je programma's kunt beginnen, kunt bekijken en uitvoeren. Onthoud maar:

NEW	Het geheugen wordt dan schoongemaakt en je kunt een nieuw programma beginnen.
LIST	De computer laat zien wat in zijn geheugen staat.
RUN	Is het startsein voor de computer om het programma te gaan uitvoeren.
STOP	Door op [STOP] te drukken onderbreekt de computer een programma. Door nogmaals op [STOP] te drukken gaat hij weer verder.
CTRL en STOP	Door [CTRL] en [STOP] tegelijk in te drukken stopt het programma.
RESET	Door op [RESET] te drukken stopt de computer en veegt zijn hele geheugen schoon. Je moet dan helemaal opnieuw beginnen. Dus let op! Het is net alsof je de computer even uitzet.

LEREN PROGRAMMEREN

INLEIDING

Dit tweede deel van het boek gaat over de taal BASIC. In BASIC worden programma's geschreven die de computer kan begrijpen. BASIC is een taal zoals Nederlands, Engels en Frans talen zijn. Een computer is alleen geen mens die talen als Nederlands en Engels zomaar kan leren en begrijpen. De computer moet je in zijn eigen taal vertellen wat hij moet doen, anders snapt hij het niet. Dat betekent, dat jij BASIC moet leren, want jij bent een mens en mensen kunnen wel nieuwe talen leren. Jij kunt ook – misschien doe je dat later nog – andere computertalen leren als PASCAL, ALGOL, LOGO enz.
BASIC is de meest gebruikte taal die bijna iedere computer snapt. De kunst is dus, om als je de computer iets voor je wilt laten doen, bijvoorbeeld reken eens een som voor me uit, dat zó te doen dat hij je begrijpt en doet wat je zegt. Bedenk dan eerst in gewoon Nederlands wat je van hem wilt en vertaal dat in BASIC.
In dit deel leer je de computertaal BASIC. Je moet dan de hoofdstukken op volgorde lezen. Als je het woordje typ ziet, moet je die regels op je eigen computer typen en doen wat er staat.
BASIC heeft maar een paar werkwoorden. Opdrachten noemen we die. Meestal staat er in elke zin (regel) één zo'n opdracht. Bijvoorbeeld het woord PRINT is de vertaling van het Nederlands: schrijf op het TV-scherm. En GOTO is de vertaling voor het Nederlands: ga naar de regel ... In ieder van de volgende hoofdstukken leer je een belangrijke opdracht. Als je deze hebt geleerd, kun je de computer al veel voor je laten doen, zoals je helpen bij het huiswerk, telefoonnummers van je vrienden onthouden, uitrekenen hoe je met je geld kunt omgaan, je eigen spelletjes maken enz.

Elk hoofdstuk ziet er hetzelfde uit. Eerst lees je welke opdracht je gaat leren en wat je de computer met die opdracht kunt laten doen. Bijvoorbeeld:

Met PRINT kun je tekst op het TV-scherm krijgen

Dan volgt een voorbeeld hoe de opdracht gebruikt wordt en de vertaling daarvan in het Nederlands. Bijvoorbeeld:

In BASIC:	*In Nederlands:*
`1Ø PRINT "DAG"`	*10 zet DAG op het TV-scherm*

Daarna krijg je programmaatjes met die opdracht die je moet typen. Bijvoorbeeld:

typ `1Ø PRINT "DAG"`
typ `2Ø PRINT "OSCAR"`
typ `3Ø END`

Het zal je waarschijnlijk niet altijd direct lukken de opdrachten zó te geven dat de computer je snapt. Je moet zijn taal dan nog iets beter leren spreken. Daarom staan achter elk hoofdstuk de vergissingen die je vaak maakt als je deze taal leert. Ook staat erbij hoe je de fouten kunt verbeteren. Soms zie je een TV-scherm staan.



Wat je in dit kader ziet, kun je op het TV-scherm van jouw computer verwachten.
Behalve de opdrachten tussendoor, staan er aan het eind van elk hoofdstuk opgaven die je moet kunnen maken als je de stof goed hebt begrepen. De antwoorden van deze opgaven vind je achterin het boek. *Veel succes!*

DE PRINT-OPDRACHT

PRINT

Met PRINT kun je tekst op het TV-scherm schrijven

In BASIC:	*In Nederlands:*
`1Ø PRINT "DAG"`	*10 zet DAG op het TV-scherm*
`2Ø X = 1Ø`	*20 zet in vakje X getal 10*
`3Ø PRINT X`	*30 zet 10 op het TV-scherm*

We gaan met het PRINT-commando de computer leren om netjes 'dag' te zeggen. Het programma ziet er zó uit: (vergeet niet NEW te typen)

typ `1Ø PRINT "DAG";` – *let op de punt/komma*
typ `2Ø PRINT "OSCAR"` – *of jouw naam*
typ `3Ø GOTO 1Ø`
typ `4Ø END`

Zo, wat heb je nu gedaan?

Je hebt vier regels in het geheugen van de computer gezet. Laten we eens kijken wat ze betekenen.

op regel 10 staat	'zet DAG op het TV-scherm'
op regel 20 staat	'zet OSCAR op het TV-scherm'
op regel 30 staat	'ga weer naar regel 10 en begin opnieuw'
op regel 40 staat	'einde van het programma'

Als de vier regels er precies staan zoals het moet, dan volgt nu een spannend moment. Je geeft de computer de opdracht om jouw programma uit te voeren.

typ `RUN` *(niet vergeten op* RETURN *te drukken)*



Als dit komt dan doet de computer wat jij wilde.

En de computer wil maar niet stoppen. De computer gaat net zo lang door met jouw naam op het scherm te schrijven tot dat jij zegt dat hij moet stoppen. In 'klaar om te programmeren' hebben we je uitgelegd, hoe je moet stoppen. Onderbreek het programma met CTRL en STOP (tegelijk indrukken).
Zo, nu heb je de computer geleerd om jou gedag te zeggen. Netjes hè?

Laten we nog een voorbeeld maken.

Programma **NAAM:**

```
typ       NEW
typ       10 PRINT "**************"
typ       20 PRINT "DIT PROGRAMMA"
typ       30 PRINT
typ       40 PRINT "    IS VAN"
typ       50 PRINT
typ       60 PRINT "OSCAR BAARDA"
typ       70 PRINT "**************"
typ       80 END

typ       RUN
```



Op regel 30 en 50 staat alleen PRINT. Als je alleen PRINT typt en niets daarachter, dan krijg je er een regel tussen. Daarmee kun je wat op het scherm komt vaak mooier maken.

Je zult het al wel gezien hebben. Telkens als je tekst op het scherm wilt laten schrijven, moet je die tekst tussen aanhalingstekens " en " zetten. Dus:

```
PRINT "***********"
```

Zie je de aanhalingstekens?

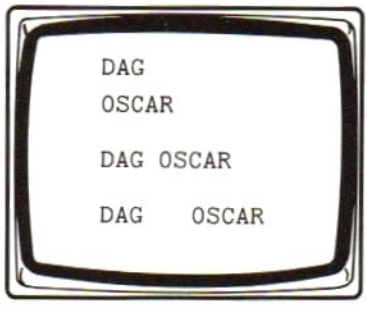
Alleen PRINT geeft een regel waar niets staat

Kijk eens naar de volgende voorbeelden.

```
typ     10 PRINT "DAG"      niets er achter
typ     20 PRINT "OSCAR"
typ     30 PRINT

typ     40 PRINT "DAG ";    punt/komma erachter
typ     50 PRINT "OSCAR"
typ     60 PRINT

typ     70 PRINT "DAG",     komma er achter
typ     80 PRINT "OSCAR"
typ     90 END
typ     RUN
```



Je kunt dezelfde teksten op verschillende manieren op het TV-scherm laten verschijnen.

- onder elkaar — *doet hij uit zichzelf*
- achter elkaar — *dan; (punt/komma typen)*
- met ruimte ertussen — *dan, (komma typen)*

Met PRINT kun je tekst op het TV-scherm schrijven

herstel

1 Ben je de " (aanhalingstekens) vergeten?
Dan moet je de regel opnieuw typen.

2 Ben je de spaties vergeten?

Meestal is dat niet zo erg. Het programma draait toch wel. Het programma ziet er echter niet zo mooi uit. Het is verstandiger om de spaties toch aan te brengen. Zoals je weet, krijg je een spatie door de spatiebalk in te drukken.

3 Ben je de ; (punt/komma) vergeten?

Dan schrijft de computer de volgende regel op een nieuwe lijn en dat wou je niet. De punt/komma zorgt er voor dat de teksten achter elkaar worden gezet. Regel opnieuw typen.

4 Heb je PINT OF PRIT getypt?

Typ dan de regel opnieuw en let er op dat je PRINT typt.

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten een* RETURN *te geven)*

opgaven

1 Zet op regel 20 de dag, bijvoorbeeld zondag, neer en kijk of je het met RUN op het TV-scherm krijgt.

typ 1Ø PRINT "VANDAAG IS HET"
typ 2Ø ....
typ 3Ø END

2 Verander het volgende programma zodanig dat de teksten achter elkaar komen op het TV-scherm.

typ 1Ø PRINT "DE ZON"
typ 2Ø PRINT "SCHIJNT"
typ 3Ø END

Nu staat er



Het moet worden



3

Nu moet het worden



4

Nu moet het worden



– *een regel ertussen*

DE INPUT-OPDRACHT

INPUT

Met INPUT vraag je de gebruiker om een getal of een naam

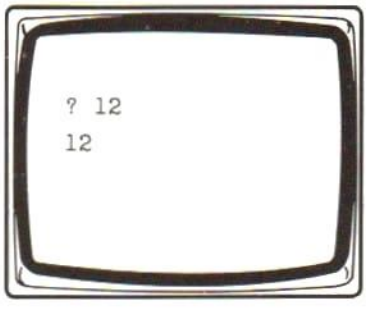
In BASIC:	*In Nederlands:*
`1Ø INPUT X`	*10 geef een getal*
`2Ø INPUT N$`	*20 geef een naam*

Met de opdracht INPUT vraag je de gebruiker vanuit het programma om een getal of een naam. Een getal kan 1, 5 of veel groter zijn. Een naam kan A, H, PIET of wat dan ook zijn. Elke combinatie van een aantal tekens is een naam. Je kunt zo complete zinnen typen.

Twee voorbeelden:

```
1Ø INPUT X
2Ø PRINT X
3Ø END
```

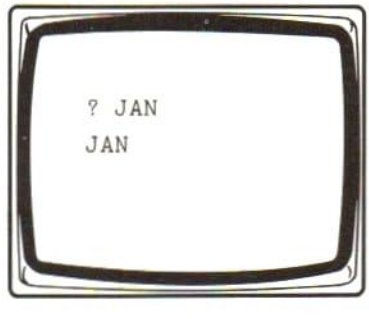
typ RUN



typ een getal

```
1Ø INPUT N$
2Ø PRINT N$
3Ø END
```

typ RUN



typ een naam

Opletten: Als je na INPUT geen $ geeft, bijvoorbeeld X, L of K, dan moet je een getal intypen en niets anders.

Als je namen wilt gaan gebruiken in je programma, moet je na de letters achter INPUT een $ zetten. De $ vind je boven de 4.

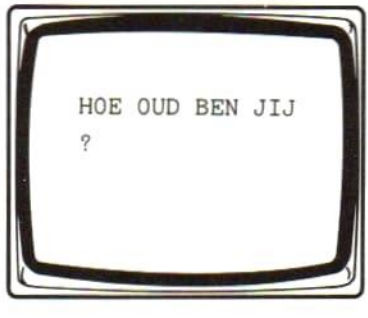
Een $ noemen we een stringteken

Een stringteken geeft aan dat je met andere tekens dan alleen getallen wilt werken. Als je er getallen instopt (bijvoorbeeld een telefoonnummer als 888945), kun je niet met deze getallen rekenen. De computer denkt dat het getal een naam is.

Programma **LEEFTIJD:**

```
typ  10 PRINT "HOE OUD BEN JIJ"
typ  20 INPUT L
typ  30 PRINT "WELK JAAR IS HET"
typ  40 PRINT "BIJVOORBEELD 1986"
typ  50 INPUT J
typ  60 PRINT "DAN BEN JIJ IN HET"
typ  70 PRINT "JAAR 2000 "; L (2000 - J)
typ  80 END

typ  RUN
```



– *typ een getal*
bijvoorbeeld 12

Je ziet dat er hier wordt gevraagd om getallen via INPUT L en INPUT J. Een naam is niet mogelijk, want er staat achter L en J geen $ (stringteken).

Programma **NAAM:**

```
typ   1Ø PRINT "WAT IS JE VOORNAAM"
typ   2Ø INPUT V$
typ   3Ø PRINT "DANK JE WEL "; V$
typ   4Ø PRINT "WAT IS JE ACHTERNAAM"
typ   5Ø INPUT A$
typ   6Ø PRINT "DUS JIJ BENT "; V$ ; " " ; A$
typ  1ØØ PRINT "TOT ZIENS"
typ  11Ø END

typ  RUN
```

Als je zelf programma's gaat maken, is het wel eens nodig dat je de computer laat vragen of je door wilt gaan. Dat kan met het volgende gedeelte dat past in het programma 'NAAM'.

```
typ  7Ø PRINT "WIL JE STOPPEN. JA OF NEE."
typ  8Ø INPUT J$
typ  9Ø IF J$ = "NEE" THEN GOTO 1Ø
typ  RUN
```

Met INPUT vraag je de gebruiker om een getal of een naam

herstel

1 Een punt in plaats van een komma.

```
? 12,5                                    (fout)
```

De computer neemt alleen 12 als getal mee. Als je toch 12,5 wilt typen, moet je een punt zetten in plaats van een komma, dus 12.5.

```
? 12.5                                    (goed)
```

2 Aanhalingstekens vergeten als je $ bij INPUT hebt gebruikt. Als je in je programma bijvoorbeeld X$ gebruikt, zoals in:

```
2Ø IF X$ = JAN THEN GOTO 1ØØ          (fout)
```

dan moet je steeds aan de aanhalingstekens denken. Ook bij getallen.

Dus

```
2Ø IF X$ = 6 THEN GOTO 1ØØ            (fout)
2Ø IF X$ = "6" THEN GOTO 1ØØ          (goed)
```

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten een*  *te geven)*

---

opgaven

1 Is dit programma goed?

```
1Ø PRINT "WAT IS JE NAAM?"
2Ø INPUT L
3Ø PRINT "JE NAAM IS ";L
4Ø END
```

2 Is dit programma goed?

```
1Ø PRINT "WAT IS JE LEEFTIJD ?"
2Ø INPUT L$
3Ø MN = L$ * 12
4Ø PRINT " JE BENT ";MN; " MAANDEN JONG"
5Ø END
```

3 Verander programma 1 en 2 zodanig dat ze goed zijn.

DE GOTO-OPDRACHT

GOTO

Met GOTO kun je een sprong maken in het programma

In BASIC:	*In Nederlands:*
`1Ø GOTO 5Ø`	*10 ga naar regel 50*
`5Ø END`	*50 einde*

GOTO bestaat uit de Engelse woordjes GO en TO en betekent 'ga naar'. Na GOTO volgt meestal een regelnummer, bijvoorbeeld GOTO 80 of GOTO 10. Met deze opdracht kun je een sprong maken in het programma.
Er zijn veel mogelijkheden:

spring naar een bepaalde regel
spring naar laatste regel
spring naar eerste regel.

Naar boven springen of naar beneden springen is allemaal mogelijk.

Voorbeeld van met GOTO springen.
Als in regel 40 X gelijk is aan R wordt er naar regel 70 gesprongen. En in regel 60 springt de computer terug naar regel 20. Je ziet dat je zowel naar boven als naar beneden kunt springen met GOTO.

```
typ  1Ø R = 3
typ  2Ø PRINT "RAAD EEN GETAL ?"
typ  3Ø INPUT X
typ  4Ø IF X = R THEN GOTO 7Ø
typ  5Ø PRINT "PROBEER HET NOG EENS"
typ  6Ø GOTO 2Ø
typ  7Ø PRINT "JA, HET IS ";X
```

Typ geen NEW. Als je het volgende programma achter het vorige zet heb je een spelletje.

Voorbeeld van met GOTO naar het einde (hier regel 140) springen.

```
typ   8Ø PRINT "WIL JE STOPPEN ?"
typ   9Ø PRINT "TYP JA OF NEE"
typ  1ØØ INPUT J$
typ  11Ø IF J$ = "JA" THEN GOTO 14Ø
typ  12Ø R = R + 2
typ  13Ø GOTO 2Ø
typ  14Ø END

typ  RUN
```

Dus je ziet dat je met GOTO naar het begin of het einde kunt springen. Maar ook naar een regel die ergens in het midden staat, is mogelijk. Met GOTO kun je overal naar toe springen.

Met GOTO kun je een sprong maken in het programma

herstel

1 Naar de volgende regel springen

```
6Ø GOTO 7Ø                                    (fout)
7Ø PRINT .......
```

Dit is niet zozeer fout maar overbodig. Na regel 60 voert de computer zelf al de volgende regel (70) uit. Deze sprong hoef je niet te maken.

2 Naar een opdracht springen.

```
6Ø GOTO END                                   (fout)
```

Dit kan niet. De meeste computers verwachten achter GOTO een regelnummer (bijvoorbeeld 100) en niet een opdracht (bijvoorbeeld END).

```
 6Ø GOTO 1ØØ
1ØØ END                                       (goed)
```

3 Naar zichzelf springen.

```
6Ø GOTO 6Ø                                    (fout)
```

Deze 'ga naar' springt naar zichzelf. Het gevolg is dat de computer eindeloos doorgaat. Van 60 naar 60 naar 60 naar 60 enz. Je zult hem dan met CTRL en STOP stil moeten zetten.

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten een* RETURN *te geven)*

---

opgaven

1 Welke regelnummers moet je hier intypen?

```
typ   1Ø Y = 6
typ   2Ø PRINT "GEEF EEN GETAL TUSSEN 1 EN 1Ø"
typ   3Ø INPUT X
typ   4Ø IF X = Y THEN GOTO ...
typ   5Ø PRINT "NEE, DAT IS MIJN GETAL NIET"
typ   6Ø GOTO ...
typ   7Ø PRINT "JA, DAT IS MIJN NUMMER " ; X
```

2 Geef geen NEW na opgave 1, we gaan door met dat programma.
Welke regelnummers kun je intypen om dit programma af te maken?

```
typ   8Ø PRINT "WIL JE STOPPEN ?"
typ   9Ø PRINT "TYP JA OF NEE"
typ  1ØØ INPUT J$
typ  11Ø IF J$ = "JA" THEN GOTO ...
typ  12Ø PRINT "DUS JE WILT NIET STOPPEN"
typ  13Ø GOTO ..
typ  14Ø END
```

3 Hoe kun je het volgende programma mooier maken door van GOTO gebruik te maken ?

```
typ   1Ø N = 1
typ   2Ø PRINT N
typ   3Ø N = N + 1
typ   4Ø PRINT N
typ   5Ø N = N + 1
typ   6Ø PRINT N
typ   7Ø N = N + 1
typ   8Ø PRINT N
typ   9Ø END
```

DE GOSUB- EN RETURN-OPDRACHTEN

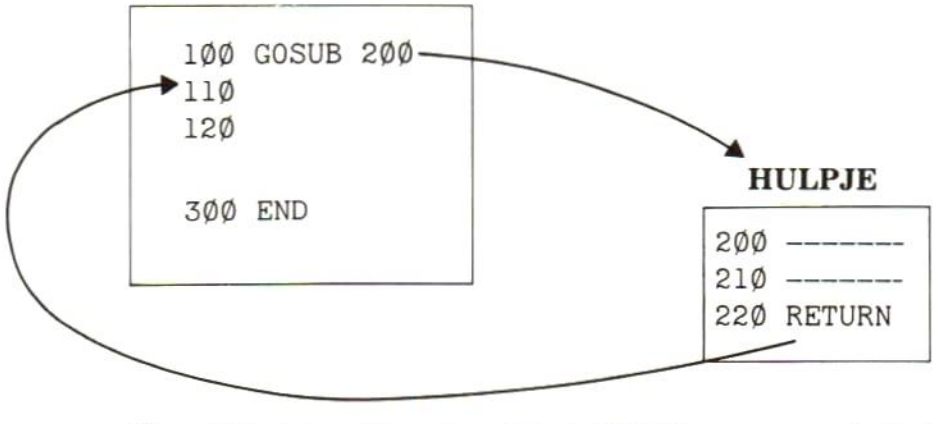
GOSUB EN RETURN

Met GOSUB maak je een sprong en met RETURN ga je weer terug

In BASIC:	*In Nederlands:*
`1Ø GOSUB 1ØØ`	*10 ga naar regel 100*
`2Ø`	*20*
`3Ø`	*30*
`1ØØ PRINT "HULPJE"`	*100 begin hulpprogramma*
`12Ø RETURN`	*120 ga terug en begin bij regel 20*

GOSUB betekent 'doe eerst het hulpprogramma vanaf regel ..'. Met GOSUB maak je een sprong naar een hulpprogramma. Ieder hulpprogramma moet eindigen met een RETURN. Dat betekent 'ga terug'. GOSUB en RETURN horen bij elkaar. Als je GOSUB gebruikt, moet je ook RETURN gebruiken

HOOFDPROGRAMMA



Vanuit het hoofdprogramma spring je naar een gedeelte van het programma (hulpprogramma) en daarna weer terug naar de volgende regel in het hoofdprogramma.

Programma **REKEN:**

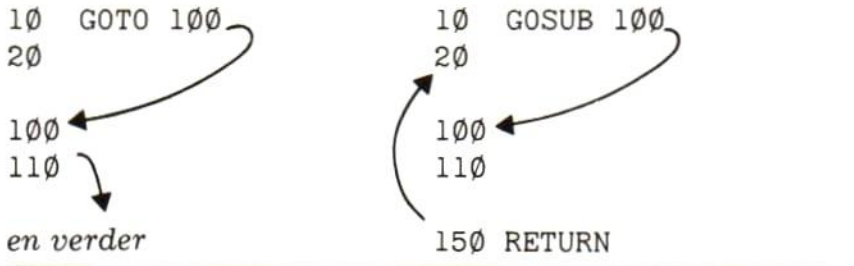
```
typ    1Ø PRINT "REKENPROGRAMMA"
typ    2Ø PRINT "GEEF EEN GETAL"
typ    3Ø INPUT G1
typ    4Ø PRINT "GEEF NOG EEN GETAL"
typ    5Ø INPUT G2
typ    6Ø GOSUB 1ØØ
typ    7Ø GOSUB 2ØØ
typ    8Ø GOTO 3ØØ

typ   1ØØ PRINT G1;" + ";G2;" = ";G1 + G2
typ   11Ø RETURN
typ   2ØØ PRINT G1;" - ";G2;" = ";G1 - G2
typ   21Ø RETURN
typ   3ØØ END

typ   RUN
```

Nu zie je gelijk wat het voordeel is van GOSUB en RETURN. Je kunt er zo vaak als nodig is naar toe springen. Als je vaak hetzelfde moet doen in een programma, dan is het handig om een hulpprogramma te maken waar je iedere keer naar toe springt.
GOSUB lijkt een beetje op GOTO ('ga naar'). Maar met GOSUB kom je weer terug en met GOTO niet. Met GOTO blijf je waar je bent.

Verschil GOTO en GOSUB



GOSUB en RETURN zijn handig als je vaak hetzelfde moet doen. Net zoals in sommige liedjes waar je achter ieder couplet een refrein moet zingen.

HOOFDPROGRAMMA

Alles in de wind,
alles in de wind,
't is maar een schipperskind.

GOSUB

O, wat een spijt,
o, wat een spijt,
nu is m'n zusje kwijt.

GOSUB

O, wat een nood,
o, wat een nood,
nu is mijn zusje dood.

GOSUB

Op die brug,
op die brug,
vond ik m'n zusje terug.

GOSUB

Einde

HULPPROGRAMMA

refrein:

Kom hier, Rosa,
je bent m'n zusje,
je bent m'n zusje,
kom hier, Rosa,
je bent m'n zusje, ja, ja.

RETURN

Hier moet je op vier verschillende plaatsen het refrein 'kom hier, Rosa' doen. Dat is vier keer hetzelfde. En als je vaak hetzelfde op verschillende plaatsen moet doen, kun je er een hulpprogramma van maken met GOSUB en RETURN.
Je ziet dat er vier keer GOSUB staat en maar één keer RETURN.

Programma **GETALLEN RADEN:**

```
typ   10 PRINT "GETALLEN RADEN"
typ   20 X=7
typ   30 GOSUB 100
typ   40 X=4
typ   50 GOSUB 100
typ   60 X=9                                   } hoofd-
typ   70 GOSUB 100                               programma
typ   80 PRINT "EN NU ZIJN WE KLAAR"
typ   90 GOTO 200

typ  100 PRINT "WELK GETAL HEB IK?"
typ  110 INPUT G
typ  120 IF G = X THEN GOTO 150
typ  130 PRINT "PROBEER HET NOG EENS"          } hulp-
typ  140 GOTO 110                                programma
typ  150 PRINT "GOED ZO, DE VOLGENDE.
typ  160 RETURN

typ  200 END

typ  RUN
```

Met GOSUB maak je een sprong en met RETURN ga je weer terug

herstel

1 Een RETURN vergeten.

```
 1Ø GOSUB 5Ø
 2Ø GOTO 1ØØ

 5Ø PRINT "DIT IS REGEL 5Ø"          (fout)

1ØØ END
```

```
 1Ø GOSUB 5Ø
 2Ø GOTO 1ØØ

 5Ø PRINT "DIT IS REGEL 5Ø"          (goed)
 6Ø RETURN

1ØØ END
```

2 GOTO in plaats van GOSUB gebruikt.

```
 1Ø GOTO 1ØØ

1ØØ PRINT "DIT IS REGEL 1ØØ"        (fout)

12Ø RETURN
```

RETURN is alleen mogelijk in combinatie met een GOSUB-opdracht. Dus:

```
 1Ø GOSUB 1ØØ

1ØØ PRINT "DIT IS REGEL 1ØØ"        (goed)

12Ø RETURN
```

3 Achter de laatste GOSUB moet altijd een regel staan. Anders springt een RETURN terug naar niets en dat kan de computer niet.

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten een*  *te geven)*

---

opgaven

1 De bedoeling is dat alle getallen X worden geprint. Waarom doet hij dat niet?

```
 1Ø X = 7
 2Ø GOSUB 2ØØ
 3Ø X = 9
 4Ø GOSUB 2ØØ
 5Ø X = 12
 6Ø GOSUB 2ØØ
 7Ø GOTO 3ØØ
2ØØ PRINT X
3ØØ END
```

2 De bedoeling is dat alle getallen X worden geprint. Waarom doet hij dat niet?

```
 1Ø X = 7
 2Ø GOTO 2ØØ
 3Ø X = 9
 4Ø GOTO 2ØØ
 5Ø X = 12
 6Ø GOTO 2ØØ
 7Ø GOTO 3ØØ
2ØØ PRINT X
21Ø RETURN
3ØØ END
```

3 Wanneer kun je beter een GOSUB en RETURN gebruiken dan een GOTO?

DE REM-OPDRACHT

REM

Via REM kun je mededelingen in het programma zetten (voor jezelf)

In BASIC:	*In Nederlands:*
`1Ø REM PROGRAMMA 1`	*10 mededeling PROGRAMMA 1*

REM is een afkorting van het Engelse woordje REMark en betekent opmerking. Wat achter REM staat wordt door het programma niet uitgevoerd. Het is meer een handigheidje. Je weet hier bijvoorbeeld dat het om programma 1 gaat.

REM doet in het programma niets. De REM-opdracht wordt overgeslagen. Je kunt er dus alles wat je wilt achter zetten, bijvoorbeeld mededelingen of een deel van het programma mee uitleggen. Vooral als je een programma aan een ander wilt geven, kun je achter REM mededelingen zetten. Kijk maar eens naar het volgende programma:

```
typ    1Ø REM DIT PROGRAMMA IS VAN AAD EN BEN
typ    2Ø REM VRAAG OM EEN VOORNAAM
typ    3Ø PRINT "WAT IS JE VOORNAAM?"
typ    4Ø INPUT V$
typ    5Ø REM VRAAG OM EEN ACHTERNAAM
typ    6Ø PRINT "WAT IS JE ACHTERNAAM?"
typ    7Ø INPUT A$
typ    8Ø REM ZET VOOR- EN ACHTERNAAM OP
          HET TV-SCHERM
typ    9Ø PRINT "DAG ";V$; " ";A$
typ   1ØØ PRINT "LEUK MET JE KENNIS GEMAAKT TE
          HEBBEN"
typ   11Ø END

typ   RUN
```

Je ziet dat de opmerking achter de REM-opdracht door de computer niet wordt uitgevoerd. Waarom is het dan nodig dat er REM-opdrachten in staan?
Met REM-opdrachten kun je het programma veel duidelijker maken. Zowel voor jezelf als voor anderen. Als je het programma later nog eens wilt bekijken dan weet je door de REM-opdrachten waar alles voor dient. Ook als je het programma aan een ander wilt geven kun je via REM-opdrachten mededelingen in het programma zetten. Vooral voor lange programma's zijn REM-opdrachten handig. De programma's zijn dan veel duidelijker. Let er wel op dat er hoogstens zes regels tekst achter REM komen. Meer kan niet.

Kijk maar eens naar het volgende programma. Het is precies hetzelfde als het programma dat we net gemaakt hebben. Alleen de REM-opdrachten zijn er uitgelaten.

```
typ    3Ø PRINT "WAT IS JE VOORNAAM?"
typ    4Ø INPUT V$
typ    6Ø PRINT "WAT IS JE ACHTERNAAM?"
typ    7Ø INPUT A$
typ    9Ø PRINT "DAG ";V$; " ";A$
typ   1ØØ PRINT "LEUK MET JE KENNIS GEMAAKT TE
           HEBBEN"
typ   11Ø END
```

Via REM kun je mededelingen in het programma zetten (voor jezelf)

herstel

1 De REM-opdracht gaat niet zo snel fout. Zo'n zes regels achter de REM-opdracht wordt als een mededeling gezien en precies zo in de LISTing van een programma opgenomen.

opgaven

1 Maak programma LEEFTIJD op blz. 58 mooier door met REM-opdrachten er mededelingen tussen te zetten. Kies mededelingen die anderen kunnen gebruiken.

2 Doe dit ook voor programma NAAM op blz. 59

DE IF ... THEN-OPDRACHT

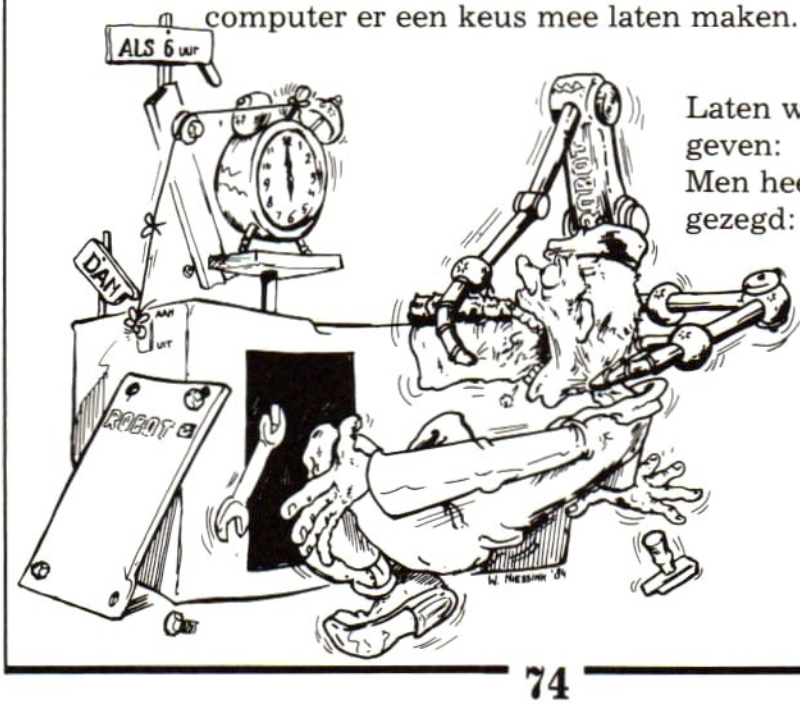
IF ... THEN

Met IF ... THEN moet de computer een keuze maken

In BASIC:	*In Nederlands:*
`1Ø IF X = 6 THEN GOTO 1ØØ`	*10 als het 6 is ga dan naar regel 100*
`1ØØ END`	*100 einde*

IF en THEN zijn twee Engelse woordjes. IF betekent ALS en THEN betekent DAN. Zou de man die BASIC heeft gemaakt een Nederlander zijn geweest, dan had hij er vast ALS ... DAN van gemaakt. Maar die man sprak Engels en dus is het IF ... THEN geworden.

De IF ... THEN opdracht is heel handig. Je kunt de computer er een keus mee laten maken.



Laten we een voorbeeld geven:
Men heeft tegen je gezegd: ALS het zes uur is DAN moet je gaan eten. Anders kun je doorgaan met waar je mee bezig bent.

Programma **TIJD:**

```
typ  1Ø PRINT "HOE LAAT IS HET ?"
typ  2Ø PRINT "IN UREN"
typ  3Ø INPUT T
typ  4Ø IF T = 6 THEN PRINT "SNEL NAAR HUIS"
typ  8Ø END

typ  RUN
```



– *Typ de tijd. Bijv. 5*

Alleen maar als T = 6 uur dan wordt de opdracht uitgevoerd.
Als T niet 6 (uur) is dan wordt de opdracht niet uitgevoerd.
Die regel wordt dan overgeslagen en de computer gaat de volgende regel uitvoeren.

Tussen IF ... THEN kunnen de volgende vergelijkingen gemaakt worden:

```
typ  4Ø IF T = 6 THEN PRINT "HET IS 6 UUR"
typ  RUN

typ  5Ø IF T > 6 THEN PRINT "HET IS AL 6 UUR
                            GEWEEST"
typ  RUN

typ  6Ø IF T < 6 THEN PRINT "HET IS VOOR 6 UUR"
typ  RUN

typ  7Ø IF T <> 6 THEN PRINT "HET IS GEEN 6 UUR"
typ  RUN
```

Dus onthouden	=	*gelijk aan*	$6 = 6$
	>	*groter dan*	$8 > 6$
	<	*kleiner dan*	$5 < 6$
	< >	*ongelijk aan*	$3 <> 6$

Er zijn nog meer vergelijkingen, maar die hoef je nog niet te onthouden.

Programma **GELUKSGETAL:**

```
typ  1Ø PRINT "RAAD MIJN GELUKSGETAL"
typ  2Ø PRINT "TYP EEN GETAL TUSSEN 1 EN 1Ø"
typ  3Ø INPUT X
typ  4Ø IF X = 7 THEN GOTO 7Ø
typ  5Ø PRINT "PROBEER HET NOG EENS"
typ  6Ø GOTO 2Ø
typ  7Ø PRINT "JA, 7 IS MIJN GELUKSGETAL"
typ  8Ø END

typ  RUN
```

Als je een foutmelding krijgt dan weet je dat de computer je niet begrijpt. Eerst de vergissing herstellen en dan nog een keer RUN typen. Lukt het niet, kijk dan even in Herstel.

Programma **MENEER of MEVROUW:**

```
typ  1Ø PRINT "BEN JE EEN MAN OF EEN VROUW ?"
typ  2Ø PRINT "TYP EEN M OF EEN V"
typ  3Ø INPUT M$
typ  4Ø IF M$ = "M" THEN M$ = " MENEER "
typ  5Ø IF M$ = "V" THEN M$ = " MEVROUW "
typ  6Ø PRINT "WAT IS JE VOORNAAM ?"
typ  7Ø INPUT V$
typ  8Ø PRINT "DAG "; M$ ; " " ; V$
typ  9Ø END

typ  RUN
```

Je ziet dat er bij INPUT weer een stringteken ($) achter de letters staat. Je hebt het nodig omdat je een naam vraagt.

Met IF ... THEN moet de computer een keuze maken

herstel

1 Aanhalingstekens vergeten?

```
4Ø IF M$ = M THEN                    (fout)
```



Aanhalingstekens vergeten.

Het stringtekentje $ vraagt om een naam of letters en die moeten tussen aanhalingstekens. Herstel die regel.

typ `4Ø IF M$="M" THEN M$="MENEER"` *(goed)*

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ `RUN` *(niet vergeten een*  *te geven)*

opgaven

1 Verander het programma TIJD op blz. 75 zodanig dat je om 5 uur naar huis moet.

2 Wat betekent het = teken ?

3 Wat betekent het > teken ?

4 Wat betekent het < teken ?

5 Wat betekent het < > teken ?

6 Verander het programma GELUKSGETAL op blz. 76 Laat nu 3 het geluksgetal van de computer zijn.

DE FOR ... NEXT-OPDRACHT

FOR ... NEXT

Met de FOR ... NEXT opdracht kun je de opdracht die er tussen staat herhalen

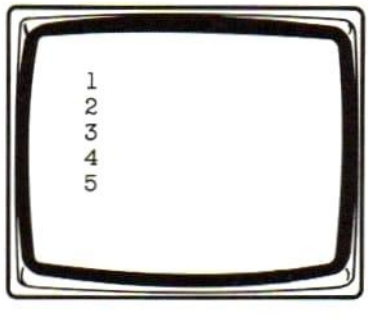
In BASIC:	*In Nederlands:*
`1Ø FOR X = 1 TO 4`	*10 doe van 1 tot 4 keer het volgende*
`2Ø PRINT "BEN"`	*20 zet BEN op TV-scherm*
`3Ø NEXT X`	*30 ga weer terug*

Deze opdracht wordt 4 keer herhaald.

Voorbeeld:

typ `1Ø FOR J = 1 TO 5`
typ `2Ø PRINT J`
typ `3Ø NEXT J`
typ `4Ø END`

typ `RUN`



Je ziet dat de FOR ... NEXT-lus een soort herhaling is van de opdracht die er tussen staat. FOR (voor) geeft het begin aan en NEXT (volgende) stuurt de computer terug naar FOR net zolang tot hij klaar is. 'FOR J = 1 TO 5' betekent dat er vijf keer een opdracht moet worden uitgevoerd.

Je kunt een lus ook goed als een pauze in je programma gebruiken.

```
typ  10 PRINT "BEGIN PAUZE"
typ  20 FOR PAUZE = 1 TO 1000
typ  30 NEXT PAUZE
typ  40 PRINT "EINDE PAUZE"
typ  50 END

typ  RUN
```

Wat er tussen regel 20 en regel 30 staat wordt 1000 keer uitgevoerd. Maar tussen regel 20 en 30 staat geen opdracht.
De computer telt eigenlijk alleen 1000 keer en dat kost tijd. Het is net als met verstoppertje spelen. De computer moet tot 1000 tellen voordat hij mag gaan zoeken. Als de computer aan het tellen is, heb jij even een pauze. Je kunt de pauze korter of langer maken door de computer korter of langer te laten tellen.

```
FOR PAUZE = 1 TO 100    : NEXT PAUZE    is 0.5 sec.
FOR PAUZE = 1 TO 1000   : NEXT PAUZE    is 2   sec.
FOR PAUZE = 1 TO 10000  : NEXT PAUZE    is 22  sec.
```

De getallen die hier worden genoemd, kunnen per computer verschillen. Vaak is het een kwestie van proberen hoe lang je de computer moet laten tellen.

Voorbeeld:

```
typ  100 FOR PAUZE = 1 TO 10
typ  110 NEXT PAUZE
typ  120 PRINT "DAG ";
typ  130 FOR PAUZE = 1 TO 100
typ  140 NEXT PAUZE
typ  150 PRINT "HOE ";
typ  160 FOR PAUZE = 1 TO 1000
typ  170 NEXT PAUZE
```

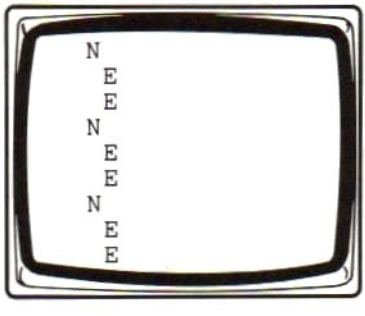
```
typ  18Ø PRINT "GAAT HET ";
typ  19Ø FOR PAUZE = 1 TO 1ØØØØ
typ  2ØØ NEXT PAUZE
typ  21Ø PRINT "MET JOU ?"
typ  22Ø END

typ  RUN
```

Bij deze voorbeelden staat er keurig een lus (bijv. regel 100 en 110) en daarna een andere lus (bijv. regel 130 en 140). Maar je kunt de ene lus ook in de andere zetten. Wat je moet onthouden is dat de computer zoekt welke FOR en NEXT het dichtst bij elkaar staan. Deze maakt hij ook het eerst af.
Kijk maar eens naar het volgende voorbeeld.

```
typ  3ØØ FOR A = 1 TO 3
typ  31Ø PRINT "N"
typ  32Ø FOR B = 1 TO 2
typ  33Ø PRINT " E"
typ  34Ø NEXT B
typ  35Ø NEXT A
typ  36Ø END

typ  RUN
```



Goed opletten dat je ze niet door elkaar haalt, want dan loopt je programma niet goed.

Programma **STRAFWERK:**

```
typ  1Ø AANTAL = 1ØØ
typ  2Ø FOR S = 1 TO AANTAL
typ  3Ø PRINT "IK MOET STILZITTEN"
typ  4Ø NEXT S
typ  5Ø END
```

Het voordeel van deze lus is, dat je het aantal makkelijk kunt veranderen. Dus als je de volgende keer 200 strafregels moet schrijven dan verander je alleen: aantal = 200.

Met de FOR ... NEXT opdracht kun je de opdracht die er tussen staat herhalen

herstel

1 Heeft elke NEXT een FOR?

```
1Ø FOR J = 1 TO 5
2Ø PRINT J
```
(fout)

```
1Ø FOR J = 1 TO 5
2Ø PRINT J
```
(goed)

```
3Ø NEXT J
```

2 Staan de goede lussen bij elkaar?

```
1Ø FOR A = 1 TO 3
2Ø FOR B = 1 TO 3
3Ø NEXT A
4Ø NEXT B
```
(fout)

Lus A loopt door lus B heen. Kijk maar:



De lussen mogen niet door elkaar lopen.

```
1Ø FOR A = 1 TO 3
2Ø FOR B = 1 TO 3
3Ø NEXT B                                    (goed)
4Ø NEXT A
```

Geef zonodig een LIST om te kijken of het programma nu goed in het geheugen zit. Geef dan de computer de opdracht om jouw programma uit te voeren.

typ RUN *(niet vergeten een*  *te geven)*

opgaven

1 Schrijf met behulp van een FOR ... NEXT lus een programma dat de getallen 1 tot en met 10 op het TV-scherm zet.

2 Zet 10 keer jouw naam op het TV-scherm met behulp van FOR ... NEXT.

3 Verander dit programma door van een FOR ... NEXT lus gebruik te maken.

typ `1Ø J = Ø`
typ `2Ø J = J + 1`
typ `3Ø PRINT J`
typ `4Ø IF J = 1Ø THEN GOTO 6Ø`
typ `5Ø GOTO 2Ø`
typ `6Ø END`

4 Met FOR ... NEXT kun je ook goed een programma schrijven dat je bijvoorbeeld laat zien wat de tafel van twee is, maar het kan ook voor de tafel van 5 of van 25 of van ...
Probeer het maar eens, achterin hebben wij er een mooi voorbeeld van.

DE READ- EN DATA-OPDRACHTEN

READ ... DATA

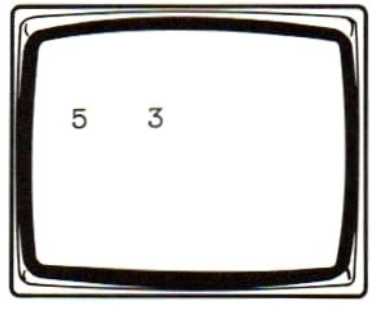
READ ... DATA zijn opdrachten waarmee je zegt Lees (READ) de Gegevens (DATA) in

In BASIC:	*In Nederlands:*
`1Ø READ A`	*10 lees A (een getal)*
`5Ø DATA 7`	*50 getal is 7*
`1ØØ READ A$`	*100 lees A$ (een naam)*
`15Ø DATA PIET`	*150 naam is PIET*

READ is het Engelse woord voor lees. DATA is het Engelse woord voor gegevens. READ zoekt de gegevens op die in een DATA-opdracht staan. Net zoals bij INPUT, is een $ voor namen (of getallen waar je niet mee hoeft te rekenen). Het verschil tussen INPUT en READ is, dat INPUT gegevens vraagt aan de gebruiker. READ haalt de gegevens ergens anders uit het programma. Als er in een programma een READ opdracht staat, gaat het programma zoeken naar een regel waar DATA staat. DATA is een ander woord voor gegevens. Met READ en DATA kun je veel leuke dingen doen.

typ `1Ø READ A, B`
typ `2Ø PRINT A, B`
typ `3Ø DATA 5, 3`
typ `4Ø END`

typ `RUN`



De READ-opdracht in regel 10 laat de computer zoeken naar een regel waar DATA staat. Het programma springt even naar regel 30. In de vakjes A en B worden dan de getallen 5 en 3 gestopt. Dan gaat het programma weer terug om de volgende regel (20) uit te voeren. Deze regel 20 zet de gegevens op het TV-scherm. Daarna is het programma klaar.
Als je net zoals hier om twee dingen vraagt (A en B), dan moeten er ook twee dingen bij DATA staan. Staat er bijvoorbeeld maar één of staan er drie, dan gaat het fout.

```
typ   1Ø DATA 5, 3
typ   3Ø READ A, B
typ  1ØØ PRINT A, B
typ  3ØØ END

typ   RUN
```

Je ziet dat het er niet toe doet, of eerst de DATA komt en dan de READ-opdracht, of andersom. Meestal wordt wel eerst de READ-opdracht gekozen en daarna de DATA-opdracht. De DATA-opdracht staat in grote programma's vaak aan het einde.

Je kunt ook namen en telefoonnummers in de DATA-opdracht zetten. Daarmee kun je dan een telefoonlijstje maken van je vrienden.

```
typ   1Ø READ N$, T$
typ   2Ø PRINT N$, T$
typ   3Ø DATA PIET, Ø3Ø-876321
typ   4Ø END

typ   RUN
```



Je ziet dat je hier het stringteken ($) moet gebruiken voor telefoonnummers.

Je begrijpt dat we met READ en DATA hele leuke programma's kunnen maken. Wat dacht je hiervan?

Programma **TELEFOONNUMMERS:**

```
typ   1Ø READ N$, T$
typ   2Ø IF N$ = "STOP" THEN GOTO 5ØØ
typ   3Ø PRINT N$, T$
typ   4Ø GOTO 1Ø

typ  1ØØ DATA PIET, Ø3Ø-123456
typ  11Ø DATA ELS, Ø1Ø-654321
typ  12Ø DATA HANS, Ø7Ø-112233
typ  13Ø DATA BERT, Ø2Ø-776655

typ  49Ø DATA STOP, STOP
typ  5ØØ END

typ  RUN
```

Hoe werkt dit programma nu?

`1Ø READ N$, T$`	*Lees de gegevens. Zoek regel waarin DATA staat.*
`2Ø IF N$ = "STOP" THEN GOTO 5ØØ`	*Als in vakje N$ 'STOP' komt, ga dan naar regel 500.*
`3Ø PRINT N$, T$`	*Zet de gegevens op TV.*
`4Ø GOTO 1Ø`	*Ga weer naar regel 10 om via READ de volgende gegevens in te lezen.*

```
100 DATA PIET, 030-123456
110 DATA ELS, 010-654321
120 DATA HANS, 070-112233
130 DATA BERT, 020-776655
```

Hier staan de gegevens. Door de READ-opdracht op regel 10 worden er telkens twee gelezen.

Tussen regel 130 en 490 kun je nog veel meer namen en telefoonnummers zetten.

```
490 DATA STOP, STOP
```

Einde van de gegevens.

```
500 END
```

Einde van programma.

READ ... DATA zijn opdrachten waarmee je zegt Lees (READ) de Gegevens (DATA) in

herstel

1 Een komma te veel.

```
10 DATA 5, 10
20 READ A, B,                    (fout)
```

De komma achter B is te veel.

2 Vergeten de DATA-opdracht te geven:

```
1Ø PRINT 7, 1Ø
2Ø READ A, B                     (fout)
```

```
1Ø DATA 7, 1Ø                    (goed)
2Ø READ A, B
3Ø PRINT A, B
```

3 Te veel gegevens achter de DATA-opdracht.

```
1Ø DATA 5, 1Ø, 9
2Ø READ A, B                     (fout)
```

Achter READ moeten net zoveel letters staan als getallen achter DATA. Als er drie getallen staan, moeten er ook drie letters zijn.

```
1Ø DATA 5, 1Ø, 9
2Ø READ A, B, C                  (goed)
```

4 Komma in plaats van een punt.

```
1Ø READ A                        (fout)
2Ø DATA 25, 5Ø
```

Als je een komma achter de 25 zet, dan wordt 50 het volgende getal. De computer denkt dat het twee getallen zijn, 25 en 50. Met READ A vertel je de computer dat hij één getal moet lezen. Dus dat gaat fout.

```
1Ø READ A
2Ø DATA 25.5Ø                    (goed)
```

5 Naam en getal verwisseld.

```
1Ø READ A, N$                    (fout)
2Ø DATA PIET, 17
```

Met READ vraag je eerst om een getal voor vakje A. Maar in DATA staat eerst een naam.

```
1Ø READ A, N$
2Ø DATA 17, PIET                 (goed)
```

opgaven

1 Je vrienden Jan, Klaas en Wim zijn 10, 12 en 15 jaar oud. Maak een programma met READ en DATA.
Laat het programma de naam en leeftijd van je vrienden op het TV-scherm zetten.

2 Je hebt een vriendin Mariska van 14 jaar erbij gekregen. Pas het programma aan.

3 Maak een telefoonlijst van je beste vrienden. Gebruik de READ- en DATA-opdracht.

4 Doe nu hetzelfde voor vrienden, familie en kennissen. (Achterin staat hoe je dit programma op cassette kunt bewaren.)

5 Wat moet er eerst in een programma worden gebruikt, READ of DATA?

DE RND-OPDRACHT

RND

RND is een opdracht waarmee je een verrassingsgetal van de computer krijgt

In BASIC:	*In Nederlands:*
`1Ø X = RND (1)`	*10 zet in vakje X een verrassingsgetal*

Vooral voor veel spelletjes is het handig als je een soort dobbelsteen hebt. Iets dat getallen geeft die je van tevoren niet wist, die een verrassing zijn. Ook bij een dobbelsteen weet je van te voren niet welk getal er boven zal liggen. RND is een soort van computerdobbelsteen, dit geeft telkens een verrassingsgetal. Dit getal ligt tussen 0 en 1.
Typ het volgende programma maar in, dat zal het je laten zien.

typ `2Ø X = RND (1)`
typ `3Ø PRINT X`
typ `5Ø END`

typ `RUN`

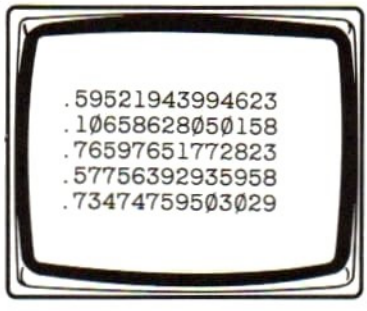


De computer heeft hier .59521943994623 als verrassingsgetal genomen. En als je nog eens RUN intypt, komt er weer hetzelfde getal.

Volgende voorbeeld:

```
typ   10 FOR I = I TO 5
typ   20 X = RND (1)
typ   30 PRINT X
typ   40 NEXT I

typ   RUN
```



5 verrassingsgetallen

Met deze getallen kunnen we nog niet zoveel. Veel mooier is het als we getallen tussen 1 en 10 kunnen krijgen. Dat gaat op de volgende manier.

```
typ   10 X = RND (1)
typ   20 PRINT X
typ   30 Y = 10 * X
typ   40 PRINT Y
typ   50 Z = INT (Y)
typ   60 PRINT Z
typ   70 END

typ   RUN
```

Wat hebben we nu gedaan:

Een RND getal is bijvoorbeeld .59521943994623
Dat nemen we 10 keer 10 keer
5.9521943994623
Afkappen met INT is 5

INT maakt er een heel getal van. Bijvoorbeeld:

INT (3.757865) = 3
INT (2.489265) = 2

Zo, nu kunnen we de computer een getal tussen 1 en 10 laten kiezen. Het kan nog iets mooier en dat staat hieronder:

```
X = INT (RND(1) * 1Ø    +1) geeft getal van 1 t/m 10
X = INT (RND(1) * 1ØØ   +1) geeft getal van 1 t/m 100
X = INT (RND(1) * 1ØØØ +1) geeft getal van 1 t/m 1000
```

Let er goed op dat de verrassingsgetallen wel altijd in dezelfde volgorde komen.

Programma **RAAD EEN GETAL:**

```
typ  1Ø PRINT "RAAD EEN GETAL VAN 1 T/M 1Ø"
typ  2Ø X = INT (RND(1) * 1Ø + 1)
typ  3Ø INPUT G
typ  4Ø IF G = X THEN GOTO 7Ø
typ  5Ø PRINT "NEE, NOG NIET GERADEN"
typ  6Ø GOTO 3Ø
typ  7Ø PRINT "JA, JE HEBT MIJN GETAL GERADEN"
typ  8Ø PRINT "HET GETAL IS"; X
typ  9Ø END

typ  RUN
```

Gaat alles goed? Dan heb je een spelletje gemaakt.

RND is een opdracht waarmee je een verrassingsgetal van de computer krijgt

herstel

1 Een haakje te weinig.

```
Bijv. 1Ø X = INT (RND (1) * 1Ø + 1
```



Er moeten namelijk aan de linkerkant even veel haakjes staan als aan de rechterkant en dat klopt hier niet.

opgaven

1 Maak het programma GETALLEN RADEN op blz. 68 leuker door op de regels 20, 40 en 60 een RND-getal te nemen van 1 t/m 10.

2 Hoe kun je een RND(random)-getal van 1 t/m 5 krijgen?

3 Hoe kun je een RND-getal van 1 t/m 250 krijgen?

4 Verander het programma RAAD EEN GETAL van 1 t/m 10 zó, dat hij dobbelsteengetallen geeft, dus van 1 t/m 6. (Zie voor een mooi voorbeeld blz. 108)

FOUTEN IN JE PROGRAMMA'S

Als je in je programma een fout maakt, ontdek je dat vanzelf als je met het programma wilt gaan werken. Typ het onderstaande programma maar eens over. Typ als je fouten in het programma ziet, ook die precies zo over. Dus je typt PINT en geen PRINT.

```
1Ø PINT "DIT GAAT NIET GOED"
2Ø END
```

In dit programma zit een fout, er staat PINT in plaats van PRINT. Je ontdekt deze fout als je het programma uitvoert. Typ maar eens RUN.



SYNTAX ERROR

Dit betekent dat er een fout staat in regel 10. Dat klopt, want daar staat PINT. Verder vertelt hij wat voor soort fout het is. Het is een SYNTAX fout. Een 'SYNTAX ERROR' wil zeggen dat je een woord niet goed getypt hebt, of je hebt bijvoorbeeld de aanhalingstekens " en " vergeten. DE SYNTAX ERROR is één van de meest voorkomende fouten.

We zullen hier nog een paar andere fouten bespreken.

EXTRA IGNO-RED

Dit gebeurt nogal eens bij INPUT. In de volgende regels vraagt de computer een getal:

```
1Ø PRINT "HOE OUD BEN JE"
2Ø INPUT L
```

Je typt bijvoorbeeld 15,6 in omdat je 15 en een half bent. De computer zal na RUN zeggen: EXTRA IGNORED, dat betekent dat hij alleen 15 onthoudt, en het extra getal 6 'vergeet' (ignored betekent: vergeten). Door de komma denkt hij dat je twee getallen aan hem geeft, namelijk 15 en 6. Wil je dat hij 15 1/2 opneemt, dan moet je 15.6 intypen. Een punt in plaats van een komma. Probeer het maar eens.

REDO FROM START

Typ het programma bij EXTRA IGNORED nog eens in. Als je de computer antwoordt dat je TIEN bent; als je het dus in letters in plaats van cijfers (10) doet, geeft hij deze fout. Hij verwacht namelijk cijfers en geen letters. Wil je letters gebruiken, dan moet je om INPUT L$ vragen. Het $ teken geeft aan dat de computer een antwoord in letters kan verwachten. Als je nu 10 typt, gaat het programma verder waar het was gebleven.

OUT OF DATA

Dit komt bij de READ-opdracht voor. Het volgende programma geeft zo'n foutmelding.

```
1Ø READ A, B, C
2Ø DATA 6, 7
```

Je vraagt hem hier drie dingen te lezen: A, B en C. Er staan bij data echter slechts twee getallen: 6 en 7. De computer weet dat hij nog een getal moet geven, maar hij heeft niks meer bij DATA staan, DATA is op.

UNDEFINED LINE NUMBER

Typ het volgende programma eens in:

```
1Ø PRINT "HOE OUD BEN JE"
2Ø INPUT L
3Ø IF L > 3Ø THEN GOTO 8Ø
4Ø PRINT "DAN BEN JE JONG"
5Ø GOTO 7Ø
6Ø PRINT "DAN BEN JE NIET OUD"
7Ø END
```

Als je RUN intypt en je antwoordt de computer dat je bijvoorbeeld 88 bent, geeft hij als fout UNDEFINED LINE NUMBER. Weet je waarom? Inderdaad in regel 30

staat dat wanneer je als antwoord geeft dat je ouder bent dan 30, de computer naar regel 80 moet gaan. Maar regel 80 is er niet, dat had 60 moeten zijn. De computer moet iets doen wat hij niet kent (undefined is: onbekend).

Dit zijn de belangrijkste fouten. Je zult ze allemaal wel eens maken. Je moet maar denken, van je fouten kun je leren. En je weet nu hoe je de meest voorkomende kunt verbeteren.

DEEL 3

PROGRAMMA'S

REKENEN
MET DE COMPUTER

Heb je vaak (huis)werk waarbij je moet rekenen? Als het antwoord hier ja op is, dan is het volgende iets voor jou. Je kunt de computer heel goed gebruiken bij het maken van je rekenwerk. De computer is een geduldige leraar die je goed kan helpen met leren rekenen of rekensommen maken. Optellen, aftrekken, delen of vermenigvuldigen, het maakt de computer niets uit. Grote of kleine getallen, veel sommen of weinig sommen; de computer doet het voor je. Het programma dat je daarvoor nodig hebt, kun je zelf maken. Laten we eens een rekenprogramma stap voor stap opzetten. Je moet dan iedere keer als er staat typ, bijvoorbeeld:

typ `1Ø PRINT "REKENEN"`

die regel precies zo intypen. Tussen die regels zullen we stap voor stap kijken waar die regel voor nodig is en wat je verder moet doen. Aan het einde van die regels heb je een compleet rekenprogramma wat je net zo vaak kunt gebruiken als je maar wilt. Aan het einde zullen we het programma nog eens in zijn geheel geven.
Dus als je niet alle stappen wilt volgen, kun je gelijk het hele programma intypen. Je kunt altijd daarna nog kijken welke stappen je gemaakt hebt en waar ze voor nodig zijn.
Zo, laten we beginnen.

PROGRAMMA REKENEN:

We willen een programma maken waarmee we kunnen optellen, aftrekken, delen en vermenigvuldigen.
Allereerst typ je dan de regels waarmee je de computer vraagt of je wilt optellen, aftrekken, delen of vermenigvuldigen.

```
typ  1Ø PRINT "*** REKENEN ***"
typ  2Ø PRINT "TYP 1 VOOR OPTELLEN"
typ  3Ø PRINT "TYP 2 VOOR AFTREKKEN"
typ  4Ø PRINT "TYP 3 VOOR DELEN"
typ  5Ø PRINT "TYP 4 VOOR VERMENIGVULDIGEN"
```

Na deze regels moeten we ervoor zorgen dat de keuze in een vakje wordt opgeborgen. Laten we dat vakje de naam KEUZE geven.

```
typ  6Ø INPUT KEUZE
```

Regel 70 slaan wij even over, die gebruiken we straks. Zo nu kunnen we in dat vakje het getal 1, 2, 3 of 4 zetten. De getallen waarmee we gaan rekenen moeten ook in vakjes komen. In het eerste vakje een getal en in het tweede vakje een getal. Die getallen kunnen we daarna bij elkaar optellen, van elkaar aftrekken, met elkaar vermenigvuldigen of door elkaar delen. Laten we de vakjes EERSTE en TWEEDE noemen.

```
typ   8Ø PRINT "GEEF NU TWEE GETALLEN OP"
typ   9Ø PRINT "EERSTE GETAL = "
typ  1ØØ INPUT EERSTE
typ  12Ø PRINT "TWEEDE GETAL = "
typ  13Ø INPUT TWEEDE
```

We hebben drie vakjes gemaakt waarin de keuze zit (optellen, aftrekken, delen of vermenigvuldigen), het eerste getal en een vakje waarin het tweede getal zit.

	KEUZE	*EERSTE*	*TWEEDE*
Bijv.	*1*	*8*	*4*

Nu moeten we de computer laten kijken of er in het vakje KEUZE een 1, 2, 3 of 4 zit

want is het een 1 dan moet hij optellen
is het een 2 dan moet hij aftrekken
is het een 3 dan moet hij delen
is het een 4 dan moet hij vermenigvuldigen.

Zoals je weet kun je daar de IF ... THEN opdracht voor gebruiken. IF ... THEN betekent ALS ... DAN.

Dus ALS 1 DAN +
ALS 2 DAN –
ALS 3 DAN / (delen)
ALS 4 DAN * (vermenigvuldigen)

Het getal 1, 2, 3 of 4 staat in het vakje KEUZE. En de getallen die we bijvoorbeeld willen optellen staan in de vakjes EERSTE en TWEEDE.

Als we een sommetje op papier maken ziet dat er bijvoorbeeld zo uit:

12 = 8 + 4

Getal 8 zit in het EERSTE vakje en getal 4 zit in het TWEEDE vakje. Waar moet het getal 12, de som, dan in? Daar moeten we ook een vakje voor maken en dat kunnen we SOM noemen.

We hebben nu de volgende vakjes:

	KEUZE	*SOM*	*EERSTE*	*TWEEDE*
Bijv.	*1*	*12*	*8*	*4*

We moeten nu de keuze door de computer laten maken. Als de computer het getal van de keuze (bijv. 1) gevonden heeft, moet hij de optelling gaan maken. Laten we het eens uitschrijven:

als KEUZE = 1 dan SOM = EERSTE + TWEEDE
als KEUZE = 2 dan SOM = EERSTE – TWEEDE
als KEUZE = 3 dan SOM = EERSTE / TWEEDE (delen)
als KEUZE = 4 dan SOM = EERSTE * TWEEDE (verm.)

Nu nog iets aanpassen zodat de computer het begrijpt:

```
typ  14Ø IF KEUZE = 1 THEN SOM = EERSTE + TWEEDE
typ  15Ø IF KEUZE = 2 THEN SOM = EERSTE - TWEEDE
typ  16Ø IF KEUZE = 3 THEN SOM = EERSTE / TWEEDE
typ  17Ø IF KEUZE = 4 THEN SOM = EERSTE * TWEEDE
```

Zo, we zijn nu bijna klaar. De KEUZE is gemaakt. Het EERSTE en TWEEDE getal hebben we en de SOM is gemaakt.
We moeten het antwoord nog op het TV-scherm krijgen. Dat gaat als volgt:

typ `3ØØ PRINT "ANTWOORD = " ; SOM`

We zouden het programma kunnen stoppen. Als je een som moet maken of je wilt wat oefenen met rekenen dan kun je dit programma gebruiken. Iedere keer typ je RUN en het programma helpt je.
Het programma is alleen nog onhandig. Je hebt meestal meer dan één som. Soms wel tien of nog meer. Dan is iedere keer RUN typen vervelend. We kunnen het programma heel gemakkelijk handiger maken door de computer iedere keer weer de volgende som te laten vragen. Dit gaat als volgt:

typ `31Ø GOTO 8Ø`

GOTO 80 betekent ga naar regel 80. En daar staat ' geef nu de twee getallen op'. Je kunt dan gewoon met de volgende som beginnen.
Door de computer nu iedere keer opnieuw te laten beginnen heb je bereikt dat je een aantal sommen achter elkaar kunt maken. Je kunt nu maar doorgaan en doorgaan.
Je moet zelfs doorgaan omdat het programma niet weet hoe te stoppen. Als je bijvoorbeeld tien sommen hebt gemaakt en je wilt stoppen, hoe moet dat dan? Je kunt altijd de computer uitzetten of een stoptoets gebruiken, maar dat is niet mooi. Nee, laten we de mogelijkheid om te stoppen netjes in het programma opnemen.

typ `7Ø PRINT "JE KUNT STOPPEN DOOR TE TYPEN 999"`

Dit moeten we iedere keer controleren. Bijvoorbeeld:

als 999 dan stoppen

Het beste kunnen we dat doen na de vraag om het eerste getal (regel 90). Komt in het vakje EERSTE 999 dan wil

je stoppen. In computertaal wordt dit:

```
typ  11Ø IF EERSTE = 999 THEN GOTO 4ØØ
```

En op regel 400 moeten we dan het programma stoppen. Want GOTO 400 betekent 'ga naar regel 400'.

```
typ  4ØØ END
```

Zo, dit is het hele programma. Als je overal waar typ voor staat hebt ingetypt dan ben je nu klaar met het programma. Hier volgt het hele programma achter elkaar zonder commentaar erbij.

Programma **REKENEN**

```
typ   1Ø PRINT "*** REKENEN ***"
typ   2Ø PRINT TYP 1 VOOR OPTELLEN"
typ   3Ø PRINT "TYP 2 VOOR AFTREKKEN"
typ   4Ø PRINT "TYP 3 VOOR DELEN"
typ   5Ø PRINT "TYP 4 VOOR VERMENIGVULDIGEN"
typ   6Ø INPUT KEUZE
typ   7Ø PRINT "JE KUNT STOPPEN DOOR TE TYPEN 999"
typ   8Ø PRINT "GEEF NU TWEE GETALLEN OP"
typ   9Ø PRINT "EERSTE GETAL = "
typ  1ØØ INPUT EERSTE
typ  11Ø IF EERSTE = 999 THEN GOTO 4ØØ
typ  12Ø PRINT "TWEEDE GETAL = "
typ  13Ø INPUT TWEEDE
typ  14Ø IF KEUZE = 1 THEN SOM = EERSTE + TWEEDE
typ  15Ø IF KEUZE = 2 THEN SOM = EERSTE - TWEEDE
typ  16Ø IF KEUZE = 3 THEN SOM = EERSTE / TWEEDE
typ  17Ø IF KEUZE = 4 THEN SOM = EERSTE * TWEEDE
typ  3ØØ PRINT "ANTWOORD = " ; SOM
typ  31Ø GOTO 8Ø
typ  4ØØ END
```

Het programma is klaar en je kunt er mee rekenen. Je kunt het programma ook nog uitbreiden. Heel leuk is het om eerst zelf een antwoord te bedenken en dan de computer te laten controleren of jouw antwoord goed is.

Je gebruikt de computer dan om jou te overhoren. En je weet dat de computer heel veel geduld heeft en nooit kwaad wordt. Je kunt er net zo lang over doen als je zelf wilt en net zo vaak als je wilt.

TOETSENBORD

```
1ØØ REM EEN KLEIN PROGRAMMA OM
11Ø REM VERTROUWD TE RAKEN MET
12Ø REM HET TOETSENBORD
13Ø REM
14Ø REM BAARDA EN VAN LONDEN
15Ø REM ALLEEN HOOFDLETTERS
16Ø PRINT "NA HOEVEEL FOUTEN WIL JE STOPPEN ";
165 INPUT F
17Ø IF F = Ø THEN F = 1
18Ø PRINT: PRINT "TYP HET LAATSTE TEKEN NA"
19Ø PRINT:PRINT "JE KUNT STOPPEN MET SPATIEBALK"
2ØØ PRINT
21Ø FALSE = Ø : OK = Ø
22Ø X = INT (RND (1) * 1ØØ + 1)
23Ø IF X < 33 THEN 22Ø
24Ø IF X > 9Ø THEN 22Ø
25Ø X$=CHR$ (X)
26Ø PRINT "-> "; X$ ;" ";
27Ø A$ = INKEY$
28Ø IF A$ = " " THEN 27Ø
29Ø REM
3ØØ PRINT A$;
31Ø IF A$ = X$ THEN PRINT " IS GOED" : OK =
    OK+1
32Ø IF A$ = CHR$ (32) THEN 37Ø
33Ø IF A$ THEN PRINT " IS FOUT. HET WAS ";X$:
    FALSE=FALSE+1
34Ø PRINT
35Ø IF FALSE>=F THEN 37Ø
36Ø GOTO 22Ø
37Ø PRINT:PRINT "JE SCORE IS "; INT
    ((OK/(OK+FALSE)) * 1ØØ); " PROCENT"
38Ø END
```

KLEURENDEMONSTRATIE

```
 1Ø SCREEN 1
 2Ø PRINT
 3Ø PRINT "KLEUREN-DISKO"
 4Ø FOR X=Ø TO 15
 5Ø FOR Y=1 TO 15
 6Ø FOR Z=2 TO 15
 7Ø COLOR X, Y, Z
 8Ø FOR PAUZE=1 TO 1Ø : NEXT PAUZE
 9Ø NEXT Z
1ØØ NEXT Y
11Ø NEXT X
12Ø COLOR 15, 4, 7
```

RISING SUN

```
 1Ø SCREEN 2
 2Ø COLOR 11,2,9
 3Ø CLS
 4Ø FOR I=1 TO 5Ø
 5Ø CIRCLE (125,1ØØ), I
 6Ø NEXT I
 7Ø FOR P=1 TO 2ØØØ : NEXT P
1ØØ COLOR 15,4,7
11Ø END
```

TOONLADDERS

```
1ØØ REM TOONLADDERS
11Ø FOR O=1 TO 8
12Ø FOR T= 1 TO 7
13Ø READ T$
14Ø A$= "O" + CHR$(48 + O) + T$
15Ø PRINT "TOON ";A$
16Ø PLAY A$
17Ø FOR P=1 TO 4ØØ : NEXT P
18Ø NEXT T
19Ø RESTORE 3ØØ
2ØØ NEXT O
```

```
300 DATA C,D,E,F,G,A,B
400 END
```

OLD MAC DONALD

```
100 REM TONEN OLD MAC DONALD
110 READ A$
120 IF A$="X" THEN GOTO 500
130 B$= "0"; + A$
140 C$= "02" + A$
150 PLAY B$,C$
160 GOTO 110
200 REM DATA OLD MAC
210 DATA G,G,G,G,E,E,D,E,B,B
220 DATA A,A,G,G,D,G,G,G,G,E
230 DATA E,D,D,B,B,A,A,G,G,G
240 DATA X
500 END
```

HAPPY BIRTHDAY

```
100 REM TONEN HAPPY BIRTHDAY
110 READ A$
120 IF A$="X" THEN GOTO 500
130 B$= "02" + A$
140 C$= "02" + A$
150 PLAY B$,C$
160 GOTO 110
200 REM DATA HAPPY BIRTHDAY
210 DATA C,C,D,C,F,E,C,C,D,C
220 DATA F,E,C,C,B,A,F,F,E,D
230 DATA B,B,A,F,G,F
240 DATA X
500 END
```

OLD MAC

```
100 REM TONEN EN DUUR OLD MAC
110 READ A$,D$
120 IF A$="X" THEN GOTO 500
```

```
13Ø B$= "02" + A$ + D$
14Ø C$= "02" + A$ + D$
15Ø PLAY B$,C$
16Ø GOTO 11Ø
2ØØ REM OLD MAC
21Ø DATA G,4,G,4,G,4,G,4,E,4,E,4
22Ø DATA D,2,B,4,B,4,A,4,A,4,G,2
23Ø DATA D,4,G,4,G,4,G,4,G,4,E,4
24Ø DATA E,4,D,2,B,4,B,4,A,4,A,4
25Ø DATA G,1,X,Ø
5ØØ END
```

DE COMPUTER IS VERLIEFD

```
 1Ø REM DE COMPUTER IS VERLIEFD
 2Ø REM BAARDA EN VAN LONDEN
 3Ø REM
 4Ø PRINT "HALLO, HOE HEET JIJ?";
 5Ø INPUT N$
 6Ø PRINT " I ♥ ";N$;
 7Ø GOTO 6Ø
 8Ø REM ******************************
 9Ø REM * STOPPEN MET CTRL EN STOP  *
1ØØ REM ******************************
```

GROOTSTE GEMENE DELER

```
 1Ø REM GROOTSTE GEMENE DELER
 2Ø REM BAARDA EN VAN LONDEN
 3Ø PRINT "GROOTSTE GEMENE DELER"
 4Ø PRINT "EERSTE GETAL " ;: INPUT A
 5Ø PRINT "TWEEDE GETAL " ;: INPUT B
 6Ø REM KIJKEN OF A KLEINER IS DAN B
 7Ø REM ZO JA, DAN A EN B OMKEREN
 8Ø IF A<B THEN H=A : A=B : B=H
 9Ø REM GGD BEREKENING
1ØØ G = INT (A/B) : R = A - (G*B)
11Ø IF R < > Ø THEN A=B : B=R : GOTO 1ØØ
12Ø PRINT:PRINT "DE GGD IS "; B
```

UITROLLENDE DOBBELSTEEN

```
 1Ø REM UITROLLENDE DOBBELSTEEN
 2Ø REM BAARDA EN VAN LONDEN
 3Ø REM
 4Ø X=2 : Y=135
 5Ø REM
 6Ø REM
 7Ø Y=Y+1Ø
 8Ø FOR J=1 TO X : NEXT J
 9Ø X = X*1.2 + 5Ø
1ØØ R=INT (6*RND(1)+1)
11Ø PRINT R;" ";
12Ø REM
13Ø IF X<15ØØ THEN 7Ø
14Ø PRINT
15Ø GOTO 4Ø
16Ø END
```

TWEE DOBBELSTENEN

```
 1Ø REM TWEE DOBBELSTENEN
 2Ø REM BAARDA EN VAN LONDEN
 3Ø PRINT "HOEVEEL KEER WIL JE GOOIEN ";
 4G INPUT N
 5Ø PRINT
 6Ø PRINT "EERSTE TWEEDE TOTAAL"
 7Ø REM LUS VOOR AANTAL WORPEN
 8Ø FOR I = 1 TO N
 9Ø REM DE DOBBELSTEEN GETALLEN
1ØØ R1 = INT (6 * RND (1) + 1)
11Ø R2 = INT (6 * RND (1) + 1)
12Ø SOM = R1 + R2
13Ø PRINT " ";R1;" ";R2;" "; SOM
14Ø NEXT I
15Ø END
```

GETALLEN OMKEREN

```
 1Ø REM GETALLEN OMKEREN
 2Ø REM BAARDA EN VAN LONDEN
 3Ø PRINT "GETALLEN OMKEREN"
 4Ø PRINT
 5Ø PRINT "TYP 2 GETALLEN DOOR EEN KOMMA
    GESCHEIDEN"
 6Ø PRINT "BIJVOORBEELD 21,14"
 7Ø INPUT A,B
 8Ø PRINT "JOUW GETALLEN ZIJN";
 9Ø PRINT A;" EN ";B
1ØØ PRINT
11Ø R=A : PRINT "RESERVE KRIJGT EERSTE GETAL = ";R
12Ø A=B : PRINT "EERSTE KRIJGT NU DE TWEEDE = ";A
13Ø B=R : PRINT "TWEEDE KRIJGT NU DE RESERVE= ";B
14Ø PRINT
15Ø PRINT "DE COMPUTER MAAKT ERVAN ";A;" EN ";B
16Ø END
```

NAMEN OMKEREN

```
 1Ø REM NAMEN OMKEREN
 2Ø REM BAARDA EN VAN LONDEN
 3Ø REM
 4Ø PRINT "NAMEN OMKEREN"
 5Ø PRINT "GEEF 2 NAMEN"
 6Ø INPUT A$
 7Ø INPUT B$
 8Ø PRINT "JOUW NAMEN ZIJN ";
 9Ø PRINT A$; " EN ";B$
1ØØ PRINT
11Ø R$=A$ : PRINT "RESERVE KRIJGT EERSTE NAAM = ";R$
12Ø A$=B$ : PRINT "EERSTE KRIJGT NU DE TWEEDE = ";A$
13Ø B$=R$ : PRINT "TWEEDE KRIJGT NU DE RESERVE = ";B$
14Ø PRINT
15Ø PRINT "DE COMPUTER MAAKT ERVAN ";A$;" EN ";B$
16Ø END
```

GEDACHTENLEZEN

```
 1Ø PRINT "GEDACHTENLEZEN"
 2Ø PRINT "NEEM TWEE GETALLEN ONDER DE TIEN"
 3Ø PRINT "IN GEDACHTEN"
 4Ø PRINT "SCHRIJF ZE OP"
 5Ø PRINT "VERMENIGVULDIG HET EERSTE CIJFER MET 5"
 6Ø GOSUB 3ØØ
 7Ø PRINT "TEL HIER 7 BIJ OP"
 8Ø GOSUB 3ØØ
 9Ø PRINT "VERMENIGVULDIG DIT MET 2"
1ØØ GOSUB 3ØØ
11Ø PRINT "TEL HIER HET ANDERE GETAL DAT JE IN"
12Ø PRINT "GEDACHTEN HAD, BIJ OP"
13Ø PRINT "VERTEL MIJ NU WAT ER OP JE BLAADJE"
14Ø PRINT "STAAT ";
15Ø INPUT X
16Ø PRINT "IK MOET NU EVEN NADENKEN"
17Ø FOR N=1 TO 25Ø
18Ø PRINT ".DENK.";
19Ø NEXT N
2ØØ G1=INT((X-14)/1Ø)
21Ø G2=X-(G1*1Ø)-14
22Ø PRINT
23Ø PRINT "VOLGENS MIJ IS HET "; G1; "EN"; G2
24Ø PRINT "KLOPT HET? JA/NEE"
25Ø INPUT K$
26Ø IF K$="JA" THEN GOTO 28Ø
27Ø IF K$="NEE" THEN PRINT "BEGIN DAN NOG EENS
    OPNIEUW": GOTO 5Ø
28Ø PRINT "KNAP HE"
29Ø END
3ØØ PRINT "KLAAR? JA/NEE ";
31Ø INPUT J$
32Ø IF J$< >"JA" THEN GOTO 3ØØ
33Ø PRINT "SCHRIJF DE UITKOMST OP"
34Ø RETURN
```

PRIEMGETALLEN

```
 1Ø REM PRIEMGETALLEN
 2Ø REM BAARDA EN VAN LONDEN
 3Ø REM
 49 PRINT "TOT WELK GETAL WIL JE"
 5Ø PRINT "DE PRIEMGETALLEN WETEN"
 6Ø INPUT M
 7Ø PRINT 2,
 8Ø FOR I=3 TO M
 9Ø Q=SQR (I)
1ØØ N=1
11Ø N=N+1
12Ø IF INT (I/N)< > I/N THEN 14Ø
13Ø GOTO 16Ø
14Ø IF N<Q THEN 11Ø
15Ø PRINT I,
16Ø NEXT I
17Ø END
```

NONSENS GENERATOR

```
 1Ø REM NONSENS GENERATOR
 2Ø REM BAARDA EN VAN LONDEN
 3Ø REM
 4Ø PRINT "DE COMPUTER IS GEK"
 5Ø PRINT "GEWORDEN. HIJ PRAAT"
 6Ø PRINT "NONSENS"
 7Ø PRINT:PRINT
 8Ø M=INT (5*RND (1) + 1)
 9Ø IF M<2 THEN 8Ø
11Ø FOR J = 1 TO M
12Ø X=INT (26*RND(1)+65)
13Ø W$=CHR$ (X)
14Ø R$=R$+W$
15Ø NEXT J
16Ø PRINT R$;" ";
17Ø R$=" "
2ØØ GOTO 8Ø
```

HUISWERKHULP

```
 1Ø REM BAARDA EN VAN LONDEN
 2Ø PRINT "HUISWERKHULP. JE MOET TELKENS"
 3Ø PRINT "TWEE DINGEN OPGEVEN. BIJV."
 4Ø PRINT "LAND EN HOOFDSTAD OF"
 5Ø PRINT "ENGELS EN NEDERLANDS"
 6Ø REM MAAK PLAATS VOOR 25 VRAGEN
 7Ø DIM A$ (25) : DIM X$ (25) : DIM W$ (15)
 8Ø DIM B$ (25) : DIM Y$ (25)
 9Ø N=Ø : T=Ø : G= Ø
1ØØ PRINT
11Ø INPUT "1 = "; A$
12Ø INPUT "2 = "; B$
13Ø PRINT
14Ø PRINT "JE MOET EERST DE STOF OPGEVEN"
15Ø PRINT "BIJV. NEDERLAND EN AMSTERDAM"
16Ø PRINT "ALS JE KLAAR BENT TYP JE STOP"
17Ø PRINT
18Ø REM HET HUISWERK
19Ø FOR I = 1 TO 25
2ØØ PRINT A$;I: INPUT X$(I)
21Ø IF X$ (I) = "STOP" THEN GOTO 25Ø
22Ø PRINT B$;I: INPUT Y$ (I)
23Ø N=N+1
24Ø NEXT I
25Ø X=Ø
26Ø PRINT:PRINT "IK GEEF JE NU "; A$
27Ø REP=X : X=INT (RND (1) * N +1)
28Ø IF X=REP THEN GOTO 27Ø
29Ø T=T+1
3ØØ PRINT:PRINT X$ (X)
31Ø INPUT " --> "; W$
32Ø IF W$ ="STOP" THEN GOTO 36Ø
33Ø IF W$=Y$(X) THEN PRINT "JA GOED!!" : G=G+1
    : GOTO 27Ø
34Ø PRINT "NEE JAMMER, HET ANTWOORD IS ": PRINT
    Y$(X)
35Ø GOTO 27Ø
```

```
36Ø PRINT
37Ø PRINT "JE HAD ER "; G; " GOED"
38Ø PRINT "VAN DE "; T-1; " POGINGEN"
39Ø PRINT "TOT DE VOLGENDE KEER"
4ØØ END
```

TELEFOONBOEK

```
 1Ø REM TELEFOONBOEK
 2Ø REM BAARDA EN VAN LONDEN
1ØØ PRINT:PRINT "WAT IS DE NAAM"
11Ø INPUT N$
12Ø READ A$, T$
13Ø IF A$="STOP" THEN PRINT "DIE KEN IK NIET" :
    GOTO 5ØØ
14Ø IF N$ < > A$ THEN GOTO 12Ø
15Ø PRINT "TELEFOONNUMMER= "; T$
2ØØ DATA KEES, Ø1Ø-12345
21Ø DATA MARIE, Ø1234-3344
22Ø DATA OPA, Ø7Ø-99999
3ØØ REM
31Ø REM ****************************
32Ø REM * VANAF REGEL 23Ø T/M 48Ø  *
33Ø REM * RUIMTE VOOR MEER NAMEN   *
34Ø REM * EN TELEFOONNUMMERS       *
35Ø REM ****************************
36Ø REM
49Ø DATA STOP, STOP
5ØØ END
```

REKENKUNDIGE TAFELS

```
 1Ø REM REKENKUNDIGE TAFELS
 2Ø REM BAARDA EN VAN LONDEN
 3Ø PRINT "WELKE TAFEL WIL JE"
 4Ø PRINT "DE TAFEL VAN ";: INPUT X
 5Ø PRINT
 6Ø FOR N= 1 TO 1Ø
 7Ø PRINT N;" MAAL ";X;" = "; N * X
```

```
8Ø NEXT N
9Ø END
```

OVERHOREN TAFELS

```
 1Ø REM OVERHOREN TAFELS
 2Ø REM BAARDA EN VAN LONDEN
 3Ø PRINT "MET WELKE TAFEL WIL JE OEFENEN"
 4Ø PRINT "DE TAFEL VAN ";: INPUT X
 5Ø PRINT
 6Ø FOR N=1 TO 1Ø
 7Ø PRINT N;" MAAL ";X;" = ";
 8Ø INPUT A
 9Ø IF A = N * X THEN PRINT "JA GOED!!!"
1ØØ IF A< >N * X THEN PRINT "JAMMER HET IS";N*X
11Ø NEXT N
12Ø END
```

BIJLAGEN

BIJLAGE 1

DE CASSETTERECORDER

BEWAREN

Het is vervelend dat telkens als je de computer afzet, je programma weg is. Je moet dan steeds opnieuw programma's intypen. Dat is ook niet nodig. Je kunt ze namelijk bewaren op een cassettebandje. Als je de volgende keer het programma wilt gebruiken, stop je de cassette in de recorder die bij de computer zit. Je geeft dan de computer de opdracht te lezen wat er op de cassette staat. En in plaats van typen zit het programma in een paar seconden weer in het geheugen van de computer. Als je zelf een programma hebt gemaakt, kun je die ook op een cassette bewaren.

Laten wij een voorbeeldje nemen. Typ het volgende programma maar eens in:

```
1Ø PRINT "IS HET MOOI WEER"
2Ø PRINT "TYP J OF N"
3Ø INPUT W$
4Ø IF W$ = "J" THEN PRINT "FIJN, HOPELIJK
   BLIJFT HET ZO"
5Ø IF W$ = "N" THEN PRINT "JAMMER, HOPELIJK IS
   HET MORGEN BETER"
6Ø END
```

CSAVE

Wij willen dit programma bewaren. Je doet dit met CSAVE. Maar allereerst moet je de REC-knop en de PLAY-knop op de cassetterecorder tegelijk indrukken, je moet wel stevig drukken. Eerst REC, deze vasthouden en dan PLAY indrukken. Als je een speciale BITCORDER van SONY hebt dan alleen op SAVE drukken. Nu moet je `CSAVE "WEER"` typen en op RETURN drukken.
Controleer nog even als de cassette draait of beide knoppen ingedrukt zijn, je kunt dat zo zien, en je hoort het ook, als hij geen gepiep geeft is het niet goed gegaan.

Als de computer klaar is met het programma op cassette zetten, dan geeft hij OK en de cursor (het blokje) staat er weer. Het beeld moet er zó uitzien:

cursor



Probeer het maar eens.

Let er wel op dat de naam tussen aanhalingstekens maximaal uit zes letters mag bestaan. Als je bijvoorbeeld CSAVE "HET WEER" zou willen nemen dan lukt dat niet. De computer maakt er dan zelf zes letters van. HET WEER wordt dan HET WE. Je ziet dat de spatie ook meetelt.

LADEN

Zoals je weet, kun je je eigen programma's op een cassettebandje bewaren (zie het deel over BEWAREN). Maar je kunt ook de programma's van anderen in jouw computer laden via een cassettebandje. Het programma van een vriend bijvoorbeeld. En programma's die je in winkels koopt. Spelletjes als PACMAN en DONKEY KONG zijn niets anders dan hele lange programma's, geschreven in een computertaal, waarin tekenen, schrijven, rekenen en geluid maken met de computer zijn verwerkt. Stel je krijgt een cassette met PACMAN erop en je wilt dat spelletje gaan spelen. Hoe gaat dat dan?

CLOAD

Je doet dat met CLOAD. Je stopt de cassette dan in de recorder. Je moet de cassette wel terugspoelen naar waar PACMAN begint. Je doet dat met de REW-knop die op de cassetterecorder zit. Als je daar op drukt spoelt hij

terug. Je moet nu de computer de opdracht geven dat hij PACMAN moet laden. Maar eerst moet je op de knop PLAY van de cassetterecorder drukken. Als je een BITCORDER van SONY hebt dan op LOAD drukken. Doe dit maar eens. Er gebeurt nu nog niets. Hij start pas als je CLOAD "WEER" typt en op RETURN drukt. Je hoort nu een soort gesuis en dan allemaal piepjes, dat betekent dat hij leest wat er op de cassette staat. Je kunt ook controleren welk programma gelezen wordt. FOUND:WEER betekent eigenlijk GEVONDEN programma WEER. Als hij klaar is verschijnt er OK op het scherm.



cursor

Je kunt nu beginnen door RUN te typen en op RETURN te drukken. Als het goed is dan begint het programma. Klopt het niet, begin dan nog eens te lezen bij CLOAD. Als je alleen CLOAD typt, zonder naam erachter, dan wordt het eerste programma gelezen dat de computer tegenkomt.

BIJLAGE 2
ANTWOORDEN OP DE VRAGEN

Rekenen op de computer

1
```
PRINT 7 + 6                    (druk op RETURN)
13

PRINT 18 + 33                  (druk op RETURN)
51
```
enzovoort

2
```
PRINT 7 + 6 - 3                (druk op RETURN)
1Ø
```

3 Let erop dat je een punt in plaats van een komma typt.
```
PRINT 2.5Ø - Ø.75              (druk op RETURN)
1.75
```

4
```
PRINT 5Ø - 3 * (12.5Ø - 3)
21.5Ø
```

De PRINT-opdracht

1
```
1Ø PRINT "VANDAAG IS HET"
2Ø PRINT "ZONDAG"
3Ø END
```

2
```
1Ø PRINT "DE ZON ";
2Ø PRINT "SCHIJNT"
3Ø END
```
spatie
punt/komma

3
```
1Ø PRINT "DE ZON",
2Ø PRINT "SCHIJNT"
3Ø END
```
komma

4
```
1Ø PRINT "DE ZON"
15 PRINT
2Ø PRINT "SCHIJNT"
3Ø END
```

De INPUT-opdracht

1 Dit programma is niet goed. In regel 10 wordt er een naam gevraagd. In regel 20 wordt via INPUT L een getal gevraagd. Als je een naam vraagt moet je achter L het stringteken $ gebruiken. Als er staat INPUT L$ dan verwacht de computer een naam.

2 Dit programma is niet goed. Via regel 20 vraag je om een naam. Het $ (stringteken) betekent dat de computer alles als een naam zal behandelen. In regel 30 laat je dan de computer rekenen met een naam en dat kan hij niet.

3 Opgave 1 moet worden:

```
1Ø PRINT "WAT IS JE NAAM?"
2Ø INPUT L$
3Ø PRINT "JE NAAM IS "; L$
4Ø END
```

Opgave 2 moet worden:

```
1Ø PRINT "WAT IS JE LEEFTIJD?"
2Ø INPUT L
3Ø MN = L * 12
4Ø PRINT "JE BENT "; MN; " MAANDEN JONG"
5Ø END
```

De GOTO-opdracht

1
```
4Ø IF X = Y THEN GOTO 7Ø
6Ø GOTO 3Ø
```

Als je in regel 60 GOTO 20 getypt hebt, is dat ook goed.

2
```
11Ø IF J$ = "JA" THEN GOTO 14Ø
13Ø GOTO 2Ø
```

3 De computer moet iedere keer N = N + 1 uitvoeren. Je zou daar het volgende programma voor kunnen maken:

```
1Ø N = 1
2Ø PRINT N
3Ø N = N + 1
4Ø GOTO 2Ø
5Ø END
```

Om een programma te maken dat precies hetzelfde doet als in de opgave staat, kun je ook het volgende nemen:

```
1Ø N = 1
2Ø PRINT N
25 IF N = 4 THEN GOTO 5Ø
3Ø N = N + 1
4Ø GOTO 2Ø
5Ø END
```

De GOSUB- en RETURN-opdrachten

1 Er staat geen RETURN achter het hulpprogramma. Als je

```
21Ø RETURN
```

erbij zet is het programma goed.

2 De GOTO-opdrachten op regel 20, 40 en 60 moeten GOSUB-opdrachten zijn. Een RETURN zonder GOSUB werkt niet.

3 Als je vaak hetzelfde moet doen in een programma is het handiger om van dat stukje een hulpprogramma te maken. Je hoeft het dan maar één keer te schrijven. Je kunt er dan iedere keer met GOSUB naar toe springen en met RETURN weer terugkomen.

De REM-opdracht

Het gebruik van REM-opdrachten is heel persoonlijk. De één zal er veel gebruik van maken en de ander maar weinig. Je zou op de volgende manier je programma's kunnen beginnen:

```
1Ø REM DIT PROGRAMMA IS VAN ........
2Ø REM DATUM ........
```

De IF ... THEN-opdracht

1 Verander regel 40 in

```
4Ø IF T = 5 THEN PRINT "SNEL NAAR HUIS"
```

Mooier is:

```
5 Y = 5
4Ø IF T = Y THEN PRINT "SNEL NAAR HUIS"
```

Als je nu nog eens de tijd wilt veranderen, hoef je alleen maar regel 5 te veranderen en die is een stuk korter dan regel 40.

2 = gelijk aan

3 > groter dan

4 < kleiner dan

5 < > ongelijk aan

6 Verander regel 40 in

```
4Ø IF X = 3 THEN GOTO 7Ø
```

Mooier is:

```
5 Y = 3
4Ø IF X = Y THEN GOTO 7Ø
```

De FOR ... NEXT-opdracht

1

```
2Ø FOR J = 1 TO 1Ø
3Ø PRINT J
4Ø NEXT J
5Ø END
```

2

```
2Ø FOR J = 1 TO 1Ø
3Ø PRINT "ARLEEN"
4Ø NEXT J
5Ø END
```

3

```
2Ø FOR J = 1 TO 1Ø
3Ø PRINT J
4Ø NEXT J
5Ø END
```

4 Zie op blz. 113 het programma REKENKUNDIGE TAFELS.

Misschien heb je zelf zoiets gemaakt:

```
1Ø PRINT "WELKE TAFEL"
2Ø INPUT TAFEL
3Ø FOR J = 1 TO 1Ø
4Ø PRINT J * TAFEL
5Ø NEXT J
6Ø END
```

De READ- en DATA-opdrachten

1

```
 1Ø READ N$ , L
 2Ø IF N$ = "STOP" THEN GOTO 5ØØ
 3Ø PRINT N$ , L
 4Ø GOTO 1Ø
1ØØ DATA JAN , 1Ø
11Ø DATA KLAAS , 12
12Ø DATA WIM , 15

49Ø DATA STOP , Ø
5ØØ END
```

2 Zet op regel 130

```
13Ø DATA MARISKA , 14
```

3 Zie het programma TELEFOONBOEK op blz. 113

4 In het programma TELEFOONBOEK kun je van regel 300 tot en met regel 480 zoveel namen en telefoonnummers zetten als je wilt.

5 Dat maakt niet uit.
Meestal zet je eerst de READ-opdracht en daarna de DATA-opdracht.

De RND-opdracht

1 Programma GETALLEN RADEN:

```
2Ø X = INT ( RND (1) * 1Ø + 1)
4Ø X = INT ( RND (1) * 1Ø + 1)
6Ø X = INT ( RND (1) * 1Ø + 1)
```

2
```
X = INT ( RND (1) * 5 + 1)
```

3
```
X = INT ( RND (1) * 25Ø + 1)
```

4 Je moet regel 10 en 20 veranderen.

```
1Ø PRINT "RAAD EEN GETAL VAN 1 T/M 6"
2Ø X = INT ( RND (1) * 6 + 1)
```



Het vierde deel uit deze serie, speciaal voor kinderen geschreven, behandelt het leren omgaan met MSX-computers. De auteurs, drs. D. B. Baarda en drs. A. van Londen, zijn beiden werkzaam aan de subfaculteit der pedagogische en andragogische wetenschappen van de Rijksuniversiteit Utrecht. Het boek leert een jonge computergebruiker, zo veel mogelijk zonder hulp van een volwassene, zijn MSX-computer te gebruiken. Het eerste deel van het boek behandelt op welke wijze een computer kan tekenen, schrijven, kleuren en geluid maken.
In het tweede deel wordt bekeken voor welke toepassingen de computer nog meer is te gebruiken. Bijvoorbeeld het onthouden en bewaren van een opstel, een tekening of een muziekstukje. Ook het uitvoeren van karweitjes zoals moeilijke sommen uitrekenen, strafregels schrijven en spelletjes ontwerpen, komen aan de orde. Kortom in dit deel wordt op speelse en nuttige wijze kennis gemaakt met de programmeertaal MSX-BASIC. Het derde deel ten slotte geeft een aantal voorbeeldprogramma's om zelf in te toetsen en te spelen. Jonge computergebruikers zullen niet alleen door de speelse benadering van de tekst, maar ook door de vormgeving zeker worden aangesproken!

ISBN 90 201 1875 7