

tweemaandelijks tijdschrift September-Oktober 1985

SOON AVAILABLE :SUPERFONT 150 styles !

**THICK3 THICK3 THICK3**

SOON AVAILABLE :SUPERFONT 150 styles !

**SEMIDECO SEMIDECO SEMIDECO**

SOON AVAILABLE :SUPERFONT 150 styles !

**SQUARE SQUARE SQUARE**

SOON AVAILABLE SUPERFONT 150

**1234567 1234567 1234567**

een uitgave van dainamic v.z.w.  
verantw. uitgever w. hermans, mottaart 20 - 3170 herselt

*International*



Herselt, okt 85

Beste Leden,

Dit nummer brengt goed nieuws voor de floppy-gebruikers : **DISKBASED DBASIC** is nu beschikbaar op de verschillende INDATA-systemen.

U vindt de aankondiging op pagina 293. In dit nummer vindt U ook een paar resultaten van plaatjes met de **LOWCOST DIGITIZER**.

Op pagina 271 geeft A.Lambrecht een paar wijzigingen voor het ontwerp om een vlotte afregeling mogelijk te maken. Het project met de joystick interface is in Frankrijk gestrand op productieproblemen. Diegenen die de interface toch nog willen bouwen vinden op p.275 het bouwschema en een voorbeeldprogramma. **KENDOS**-gebruikers vinden in dit nummer waardevolle informatie : het ZAP-programma van A.MARIATTE en een korte routine van A.LAMBRECHT om op een vlotte manier

commando's door te geven.

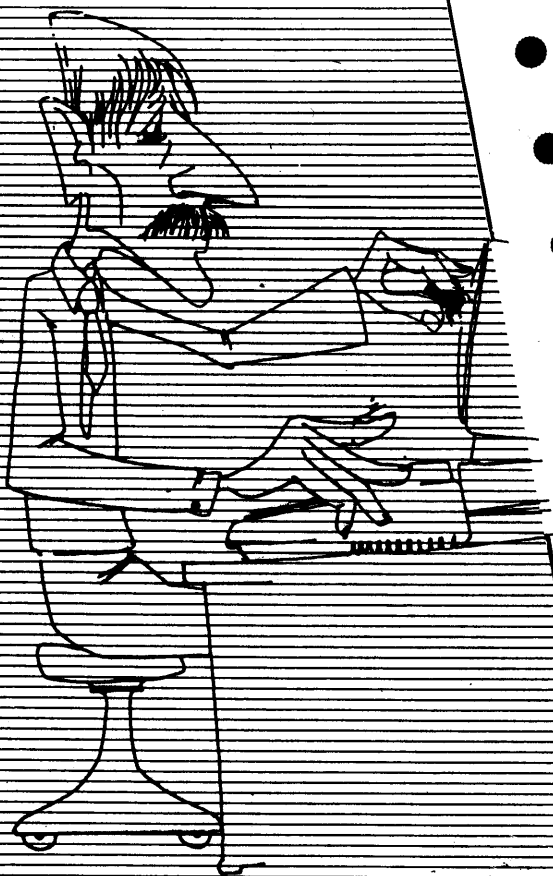
De kans is groot dat vanaf volgend jaar er een samenwerkingsverband komt met de Waalse clubs i.v.m. het tijdschrift. Dit zal betekenen dat er wel meer franse artikelen komen, maar zeker ook nog meer informatie.

Ondertussen is onze voorraad artikelen en programma's sterk aan het afslanken. We willen daarom nog eens een oproep doen aan onze leden om materiaal voor publicatie op te sturen. We ontvangen uw bijdragen liefst op cassette, DCR of schijf. Teksten worden bij voorkeur opgestuurd als FWP-files.

Iedere inzending wordt beloond met een of meer pakketten uit onze softwarebibliotheek. Traditioneel komt dit september-oktober nummer wat later in uw bus vallen. Wij hopen dat U begrip kan opbrengen voor het feit dat de eerste maanden van het schooljaar voor ons minder vrije tijd toelaten.

We wensen U veel leesgenot,  
tot de volgende keer,

W.Hermans



<b>259</b>	<b>VOORWOORD - REMARK</b>	<b>REDACTION</b>
<b>260</b>	<b>INHOUDSTAFEL - CONTENTS</b>	<b>REDACTION</b>
<b>261</b>	<b>PROGRAMMEERTECHNIKEN</b>	<b>F.DRUIJFF</b>
<b>265</b>	<b>SPIEGELSCHRIFT</b>	<b>P.NOIJ</b>
<b>271</b>	<b>CORRECTIONS DIGITIZER</b>	<b>A.LAMBRECHT</b>
<b>272</b>	<b>INDEX-MAKER</b>	<b>I.BROEKMAN</b>
<b>275</b>	<b>JOYSTICK INTERFACE</b>	<b>C.DUFOUR</b>
<b>278</b>	<b>LINEAIRE REGRESSIE</b>	<b>A.DE BRUIJN</b>
<b>280</b>	<b>NEW-YORK</b>	<b>S.DUBOURG</b>
<b>283</b>	<b>ON ERROR GOTO / LOADER</b>	<b>S.DUBOURG</b>
<b>285</b>	<b>LU DANS L.I.S.T.</b>	<b>L.I.S.T./A.MARIATTE</b>
<b>287</b>	<b>MAZE RUN</b>	<b>C.HARDS</b>
<b>289</b>	<b>SCREENDUMP</b>	<b>A.DE BRUIJN</b>
<b>290</b>	<b>DOOLHOF</b>	<b>C.DE BONT</b>
<b>294</b>	<b>BIORYTHM</b>	
<b>298</b>	<b>TEXT-SCRAMBLER</b>	<b>C.DE BONT</b>
<b>301</b>	<b>KENDOS-COMMANDS</b>	<b>A.LAMBRECHT</b>
<b>302</b>	<b>LOW COST DIGITIZER</b>	<b>A.LAMBRECHT</b>
<b>304</b>	<b>POWER-ON INITIALISATION</b>	<b>TRANSLATION</b>
<b>307</b>	<b>CHR\$(12) IN HARDWARE</b>	<b>TRANSLATION</b>
<b>308</b>	<b>PROGRAMMING TECHNIQUES</b>	<b>TRANSLATION</b>
<b>314</b>	<b>JEROEN DEMO 9</b>	<b>JEROEN</b>

Niets uit deze uitgave mag worden veelevoudigd 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.

We vervolgen de reeks artikelen over de functies in de DAI met de goniometrische functies.

Voor diegenen die al vertrouwd zijn met de goniometrie misschien een afgezaagd verhaal, maar voor de anderen toch prettig dat er eens aandacht besteed wordt aan iets wat anders alleen na een inspannende zelfstudie is te bevatten.

Om tot een goed begrip te komen is het wellicht het beste te beginnen met de goniometrische verhoudingen. We bekijken daartoe een rechthoekige driehoek met hoekpunten A, B en C.

## complicatie

Voor onze belgische lezers treedt nu al een complicatie op.

In Nederland is het gebruikelijk in de meetkunde om punten met hoofdletters aan te geven en lijnen met kleine letters. Lijnstukken geven we aan met begin- en eindpunt en tot slot hoeken op een van de drie volgende manieren :

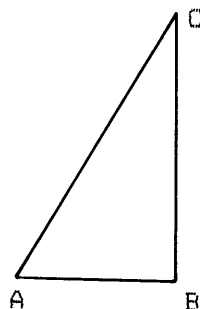
- griekse kleine letter
- hoekteken gevolgd door het hoekpunt
- hoekteken gevolgd door drie punten, het middelste punt is dan het hoekpunt van de bedoelde hoek en in combinatie met de andere twee punten levert het hoekpunt de twee lijnstukken die de benen van de hoek vormen.

In België echter geeft men juist punten met kleine letters aan en de lijnen met hoofdletters. Ondanks dat ik wiskundig de belgische methode prefereer; hij is namelijk consistent met de verzamelingstheorie ga ik toch verder met de nederlandse notatie. Ik ben namelijk bang ergens een vergissing te maken en dan is het voor beide groepen onduidelijk.

Het hoekteken zal ik met / aangeven en er dan later nog een lijntje onder zetten zodat het  $\sphericalangle$  wordt. Mocht ik dit per ongelijk ergens vergeten kunt u na deze uitleg weten dat het hoek betekent.

## goniometrische verhoudingen

We bekijken de onderstaande rechthoekige driehoek ABC. Zoals U ziet zit de rechte hoek bij B.



Deze driehoek bestaat uit twee rechthoekszijden (AB en BC) en een schuine zijde (AC); de laatste wordt ook wel hypotenusa genoemd.

In deze driehoek zijn nu de volgende zes verhoudingen aan te geven:

$$\frac{AB}{AC} \quad \frac{AB}{BC} \quad \frac{AC}{BC} \quad \frac{AC}{AB} \quad \frac{BC}{AC} \quad \text{en} \quad \frac{BC}{AB}$$

Vrij eenvoudig is in te zien dat de groottes van deze verhoudingen niet afhangen van de lengtes zelf maar van de vorm van de driehoek. Deze wordt bepaald door de grootte van de hoeken van de driehoek. Vandaar goniometrie dat uit het grieks komt en hoek betekent.

Al deze zes verhoudingen hebben in de wiskunde een naam gekregen maar die wordt meestal opgegeven in algemene termen. We spreken over overstaande en aanliggende rechthoekszijde van een van de scherpe hoeken en van de schuine zijde .

Met rechthoekszijde wordt de zijde bij de rechte hoek bedoeld, in onze tekening is dat dus zijde AB of BC. Met aanliggend geven we aan dat het hoekpunt van de hoek waaraan we de verhouding koppellen op bedoelde zijde ligt.

Ten opzichte van  $\sphericalangle A$  is dat zijde AB. Bij overstaand ligt dat hoekpunt er juist niet op.

Ten opzichte van  $\sphericalangle A$  is dat zijde BC.

## definities

We krijgen zo de volgende definities.

$$\sinus = \frac{\text{overstaande rechthoekszijde}}{\text{schuine zijde}}$$

$$\cosinus = \frac{\text{aanliggende rechthoekszijde}}{\text{schuine zijde}}$$

$$\text{tangens} = \frac{\text{overstaande rechthoekszijde}}{\text{aanliggende rechthoekszijde}}$$

$$\text{cotangens} = \frac{\text{aanliggende rechthoekszijde}}{\text{overstaande rechthoekszijde}}$$

$$\text{secans} = \frac{\text{schuine zijde}}{\text{aanliggende rechthoekszijde}}$$

$$\text{cosecans} = \frac{\text{schuine zijde}}{\text{overstaande rechthoekszijde}}$$

Voor  $\angle A$  uit onze driehoek wordt dit :

$$\sin(\angle A) = \frac{BC}{AC} \quad \text{COSEC}(\angle A) = \frac{AC}{BC}$$

$$\cos(\angle A) = \frac{AB}{AC} \quad \text{SEC}(\angle A) = \frac{AC}{AB}$$

$$\text{TAN}(\angle A) = \frac{BC}{AB} \quad \text{COTG}(\angle A) = \frac{AB}{BC}$$

We gebruiken normaal alleen de sinus, de cosinus en de tangens omdat we de andere drie gemakkelijk kunnen vinden door de reciproke van de geschikte verhouding te nemen:

$$\text{b.v.} \quad \text{COSEC}(\angle A) = \frac{1}{\sin(\angle A)}$$

Analoog voor cotangens met tangens en secans met cosinus.

Ook in onze DAI zit alleen de SIN, de COS en de TAN maar mocht u ooit eens

de behoefte gevoelen tot het gebruik van een der anderen kent u nu het recept ervoor.

De hypotenusa (schuine zijde) is in de rechthoekige driehoek vanzelfsprekend de langste van de drie zijden.

Tegenover de grootste hoek ligt altijd de langste zijde. Het is nu eenvoudig in te zien dat de SIN en de COS waarden zullen opleveren tussen 0 en 1. Met een beetje fantasie kunnen we ons ook nog voorstellen dat de grenswaarden 0 en 1 zelf gehaald kunnen worden.

Bij  $\sin(0)$  stellen we ons dan een driehoek ABC voor met  $\angle B$  recht en  $\angle A=0$ , dit betekent dat zijde AB en zijde AC boven op elkaar liggen en zijde BC lengte nul heeft. Voor de rechte hoek A een analoge fantasie.

## doordenken

We willen echter graag de goniometrische verhoudingen ook voor hoeken die stomp (meer dan recht) zijn kunnen definiëren.

We willen dit graag omdat er vele regels blijken te bestaan die niet alleen opgaan in rechthoekige driehoeken maar ook in andere.

Een voorbeeld van zo'n regel is de zogenaamde sinusregel:

## sinusregel

In elke driehoek geldt :

$$\frac{\sin(\angle A)}{BC} = \frac{\sin(\angle B)}{AC} = \frac{\sin(\angle C)}{AB}$$

Soms zien we deze regel op zijn kop staan dat wil zeggen de zijden boven en de sinussen (sini?) onder de breukstreep. Dat is dan net zo juist als de hierbovenstaande regel.

In stomphoekige driehoeken kunnen we met deze sinusregel de sinussen bepalen van stompe hoeken.

Maar we willen nog verder gaan en aan elke hoek een sinuswaarde koppelen en vanzelfsprekend een die niet strijdig is met de reeds gemaakte afspraken. We gaan dit doen met behulp van de eenheidscirkel.

## hoekmaat

In het begin van de goniometriestudie worden hoeken bijna altijd gemeten in graden. De kleinste hoek is 0 graden en een rechte hoek is 90 graden, een zogenaamde gestrekte hoek (benen in elkaars verlengde) is 180 graden en een keer helemaal rond is 360 graden. Wiskundig is er erg veel voor deze maatverdeling te zeggen. Het getal 360 kan namelijk door vreselijk veel andere getallen gedeeld worden. Delen door 1,2,3,4,5,6 levert geen breuken op en ook daarboven zijn nog zeer vele delers te vinden. Het betekent dat er voor de 'normale' driehoeken veelal hoeken gevonden worden waarbij we geen breuk hoeven te krijgen. Onze voorouders hadden het nog niet zo slecht bekeken met een dozijn eieren, zestig minuten in een uur en 24 uren per dag.

## radialen

Er zijn echter om andere wiskundige redenen ook argumenten om niet een verdeling in graden te maken. Er zijn twee andere verdelingen :

- 1-verdeling in GRAD's waarvan er 100 in een rechte hoek zitten. Deze verdeling wordt vooral gebruikt bij bepaalde technische toepassingen.
- 2-verdeling in radialen.

Bij de DAI moeten hoeken worden opgegeven in radialen. Maar wat zijn radialen nu precies ? Ik kan me er gemakkelijk vanaf maken door te stellen dat 1 radiaal gelijk is aan ongeveer 57.2958 graden. Er zullen er onder de lezers zijn die nu denken 'wat is het een gek getal!' of 'waarom ongeveer ?'. Laten we voorop stellen dat beide maten onafhankelijk zijn in de zin dat niet de een van de ander is afgeleid. Vandaar dat de ondelinge verhouding nogal vreemd is. We zouden ook kunnen zeggen dat een graad gelijk is aan ongeveer 0.017453 radiaal. We zagen al hoe we aan de graden kwamen. (eenmaal rond is 360) De 360 is arbitraal zij het gelukkig gekozen. Een radiaal vinden we door een zodanig segment uit een cirkel te nemen dat de boog precies evenlang is als de straal van de cirkel. De hoek van deze taartpunt is dan 1 radiaal.

Als u bedenkt dat voor straal ook wel het woord radius wordt gebruikt kunt u de term radiaal plaatsen. Wat is nu het grote voordeel van radialen boven graden. Heel simpel de sinus e.d. van een hoek uitgedrukt in radialen kan worden berekend.

Een bekende reeksontwikkeling is b.v.

$$\text{SIN}(X)=X - \frac{X^3}{3!} + \frac{X^5}{5!} - \frac{X^7}{7!} + \frac{X^9}{9!} - \dots$$

waarmee de sinus van X is te bepalen.

N.B.  $5! = 1 \times 2 \times 3 \times 4 \times 5$

De bovenstaande formule is alleen geldig als X wordt uitgedrukt in de reeds eerder genoemde radialen.

Aangezien de termen in deze reeks na de eerste zoveel steeds kleiner worden kan men stellen dat als alleen de eerste vijf termen uit de reeks berekend worden men al een vrij nauwkeurige waarde voor de sinus heeft. Dit laatste geldt destemee als X een kleine waarde heeft.

Maar hoeveel radialen is een slag rond nu ? Dat is natuurlijk (denk aan de definitie van radiaal) net zoveel als de omtrek van een cirkel meer is dan de straal van die cirkel. Dit is 6,28... ofwel 2 PI ( $\pi$ ). Voor de komende definitie van sinus e.d. maken we hiervan gebruik.

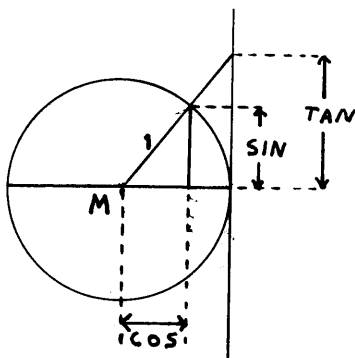
## uitbreiding definitie

Bij de oorspronkelijke definitie van sinus e.d. maakten we gebruik van een rechthoekige driehoek van willekeurige grootte. Als we echter de schuine zijde precies 1 lang maken is  $\text{SIN}(\sphericalangle A)$  gelijk aan de lengte van BC. We zouden op deze manier een redelijk nauwkeurige tabel voor de sinus kunnen maken als we AC tamelijk groot nemen b.v. 1 meter. De lengte van BC is dan zeker op millimeters nauwkeurig af te lezen en de sinus van de betrokken hoek dus op drie cijfers achter de komma bekend. We gaan die gedachte eens verder uitwerken. De draaiende beweging van AC om punt A waarbij BC de grootte van

de bij die hoek behorende sinus aangeeft is intrigerend. Lengte van EC is de afstand van C tot de horizontale lijn door A (of evt. AB). Deze definitie is voor scherpe hoeken dezelfde als voorheen. Ook de hoek  $\theta$  en de rechte hoek kloppen met ons intuitief idee hierover. Maar nu zijn ook de sinussen van hoeken boven de 90 graden bekend. Sterker er is geen enkele limiet meer aan de definitie. Ook de sinus van 770 graden vormt geen probleem meer als we bij  $\theta$  starten en tweemaal ronddraaien ( $2 \times 360$ ) en dan nog 50 graden. In deze vorm is het duidelijk te zien dat de sinus 50 graden gelijk is aan de sinus van 770 graden.

### eenheidscirkel

In de onderstaande cirkel (straal=1) kunnen we duidelijk de drie hoofdverhoudingen uit de goniometrie zien.



In het kort gezegd komt het er op neer dat punt C langs de cirkelrand ronddraait en de afstand van C tot de horizontale lijn de sinus aangeeft. De cosinus is dan de afstand tot de verticale lijn en de tangens zien we op verticale raaklijn aan de cirkel.

Gaan we de gevonden sinuswaarden op een grafiek afzetten krijgen we een golvende curve te zien. (sinus betekent ook golf) Ook de cosinus is een soortgelijke golf; in feite is het dezelfde golf die alleen op een ander punt start. Dit laatste is in de cirkel eenvoudig te zien maar ook in de oorspronkelijk driehoek zien we dat  $\text{SIN}(\theta) = \text{COS}(90-\theta)$  en omgekeerd. De 'co' van cosinus is een afkorting van complements; en twee hoeken die samen recht zijn zijn elkaars comple-

ment, vandaar !

De DAI gebruikt bij de berekening van de sinus ook zo'n reeksontwikkeling. U begrijpt dat de uitkomsten onbetrouwbaarder worden naarmate het argument groter wordt.

$\text{SIN}(200000 \times \text{PI})$  zou  $\theta$  moeten zijn maar levert bijna  $-0,1$ .

Er is hier echter nog een probleem. De waarde van PI die standaard in de DAI zit is een afgeronde waarde wordt die dan met een benadering (omzetting naar binair) van 200000 vermenigvuldigd kan de gecombineerde fout vrij groot worden. Heel extreem is dan ook  $\text{SIN}(2^20 \times \text{PI})$  dat  $\theta$  zou moeten zijn. Tik in en verbaas u en bedenk daarbij dat  $2^20$  volkomen exact is in de DAI.

Ik meende ooit een oplossing voor dit probleem te hebben in de wiskundig correcte verkleining van het argument door hem  $\text{MOD } 2 \times \text{PI}$  te nemen. De reksnelheid ging met deze methode ook enorm omhoog. Later realiseerde ik me echter dat de MOD een integerfunctie is en dat mijn argumenten modulo zes genomen werden in plaats van de gewenste modulo twee PI.

Ik dacht hieraan eventueel nog te ontkomen op de volgende manier:

- 1 - deel argument door PI
- 2 - neem het vervolgens MOD 2
- 3 - vermenigvuldig met PI

Dit gaat ook fout door de integer MOD vervang 2 door verminder met het grootste even getal onder de waarde uit 1. ( $A = A - \text{INT}(A/2.0) \times 2$ )

Dit werkt wel maar is te traag om zoden aan de dijk te zetten.

De totale verkleining ineens werkt vlotter maar nog niet vlot genoeg.  $B = 2 \times \text{PI}; \text{SIN}(A - \text{INT}(A/B) \times B)$  ipv  $\text{SIN}(A)$

Maar al zou het vlot genoeg werken bedenk wel dat de uitkomsten op zijn minst onnauwkeurig zijn.

Gebruik de goniometrische functies dus nooit met grote waarden. Wat voor u groot is hangt af van de gewenste betrouwbaarheid, maar ik zou zeggen neem als regel nooit een argument groter dan duizend.

Met deze laatste tip stop ik dit artikel maar ik ga er de volgende keer met dit onderwerp verder.

Frank H. Druijff



# Spiegelschrift op FX-80

## SPIEGELLEN OP EPSON

Omdat algemeen verondersteld wordt dat ik in spiegelschrift schrijf, besloot ik ook aldus te gaan printen. Darto ontwikkelde ik in wat lege uren het onderstaand programma, waarmee het goed spiegelen is. Ik hoop hiermee ook anderen een plezier te doen.

Het spreekt voor zich dat deze tekst met dit programma gedownload is.

De relevante regels om in spiegelschrift te kunnen printen, zijn:

11000 - 12000	:	Italic-alfabet in spiegelschrift.
10000 - 11000	:	Normal (?) alfabet in spiegelschrift;
4000 - 5000	:	Inlezen string en printen in spiegelschrift;
2500 - 3000	:	Inbrengen van de RAM-karakterzet;
2140	:	Kuize voor de (nog in te brengen) RAM-karakterzet;

Enkele bijzonderheden:

- Indien u in spiegelschrift wenst te schrijven hoeft u enkel een <RETURN> in te brengen (defaultwaarde is dus 13);
- Indien u overschakelt van of naar super/subscript, enlarged ofwel condensed, vraagt het programma automatisch naar het aantal posities per regel;
- Defaultwaarde bij de vraag naar het aantal posities is 80;
- Het is raadzaam het aantal posities ingevul van enlarged printen op de helft van het aantal posities ingevul van een normale print in te stellen;
- Deze tekst is geschreven op 42 (enlarged) cp. 9 (normal);
- Het is mogelijk de printmode te verlaten door enkel "stop" in te brengen;
- Alvorens de printmode te verlaten vraagt het programma of een form feed gewenst is: default is geen form feed (enkel <RETURN>).

Het is mijn ervaring dat het eenvoudigere printen dan schrijven is !!!

Peter Noij

```

1000 REM -----
1010 REM subroutine      : -
1020 REM .              'spiegelen op epson'(FX-80!)
1030 REM 12 maart 1985 : versie F.1
1040 REM pnoij
1090 REM -----
2000 POKE #131,1:PRINT CHR$(12):COLORT 9 14 9 9
2010 CURSOR 25,18:PRINT "pnoij"
2020 CURSOR 12,14:PRINT "'spiegelschrift op mijn FX-80 !'"
2030 CURSOR 11,12:PRINT "staat de printer klaar ? space";
2100 IF GETC<>32 GOTO 2100
2110 PRINT CHR$(12):CLEAR 2000:DIM ST$(10):RESTORE
2120 POKE #131,0:CURSOR 15,15:PRINT CHR$(27);"%";CHR$(1);CHR$(0)
2130 CURSOR 15,15:PRINT CHR$(27);":";CHR$(0);CHR$(0);CHR$(0);:POKE #131,1
2500 READ X$:IF X$="stop" GOTO 3000
2510 CURSOR 15,15:PRINT X$;:POKE #131,0:PRINT CHR$(27);"&";CHR$(0);
2520 READ X:PRINT CHR$(X);CHR$(X);:FOR J=1 TO 13:READ X:PRINT CHR$(X);:NEXT
2530 POKE #131,1:GOTO 2500
    
```

```

2990 REM - - - - -
3000 SP=80
3100 PRINT CHR$(12)
3110 CURSOR 10,22:PRINT " 0. standaarschrift : ";ST$(0)
3120 CURSOR 10,21:PRINT " 1. italic : ";ST$(1)
3130 CURSOR 10,20:PRINT " 2. emphasized : ";ST$(2)
3140 CURSOR 10,19:PRINT " 3. double strike : ";ST$(3)
3150 CURSOR 10,18:PRINT " 4. elite : ";ST$(4)
3160 CURSOR 10,17:PRINT " 5. superscript : ";ST$(5)
3170 CURSOR 10,16:PRINT " 6. subscript : ";ST$(6)
3180 CURSOR 10,15:PRINT " 7. enlarged : ";ST$(7)
3190 CURSOR 10,14:PRINT " 8. proportional : ";ST$(8)
3200 CURSOR 10,13:PRINT " 9. underlined : ";ST$(9)
3210 CURSOR 10,12:PRINT "10. condensed : ";ST$(10)
3220 CURSOR 10,10:PRINT "11. posities per regel : ";SP
3230 CURSOR 10,8:PRINT "12. defaultinstelling"
3240 CURSOR 10,6:PRINT "13. printen"
3250 CURSOR 10,4:PRINT "14. einde sessie"
3300 CURSOR 10,2:PRINT SPC(40);:CURSOR 10,2:PRINT "J E K E U Z E";
3310 A$="":GOSUB 3800:IF LEN(A$)=0 THEN A$="13"
3320 X=VAL(A$):IF X<0 OR X>14 GOTO 3300
3330 J=0:FOR I=0 TO 10:IF X=I AND LEN(ST$(I))>0 THEN ST$(I)="":J=1
3340 NEXT:IF X=5 OR X=6 OR X=7 OR X=10 THEN IF J=1 THEN J=0:X=11
3350 IF J=1 GOTO 3100
3400 IF X=0 THEN ST$(0)=CHR$(27)+"P"
3410 IF X=1 THEN ST$(1)=CHR$(27)+"4"
3420 IF X=2 THEN ST$(2)=CHR$(27)+"E"
3430 IF X=3 THEN ST$(3)=CHR$(27)+"6"
3440 IF X=4 THEN ST$(4)=CHR$(27)+"M"
3450 IF X=5 THEN ST$(5)=CHR$(27)+"S0":X=11
3460 IF X=6 THEN ST$(6)=CHR$(27)+"S1":X=11
3470 IF X=7 THEN ST$(7)=CHR$(27)+"W1":X=11
3480 IF X=8 THEN ST$(8)=CHR$(27)+"p1"
3490 IF X=9 THEN ST$(9)=CHR$(27)+"-1"
3500 IF X=10 THEN ST$(10)=CHR$(27)+CHR$(15):X=11
3510 IF X=11 THEN CURSOR 33,2:A$="":GOSUB 3800:IF LEN(A$)=0 THEN A$="80"
3520 IF X=11 THEN SP=VAL(A$):IF SP<1 GOTO 3510
3530 IF X=12 THEN GOSUB 3600
3540 IF X=13 THEN GOSUB 3700:GOSUB 4000:GOTO 3100
3550 IF X=14 THEN PRINT CHR$(12);:STOP
3580 GOTO 3100

3600 POKE #131,0:GOSUB 3700:POKE #131,1:FOR I=0 TO 10:ST$(I)="":NEXT:SP=80
3650 RETURN
3700 POKE #131,0
3710 CURSOR 0,23:PRINT CHR$(27);"5";CHR$(27);"F";
3720 CURSOR 0,23:PRINT CHR$(27);"H";CHR$(27);"P";
3730 CURSOR 0,23:PRINT CHR$(27);"T";CHR$(27);"W0";
3740 CURSOR 0,23:PRINT CHR$(27);"p0";
3750 CURSOR 0,23:PRINT CHR$(27);"-0";CHR$(18);
3760 POKE #131,1
3770 RETURN
3800 A$="":PRINT " : ";
3850 M=GETC:IF M=13 THEN RETURN
3860 IF M=8 AND LEN(A$)=0 GOTO 3850
3870 IF M=8 THEN A$=LEFT$(A$,LEN(A$)-1):GOTO 3900
3880 IF LEN(A$)>3 OR M<48 OR M>57 GOTO 3850
3890 A$=A$+CHR$(M)
3900 PRINT CHR$(M);
3910 GOTO 3850

```



10180	DATA	** 2 **	50,139,	0,	98,128,	18,128,	10,128,	6,128,	66,	0,	0				
10190	DATA	** 3 **	51,139,	0,140,	0,210,	0,162,	0,130,	0,132,	0,	0	0				
10200	DATA	** 4 **	52,139,	0,	8,	0,254,	0,136,	64,	40,	16,	8,	0,	0		
10210	DATA	** 5 **	53,139,	0,156,	2,160,	2,160,	2,160,	2,228,	0,	0	0	0			
10220	DATA	** 6 **	54,139,	0,	12,	18,	0,	18,128,	82,	32,	18,	12,	0,	0	
10230	DATA	** 7 **	55,139,	0,128,	64,160,	16,136,	4,130,	0,128,	0,	0	0	0	0		
10240	DATA	** 8 **	56,139,	0,108,146,	0,146,	0,146,	0,146,	108,	0,	0	0	0	0		
10250	DATA	** 9 **	57,139,	0,	96,144,	8,148,	2,144,	0,144,	96,	0,	0	0	0		
10260	DATA	** : **	58,139,	0,	0,	0,	0,	54,	0,	54,	0,	0,	0,	0	
10270	DATA	** ; **	59, 11,	0,	0,	0,	0,	110,	0,109,	0,	0,	0,	0,	0	
10280	DATA	** > **	60,139,	0,	0,	0,130,	0,	68,	0,	40,	0,	16,	0,	0	
10290	DATA	** = **	61,139,	0,	40,	0,	40,	0,	40,	0,	40,	0,	40,	0,	0
10300	DATA	** > **	62,139,	0,	0,	16,	0,	40,	0,	68,	0,130,	0,	0,	0	
10310	DATA	** ? **	63,139,	0,	96,128,	16,128,	10,128,	0,128,	64,	0,	0	0	0	0	
10320	DATA	** **	64,139,	0,122,	0,170,	0,170,	16,130,	68,	56,	0,	0	0	0	0	
10330	DATA	** A **	65,139,	0,	30,	32,	72,128,	8,128,	72,	32,	30,	0,	0	0	
10340	DATA	** B **	66,139,	0,108,	16,130,	16,130,	16,130,	124,	130,	0,	0	0	0	0	
10350	DATA	** C **	67,139,	0,	68,130,	0,130,	0,130,	0,130,	124,	0,	0	0	0	0	
10360	DATA	** D **	68,139,	0,	56,	68,130,	0,130,	0,130,	124,	130,	0,	0	0	0	
10370	DATA	** E **	69,139,	0,130,	0,146,	0,146,	0,146,	0,254,	0,	0	0	0	0	0	
10380	DATA	** F **	70,139,	0,128,	0,144,	0,144,	0,144,	0,254,	0,	0	0	0	0	0	
10390	DATA	** G **	71,139,	0,	92,130,	16,130,	16,130,	0,130,	124,	0,	0	0	0	0	
10400	DATA	** H **	72,139,	0,254,	0,	16,	0,	16,	0,	16,	0,254,	0,	0	0	
10410	DATA	** I **	73,139,	0,	0,	0,130,	0,254,	0,130,	0,	0,	0,	0	0	0	
10420	DATA	** J **	74,139,	0,	0,	128,124,130,	0,130,	0,	2,	12,	0,	0	0	0	
10430	DATA	** K **	75,139,	0,130,	0,	68,	0,	40,	0,	16,	0,254,	0,	0	0	
10440	DATA	** L **	76,139,	0,	2,	0,	2,	0,	2,	0,	2,	0,254,	0,	0	
10450	DATA	** M **	77,139,	0,254,	0,	64,	32,	16,	32,	64,	0,254,	0,	0	0	
10460	DATA	** N **	78,139,	0,254,	0,	4,	8,	16,	32,	64,	0,254,	0,	0	0	
10470	DATA	** O **	79,139,	0,124,130,	0,130,	0,130,	0,130,	0,130,	124,	0,	0	0	0	0	
10480	DATA	** P **	80,139,	0,	96,	0,144,	0,144,	0,144,	0,254,	0,	0	0	0	0	
10490	DATA	** Q **	81,139,	0,122,128,	4,130,	8,130,	0,130,	124,	0,	0	0	0	0	0	
10500	DATA	** R **	82,139,	0,	98,	4,152,	0,144,	0,144,	0,254,	0,	0	0	0	0	
10510	DATA	** S **	83,139,	0,	76,146,	0,146,	0,146,	0,146,	100,	0,	0	0	0	0	
10520	DATA	** T **	84,139,	0,128,	0,128,	0,254,	0,128,	0,128,	0,	0	0	0	0	0	
10530	DATA	** U **	85,139,	0,252,	2,	0,	2,	0,	2,	0,	2,252,	0,	0	0	
10540	DATA	** V **	86,139,	0,224,	16,	8,	4,	2,	4,	8,	16,224,	0,	0	0	
10550	DATA	** W **	87,139,	0,252,	2,	4,	8,	48,	8,	4,	2,252,	0,	0	0	
10560	DATA	** X **	88,139,	0,	0,130,	68,	40,	16,	40,	68,130,	0,	0	0	0	
10570	DATA	** Y **	89,139,	0,128,	64,	32,	16,	14,	16,	32,	64,128,	0,	0	0	
10580	DATA	** Z **	90,139,	0,	0,130,	64,162,	16,138,	4,130,	0,	0	0	0	0	0	
10590	DATA	** [ **	91,139,	0,	0,	0,130,	0,130,	0,254,	0,	0,	0,	0	0	0	
10600	DATA	** **	92,139,	0,	0,	2,	4,	8,	16,	32,	64,128,	0,	0	0	
10610	DATA	** ] **	93,139,	0,	0,	0,254,	0,130,	0,130,	0,	0,	0,	0	0	0	
10620	DATA	** ^ **	94,139,	0,	32,	0,	64,	0,128,	0,	64,	0,	32,	0,	0	
10630	DATA	** **	95, 11,	0,	1,	1,	1,	1,	1,	1,	1,	1,	1,	0,	0
10640	DATA	** **	96,139,	0,	0,	0,	0,	0,	64,128,	0,	0,	0,	0	0	
10650	DATA	** a **	97,139,	0,	2,	28,	32,	10,	32,	10,	32,	10,	4,	0,	0
10660	DATA	** b **	98,139,	0,	0,	28,	34,	0,	34,	0,	34,	0,254,	0,	0	
10670	DATA	** c **	99,139,	0,	0,	34,	0,	34,	0,	34,	0,	34,	28,	0,	0
10680	DATA	** d **	100,139,	0,	0,254,	0,	34,	0,	34,	0,	34,	28,	0,	0	
10690	DATA	** e **	101,139,	0,	24,	34,	8,	34,	8,	34,	8,	34,	28,	0,	0
10700	DATA	** f **	102,139,	0,	0,	0,144,	0,144,	126,	16,	0,	16,	0,	0	0	
10710	DATA	** g **	103, 11,	0,	0,126,	1,	68,	1,	68,	1,	68,	56,	0,	0	
10720	DATA	** h **	104,139,	0,	0,	30,	32,	0,	32,	0,	32,	0,254,	0,	0	
10730	DATA	** i **	105,139,	0,	0,	0,	0,	2,	0,190,	0,	34,	0,	0	0	
10740	DATA	** j **	106,139,	0,	0,	0,190,	1,	32,	1,	0,	1,	0,	0	0	
10750	DATA	** k **	107,139,	0,	0,	34,	0,	20,	0,	8,	0,254,	0,	0	0	
10760	DATA	** l **	108,139,	0,	0,	0,	0,	2,	0,254,	0,130,	0,	0	0	0	
10770	DATA	** m **	109,139,	0,	30,	32,	0,	32,	30,	32,	0,	32,	30,	0,	0

10780	DATA	*** n **	,110,139,	0,	0,	30,	32,	0,	32,	0,	32,	0,	62,	0,	0
10790	DATA	*** o **	,111,139,	0,	28,	34,	0,	34,	0,	34,	0,	34,	28,	0,	0
10800	DATA	*** p **	,112, 11,	0,	0,	56,	68,	0,	68,	0,	68,	0,	127,	0,	0
10810	DATA	*** q **	,113, 11,	0,	0,	127,	0,	68,	0,	68,	0,	68,	56,	0,	0
10820	DATA	*** r **	,114,139,	0,	0,	32,	0,	32,	0,	32,	16,	0,	62,	0,	0
10830	DATA	*** s **	,115,139,	0,	4,	42,	0,	42,	0,	42,	0,	42,	16,	0,	0
10840	DATA	*** t **	,116,139,	0,	0,	2,	32,	2,	32,	2,	252,	0,	32,	0,	0
10850	DATA	*** u **	,117,139,	0,	2,	60,	0,	2,	0,	2,	0,	2,	60,	0,	0
10860	DATA	*** v **	,118,139,	0,	32,	16,	8,	4,	2,	4,	8,	16,	32,	0,	0
10870	DATA	*** w **	,119,139,	0,	60,	2,	4,	8,	16,	8,	4,	2,	60,	0,	0
10880	DATA	*** x **	,120,139,	0,	0,	0,	34,	20,	0,	8,	0,	20,	34,	0,	0
10890	DATA	*** y **	,121, 11,	0,	64,	32,	16,	8,	4,	10,	17,	32,	64,	0,	0
10900	DATA	*** z **	,122,139,	0,	0,	0,	34,	16,	34,	8,	34,	4,	34,	0,	0
10910	DATA	*** **	,123,139,	0,	0,	130,	0,	130,	112,	0,	16,	0,	0,	0,	0
10920	DATA	*** **	,124,139,	0,	0,	0,	0,	0,	238,	0,	0,	0,	0,	0,	0
10930	DATA	*** **	,125,139,	0,	0,	0,	16,	0,	112,130,	0,	130,	0,	0,	0,	0
10940	DATA	*** **	,126,139,	0,	64,	32,	0,	32,	64,128,	0,	128,	64,	0,	0,	0
10950	DATA	*** **	,127,139,	0,	124,130,	64,162,	16,138,	4,130,	124,	0,	0,	0,	0,	0,	0
11000	REM														
11010	DATA	*** ! **	,161,139,	0,	0,	128,	64,	32,	16,	8,	0,	2,	0,	0,	0
11020	DATA	*** **	,162,139,	0,	0,	128,	64,	32,	0,	128,	64,	32,	0,	0,	0
11030	DATA	*** # **	,163,139,	0,	40,192,	56,	6,	40,192,	56,	6,	56,	0,	0,	0,	0
11040	DATA	*** \$ **	,164,139,	0,	0,	72,	16,196,	56,	70,	16,	72,	0,	0,	0,	0
11050	DATA	*** % **	,165,139,	0,	4,130,	68,	34,	16,136,	66,130,	64,	0,	0,	0,	0,	0
11060	DATA	*** & **	,166,139,	0,	16,	10,	68,168,	18,160,	66,16,	12,	0,	0,	0,	0,	0
11070	DATA	*** ' **	,167,139,	0,	0,	0,	0,128,	64,	0,	0,	0,	0,	0,	0,	0
11080	DATA	*** ( **	,168,139,	0,	0,	128,	0,	64,	34,	28,	0,	0,	0,	0,	0
11090	DATA	*** ) **	,169,139,	0,	0,	0,	0,	28,	34,	64,	0,128,	0,	0,	0,	0
11100	DATA	*** * **	,170,139,	0,	16,	64,	20,	40,	16,	40,	20,	64,	16,	0,	0
11110	DATA	*** + **	,171,139,	0,	16,	0,	80,	32,	16,	8,	36,	0,	16,	0,	0
11120	DATA	*** , **	,172, 11,	0,	0,	0,	0,	8,	4,	10,	4,	1,	0,	0,	0
11130	DATA	*** - **	,173,139,	0,	16,	0,	16,	0,	16,	0,	16,	0,	16,	0,	0
11140	DATA	*** . **	,174,139,	0,	0,	0,	0,	4,	2,	4,	2,	0,	0,	0,	0
11150	DATA	*** / **	,175,139,	0,	0,	128,	64,	32,	16,	8,	4,	2,	0,	0,	0
11160	DATA	*** 0 **	,176,139,	0,	112,	16,136,	0,130,	0,	66,	32,	28,	0,	0,	0,	0
11170	DATA	*** 1 **	,177,139,	0,	0,	0,	128,	98,	24,	70,	0,	2,	0,	0,	0
11180	DATA	*** 2 **	,178,139,	0,	96,128,	18,128,	10,128,	6,	64,	2,	0,	0,	0,	0,	0
11190	DATA	*** 3 **	,179,139,	0,	128,	76,178,	0,146,	0,130,	0,	4,	0,	0,	0,	0,	0
11200	DATA	*** 4 **	,180,139,	0,	8,192,	56,	70,	8,	32,	8,	16,	8,	0,	0,	0
11210	DATA	*** 5 **	,181,139,	0,	128,	24,132,	32,130,	32,130,	96,	4,	0,	0,	0,	0,	0
11220	DATA	*** 6 **	,182,139,	0,	0,	12,128,	18,	64,	18,	32,	18,	12,	0,	0,	0
11230	DATA	*** 7 **	,183,139,	0,	128,	64,128,	32,144,	136,132,	130,128,	0,	0,	0,	0,	0,	0
11240	DATA	*** 8 **	,184,139,	0,	96,140,	16,130,	16,130,	16,	98,	12,	0,	0,	0,	0,	0
11250	DATA	*** 9 **	,185,139,	0,	0,	96,144,	8,144,	4,144,	2,	96,	0,	0,	0,	0,	0
11260	DATA	*** : **	,186,139,	0,	0,	0,	32,	20,	34,	20,	2,	0,	0,	0,	0
11270	DATA	*** ; **	,187, 11,	0,	0,	0,	64,	40,	68,	42,	4,	1,	0,	0,	0
11280	DATA	*** < **	,188,139,	0,	0,	128,	0,	64,	2,	36,	8,	16,	0,	0,	0
11290	DATA	*** = **	,189,139,	0,	0,	32,	8,	32,	8,	32,	8,	32,	8,	0,	0
11300	DATA	*** > **	,190,139,	0,	0,	16,	8,	36,	2,	64,	0,128,	0,	0,	0,	0
11310	DATA	*** ? **	,191,139,	0,	0,	96,128,	16,128,	8,128,	2,	64,	0,	0,	0,	0,	0
11320	DATA	*** **	,192,139,	0,	112,136,	34,136,	18,128,	66,	32,	28,	0,	0,	0,	0,	0
11330	DATA	*** A **	,193,139,	0,	254,	0,136,	64,	40,	16,	8,	4,	2,	0,	0,	0
11340	DATA	*** B **	,194,139,	0,	96,140,	18,128,	18,128,	98,252,	6,	0,	0,	0,	0,	0,	0
11350	DATA	*** C **	,195,139,	0,	64,132,	0,130,	0,130,	64,	34,	28,	0,	0,	0,	0,	0
11360	DATA	*** D **	,196,139,	0,	112,136,	4,128,	2,128,	98,152,	6,	0,	0,	0,	0,	0,	0
11370	DATA	*** E **	,197,139,	0,	128,	0,130,	16,130,	16,194,	56,	6,	0,	0,	0,	0,	0
11380	DATA	*** F **	,198,139,	0,	128,	0,130,	16,130,	16,194,	56,	6,	0,	0,	0,	0,	0
11390	DATA	*** G **	,199,139,	0,	64,152,	4,144,	0,130,	64,	34,	28,	0,	0,	0,	0,	0
11400	DATA	*** H **	,200,139,	0,	192,	56,	6,	16,	0,	16,192,	56,	6,	0,	0,	0

11410	DATA	"** I **",	201, 139,	0,	0, 128,	0, 226,	24, 134,	0,	2,	0,	0,	0
11420	DATA	"** J **",	202, 139,	0, 128,	96, 152,	4, 128,	2,	0,	2,	12,	0,	0
11430	DATA	"** K **",	203, 139,	0, 128,	2, 68,	8, 32,	16, 192,	56,	4,	0,	0,	0
11440	DATA	"** L **",	204, 139,	0,	0, 2,	0,	2, 128,	98,	24,	6,	0,	0
11450	DATA	"** M **",	205, 139,	0, 192,	56, 70,	0,	48,	0, 192,	56,	6,	0,	0
11460	DATA	"** N **",	206, 139,	0, 192,	56, 6,	8, 16,	32, 192,	56,	6,	0,	0,	0
11470	DATA	"** O **",	207, 139,	0,	96, 152,	4, 130,	0, 130,	64,	50,	12,	0,	0
11480	DATA	"** P **",	208, 139,	0,	96, 144,	0, 144,	0, 144,	96,	24,	6,	0,	0
11490	DATA	"** Q **",	209, 139,	0, 104,	10, 132,	0, 138,	0,	66,	32,	26,	0,	0
11500	DATA	"** R **",	210, 139,	0,	96, 130,	20, 136,	16, 128,	96,	24,	6,	0,	0
11510	DATA	"** S **",	211, 139,	0,	64, 140,	0, 146,	0, 146,	0,	98,	4,	0,	0
11520	DATA	"** T **",	212, 139,	0, 128,	0, 128,	0, 224,	24, 134,	0,	128,	0,	0,	0
11530	DATA	"** U **",	213, 139,	0, 192,	48, 12,	2,	0,	2, 192,	60,	12,	0,	0
11540	DATA	"** V **",	214, 139,	0, 128,	64, 32,	16,	8,	4,	0, 254,	0,	0,	0
11550	DATA	"** W **",	215, 139,	0, 192,	56, 6,	8, 16,	8, 196,	56,	6,	0,	0,	0
11560	DATA	"** X **",	216, 139,	0, 128,	64, 2,	36, 24,	32, 72, 132,	2,	0,	0,	0	
11570	DATA	"** Y **",	217, 139,	0, 128,	64, 0,	32,	0,	24, 38,	64, 128,	0,	0,	0
11580	DATA	"** Z **",	218, 139,	0, 128,	64, 162,	0, 146,	0, 138,	4,	2,	0,	0,	0
11590	DATA	"** [ **",	219, 139,	0, 128,	0, 128,	2, 128,	98,	24,	6,	0,	0,	0
11600	DATA	"** **",	220, 139,	0,	0,	0,	0,	2,	12,	48, 192,	0,	0
11610	DATA	"** ] **",	221, 139,	0,	0, 128,	96, 152,	6, 128,	2,	0,	2,	0,	0
11620	DATA	"** ^ **",	222, 139,	0,	0,	32, 64, 128,	0,	64,	0,	32,	0,	0
11630	DATA	"** **",	223, 11,	0,	1,	0,	1,	0,	1,	0,	1,	0
11640	DATA	"** **",	224, 139,	0,	0,	0,	0,	64, 128,	0,	0,	0,	0
11650	DATA	"** a **",	225, 139,	0,	0,	26, 36,	10, 32,	10, 32,	10,	4,	0,	0
11660	DATA	"** b **",	226, 139,	0,	0,	24, 36,	0,	34,	0, 226,	24,	6,	0
11670	DATA	"** c **",	227, 139,	0,	0,	32,	2,	32,	2,	32,	2,	16
11680	DATA	"** d **",	228, 139,	0, 192,	56, 6,	32,	2,	32,	2,	16,	12,	0
11690	DATA	"** e **",	229, 139,	0,	0,	16, 40,	2,	40,	2,	40,	18,	12
11700	DATA	"** f **",	230, 139,	0, 128,	0, 160,	0,	96,	24,	38,	0,	32,	0
11710	DATA	"** g **",	231, 11,	0,	0,	96, 24,	70,	0,	69,	0,	37,	24
11720	DATA	"** h **",	232, 139,	0,	0,	24, 38,	0,	32,	0, 224,	24,	6,	0
11730	DATA	"** i **",	233, 139,	0,	0,	0, 128,	34,	24,	38,	0,	2,	0
11740	DATA	"** j **",	234, 139,	0,	0,	0, 160,	24,	38,	0,	1,	0,	1
11750	DATA	"** k **",	235, 139,	0,	0,	32,	0,	18,	4, 136,	96,	24,	6
11760	DATA	"** l **",	236, 139,	0,	0,	0, 128,	98, 152,	6,	0,	2,	0,	0
11770	DATA	"** m **",	237, 139,	0,	0,	24, 38,	0,	56,	6,	32,	24,	38
11780	DATA	"** n **",	238, 139,	0,	0,	0, 24,	6,	32,	0,	32,	24,	38
11790	DATA	"** o **",	239, 139,	0,	0,	24, 36,	0,	2,	32,	0,	18,	12
11800	DATA	"** p **",	240, 11,	0,	0,	48, 72,	4,	64,	4,	96,	28,	3
11810	DATA	"** q **",	241, 11,	0,	0,	96, 24,	71,	0,	68,	0,	36,	24
11820	DATA	"** r **",	242, 139,	0,	0,	0, 32,	0,	32,	16,	0,	56,	6
11830	DATA	"** s **",	243, 139,	0,	0,	32,	4,	40,	2,	40,	2,	16
11840	DATA	"** t **",	244, 139,	0,	0,	32,	0,	34, 192,	50,	12,	32,	0
11850	DATA	"** u **",	245, 139,	0,	0,	50,	12,	2,	0,	2,	0,	50
11860	DATA	"** v **",	246, 139,	0,	0,	32,	16,	8,	0,	4,	0,	62
11870	DATA	"** w **",	247, 139,	0,	56,	6,	0,	4,	24,	0,	4,	48
11880	DATA	"** x **",	248, 139,	0,	32,	0,	18,	4,	8,	16,	36,	0
11890	DATA	"** y **",	249, 11,	0,	0,	64, 32,	16,	8,	4,	18,	33,	64
11900	DATA	"** z **",	250, 139,	0,	32,	0,	50,	0,	42,	0,	38,	0
11910	DATA	"** **",	251, 139,	0,	0, 128,	0, 130,	96,	26,	4,	16,	0,	0
11920	DATA	"** **",	252, 139,	0,	0,	0, 128,	64,	32,	8,	4,	2,	0
11930	DATA	"** **",	253, 139,	0,	0,	16,	64, 176,	12, 130,	0,	2,	0,	0
11940	DATA	"** ~ **",	254, 139,	0,	64,	32,	0,	32,	64, 128,	0, 128,	64,	0
11950	DATA	"** 0 **",	255, 139,	0,	176,	72, 132,	32, 146,	8,	66,	36,	26,	0
20000	DATA	"stop"										



# Indexmaker

```
=====
=   I N D E X M A K E R   =
=====
```

Indexmaker is a program to make the minidigital cassettetapes (MDCR's) easy accessible, a CHILDPROOF solution.

The program consists of two parts:

- a. the part TAPECONTROL
- B. the part INDEXMAKER.

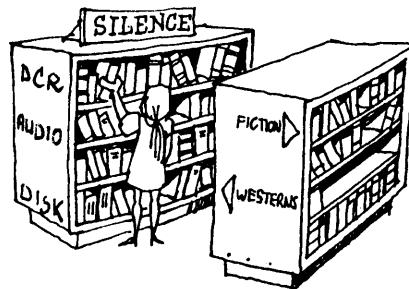
The best way to use it is:

- 1- have the program Indexmaker on an separate tape;
- 2- REWInd an unused MDCR and save the USER-program (#144-#1B4) at the start of that MDCR. See the Memocom-manual if you do no have th USER-program.
- 3- load Indexmaker, LIST 241 - 244, note the appearing advise;
- 4- perform the advise in the linenumbers 241 - 244;
- 5- insert the tape with the USER-program, type REW:VER, and SAVE the program 'TAPECONTROL'.
- 6- Behind TAPECONTROL you save the programs you want on that tape. Only one screen of 23 lines is used, so no more programs will appear in the Index.
- 7- Load Indexmaker from the other tape and run it. The program rewinds the tape and reads every programtitle, but no more than the first 25 will be stored in the datalines by the DIDACOM DATASTATEMENTGENERATOR.
- 8- If the program is ready the advise in the lines 240 - 243 will appear. You perform the advise.
- 9- Because of the fact you did already reserve enough space on the tape for TAPECONTROL, you can SAVE 'TAPECONTROL' immediate after USER, without disturbing the program following TAPECONTROL.
- 10- Remove the 'write-enable plug' from the MDCR-cassette, to make the system childproof.
- 11- Press the RESET-button to see it all happening.

When, later on, you save another program on this tape, that you want to be placed in the index, you simply load and run INDEXMAKER, and save the new TAPECONTROL on the space of the former version.

When using a LOADER for machinelanguage and BASIC-combinations in e.g. games for children, the last ones would be very gratefull, if you edit the datalines, which they should not read, and convert them in spaces. Then give the loader the name of the game.

Inno Broekman september 1985





```

10 REM indexmaker for mdcr-tapecontrole by inno broekman
11 REM avenbeek 98 2182 rz hillegom netherlands
12 REM telephone 02520 - 29279
20 MODE 0: CLEAR 10000: POKE #131,1: CALLM #F000: REM REW
30 POKE #75, #20: PRINT CHR$(12): AD$=""
40 FOR A%=1 TO 21: READ B%: AD$=AD#+CHR$(B%): NEXT A%
50 AD%=PEEK(VARPTR(AD$)) IOR (PEEK(VARPTR(AD$)+1) SHL 8)+1
60 R$="": T%=-1: X$="": POKE #131,1
80 IF T%>23 THEN PRINT : PRINT "too many programs for this index !!!": GOTO 240
90 PRINT CHR$(12): CALLM AD%
100 FOR A%=#BF7 TO #BF85 STEP -2: X%=X#+CHR$(PEEK(A%)): NEXT A%
110 GOSUB 270
120 IF T%=(-1) THEN R%=X%
130 IF X%=R% AND T%<>(-1) THEN GOTO 230
140 XX%=X%
149 X%=STR$(T%): X%=LEFT$(X%, LEN(X%)-2): IF T%<10 THEN X%=" "+X%
170 X%=X%+XX%
180 XX$="": X%=50
190 IF LEN(X%)>X% THEN X%=LEFT$(X%, X%)
195 READ I$
200 GOSUB 29056
210 T%=T%+1: GOTO 70
220 IF R$="-object-user" THEN CALLM #F000: REM REW: SKIP1: LOOK
221 IF R$<>"-object-user" THEN CALLM #F000: REM REW: LOOK
230 PRINT CHR$(12)
240 PRINT : PRINT " Index is ready."
241 PRINT : PRINT " Type < CLEAR 20000:EDIT1000-4000 >"
242 PRINT : PRINT "Type <BREAK>, <BREAK>, NEW, POKE#135,2, and SAVE TAPECONTROLE'"
243 PRINT : PRINT "Delete linenumbers between 2900 en 2999 ."
244 END
250 GOTO 220
260 DATA #F5, #C5, #D5, #E5, 1, #40, 0, #11, #B1, #80, #21, #9E, #E6, #CD, #CE, 2, #E1, #D1, #C1
, #F1, #C9
270 IF LEFT$(X%, 1)="0" THEN X$="-basic -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
280 IF LEFT$(X%, 1)="1" THEN X$="-object -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
290 IF LEFT$(X%, 1)="2" THEN X$="-array -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
300 IF LEFT$(X%, 1)="#" THEN X$="-assem -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
310 IF LEFT$(X%, 1)="P" THEN X$="-pascal -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
320 IF LEFT$(X%, 1)="#" THEN X$="-tekst -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
330 IF LEFT$(X%, 1)="#" THEN X$="-s.p.l. -"+RIGHT$(X%, LEN(X%)-1): GOTO 340
340 RETURN
1000 REM mdcr tapecontrole
1010 CLEAR 1000: MODE 0: COLORT 8 0 8 0: N%=0
1020 PRINT CHR$(12): CURSOR 10,16: PRINT CHR$(1): " M D C R T A P E C O N T R O L
E "; CHR$(1)
1030 WAIT TIME 40: PRINT CHR$(12)
1040 PRINT CHR$(12): RESTORE
1050 FOR I%=1 TO 23: READ P$: PRINT P$: NEXT I%
1070 POKE #74, 3: POKE #75, #20
1071 I1%=24: FOR I%=#B3C9 TO #BFD3 STEP 134
1072 QQQ%=PEEK(I%): IF QQQ%=#2D THEN I1%=I1%-1
1073 NEXT I%
1080 FOR X%=23 TO I1% STEP -1
1090 CURSOR 10, X%: PRINT CHR$(1)
1100 FOR Z%=1 TO 200
1110 IF GETC<>0 GOTO 1160
1120 NEXT Z%
1130 CURSOR 10, X%: PRINT CHR$(32)
1140 NEXT X%
1150 GOTO 1080

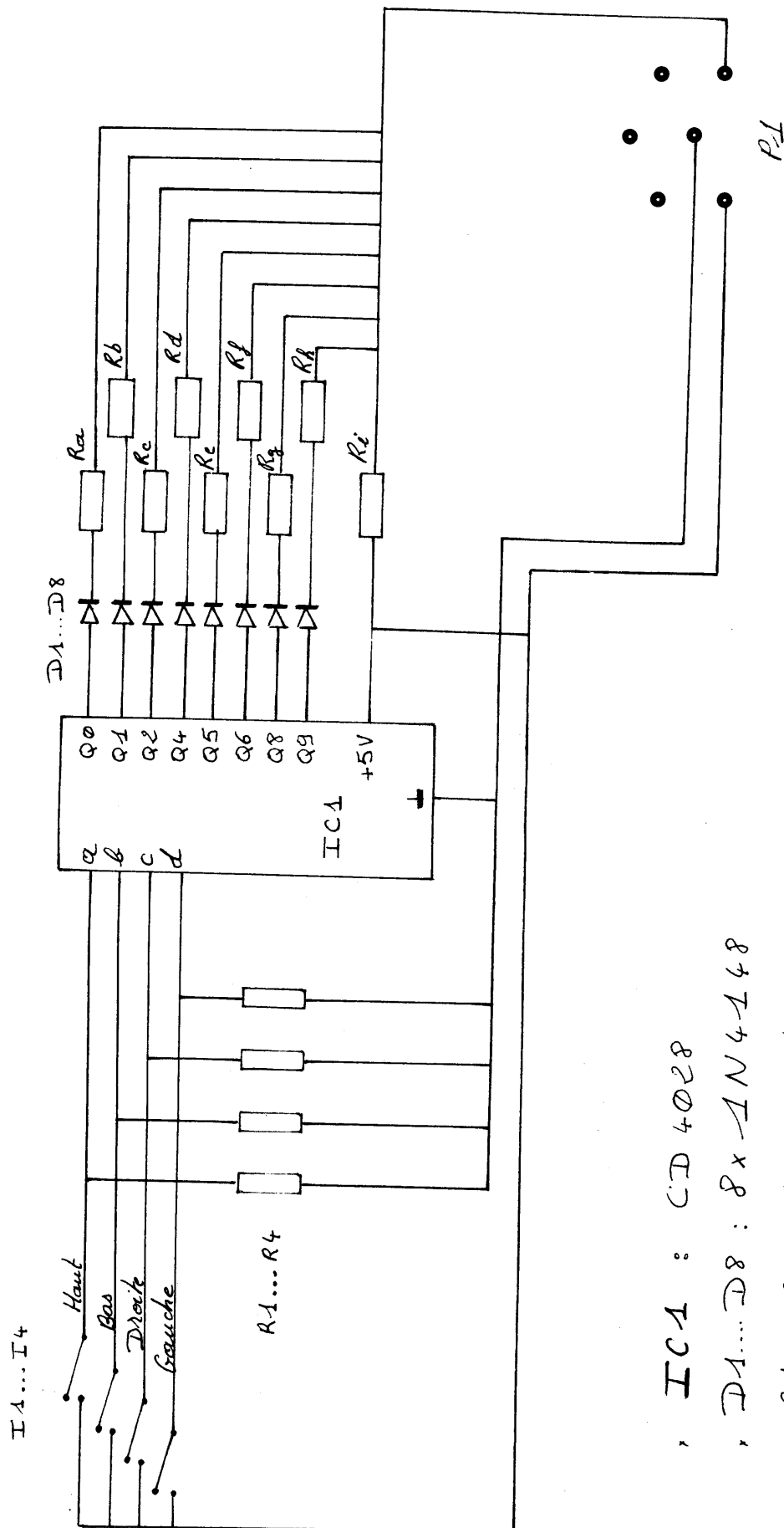
```

```

1160 CN%=23-X%:RESTORE
1170 IF CN%=0 THEN READ P#:N%=1:GOTO 1230
1180 FOR I%=0 TO CN%:READ P#
1190 IF LEFT$(P#,3)=" " THEN CURSOR 0,1:PRINT CHR$(1);" MEMOCOM does not re
cognize this program ! ";CHR$(1):WAIT TIME 200:GOTO 1070
1200 IF CN%=X% THEN GOTO 1070
1210 IF CN%>22 THEN GOTO 1240
1220 NEXT I%
1230 PRINT CHR$(12):CURSOR 0,2:PRINT CHR$(1);" MEMOCOM is searching for : ";CHR
$(13);" ";CHR$(1);P#;". ";CHR$(1):GOTO 1250
1240 PRINT CHR$(12):CURSOR 0,2:PRINT CHR$(1);" MEMOCOM skips over all programs.
";CHR$(1):GOTO 1340
1250 IF N%=1 THEN GOTO 1300
1260 IF CN%>22 THEN GOTO 1340
1270 FOR LOOP%=0 TO CN%-1
1280 CALLM #F000:REM DCR0:SKIP1
1290 NEXT LOOP%
1300 IF MID$(P#,4,1)="b" THEN LOAD
1310 IF MID$(P#,4,1)<>"b" THEN CURSOR 0,23:CALLM #F000:REM LOOK
1320 PRINT :PRINT "You are just about the start of the chosen program."
1330 PRINT :PRINT "Load accordingly the FILE-type,":PRINT :PRINT "by e.g. a mai
nprogram for textprocessing ":END
1340 CALLM #F000:REM DCR0:SKIP9:SKIP9:SKIP9
1350 END
2900 REM one dataline contains fifty spaces
2970 DATA "
2980 DATA "
2990 DATA "
3000 DATA "
3010 DATA "
3020 DATA "
3030 DATA "
3040 DATA "
3050 DATA "
3060 DATA "
3070 DATA "
3080 DATA "
3090 DATA "
3100 DATA "
3110 DATA "
3120 DATA "
3130 DATA "
3140 DATA "
3150 DATA "
3160 DATA "
3170 DATA "
3180 DATA "
3190 DATA "
3200 DATA "
3210 DATA "
3220 DATA "
3230 DATA "
4000 DATA "
29054 REM ##### didacom datastatementgenerator #####
29056 XA%=PEEK(#124)+PEEK(#125)*256+5:XS#="":XS#=CHR#(#22)
29058 FOR XI%=0 TO X%-LEN(X#):XS#=XS#+ " ":NEXT XI%
29061 X#=CHR#(#22)+X#+XS#
29062 FOR XI%=0 TO X%+1
29063 POKE XA%,ASC(MID$(X#,XI%,1))
29064 XA%=XA%+1:NEXT XI%
29067 RETURN

```

# Joystick interface



, IC1 : CD 4028

, D1...D8 : 8 x 1N4148

, R1...R4 : 4 x 10 k $\Omega$

, P1 : Connecteur DIN 6 broches

```

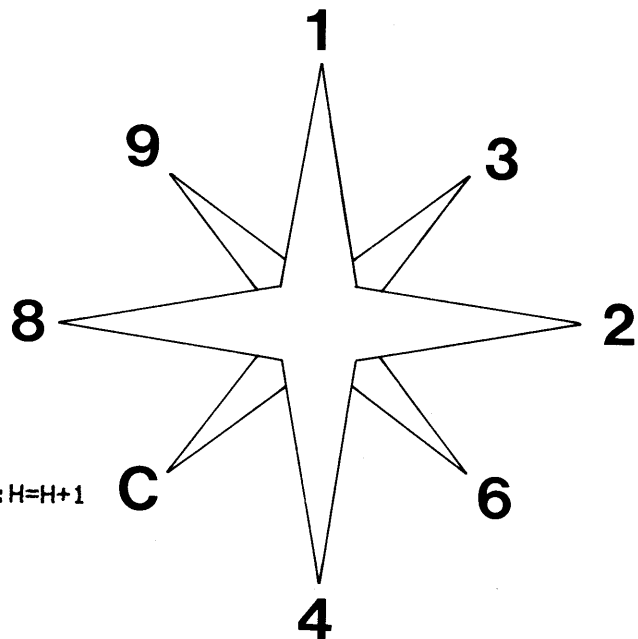
1  REM #####
2  REM ###  D E M O  P D L  ###
3  REM #####
4  REM
5  REM
6  REM
7  REM *****
8  REM * MENU+SELECTION *
9  REM *****
10 MODE 0:PRINT CHR$(12)
20 COLORT 5 15 5 0:POKE #75,32
30 POKE #B900,#FF:VPOS=1
50 CURSOR 20,21:PRINT "M E N U"
60 CURSOR 20,20:PRINT "-----"
70 CURSOR 10,16:PRINT "(1) Mode d'emploi"
80 CURSOR 10,14:PRINT "(2) Curseur"
90 CURSOR 10,12:PRINT "(3) Aide au dessin"
92 CURSOR 11,4:PRINT "POINTEZ VOTRE RUBRIQUE AVEC LE PDL"
94 CURSOR 11,3:PRINT "CHOISISSEZ EN PRESSANT LE BOUTON"
100 CURSOR 10,10:PRINT "(4) Menu"
110 IF (PEEK(#FD00) IAND #30)<>0 THEN ON VPOS GOTO 10,300,1000,500
120 SP=((PDL(0) IAND #50)/16)
130 IF SP<>0 THEN GOSUB 200
140 GOTO 110
197 REM *****
198 REM * S.P. POSITIONNEMENT CURSEUR *
199 REM *****
200 OLD=VPOS:WAIT TIME 11
210 IF SP=4 AND VPOS<>1 THEN VPOS=VPOS-1
220 IF SP=1 AND VPOS<>4 THEN VPOS=VPOS+1
230 POKE (#B900+(OLD-1)*268),0
240 POKE (#B900+(VPOS-1)*268),#FF
250 RETURN
297 REM *****
298 REM * Aide au dessin *
299 REM *****
300 COLORG 9 1 12 3
310 PRINT CHR$(12):MODE 6A
320 PRINT "* -Tracez avec le PDL"
330 PRINT "* -Levez le crayon avec le bouton"
340 PRINT "* -Retour au menu:Sortir de l'ecran avec la trace"
360 WAIT TIME 2:V=YMAX/2:H=XMAX/2
370 FILL 0,0 XMAX,YMAX 12
380 FILL 2,2 XMAX-6,YMAX-6 9
390 A=(PDL(0) IAND #F0)/16
400 IF (A IAND 1)<>0 THEN V=V+1:IF V>YMAX-7 THEN 10
410 IF (A IAND 2)<>0 THEN H=H+1:IF H>XMAX-7 THEN 10
420 IF (A IAND 4)<>0 THEN V=V-1:IF V<3 THEN 10
430 IF (A IAND 8)<>0 THEN H=H-1:IF H<3 THEN 10
440 IF (PEEK(#FD00) IAND #30)=0 THEN DOT H,V 21
450 WAIT TIME 2:GOTO 390
497 REM *****
498 REM * Mode D'emploi *
499 REM *****
500 PRINT CHR$(12)
550 PRINT "      INTERFACE DE CONVERSION DIGITAL/ANALOGIQUE"
560 PRINT "      ====="
570 PRINT :PRINT :PRINT :PRINT " Principe : "
590 PRINT " -----"
600 PRINT "L'interface D/A permet de connecter un manche de commande"
610 PRINT "Type 'contacteur' sur l'une des entrees analogiques du DAI"
620 PRINT "Pour ce faire il suffit de connecter l'interface entre le"
630 PRINT "manche et L'ordinateur.":PRINT :PRINT
640 PRINT " Utilisation":PRINT " -----"
650 PRINT "Pour savoir quel est le(s) contact(s) enfonce(s) il suffit"
660 PRINT "de lire la valeur sur le port correspondant par"

```

```

670 PRINT "1' instruction PDL. La valeur qui est renvoye se traduit"
680 PRINT "comme suit:"
690 PRINT :PRINT :PRINT "PRESSEZ LE BOUTON POUR CONTINUER..."
700 IF (PEEK(#FD00) IAND #30)=0 THEN WAIT TIME 2:GOTO 700
710 PRINT :PRINT :PRINT "Les quatres bits de poids faible ne sont pas sig
nificatifs,"
720 PRINT "Chaque bits de poid fort indique une direction."
730 PRINT "Soit pour les directions en partant du plus faible"
740 PRINT :PRINT " * Haut":PRINT " * Droite":PRINT " * Bas":PRINT " * Gau
che":PRINT
750 PRINT :PRINT "Plus pratiquement on aura:"
760 PRINT "- lire le port avec"
770 PRINT "      PORT=(PDL(0) IAND #F0)/16"
780 PRINT "- PORT aura les valeurs suivantes pour chaque directions"
790 PRINT " (en HEXAdecimal)":PRINT
800 PRINT "          9  1  3":PRINT
810 PRINT "          8      2":PRINT
820 PRINT "          C  4  6"
830 PRINT :PRINT "          (ESSAYEZ LE PDL AVEC HAUT & BAS)"
896 REM *****
897 REM * DEMO COULEUR *
898 REM *****
900 LIGN=#B3E4:COUL=5
910 A=(PDL(0) IAND #F0)/16
920 IF (PEEK(#FD00) IAND #30)<>0 THEN 10
930 IF (A IAND 2)<>0 THEN COUL=(COUL+1) MOD 16:WAIT TIME 11
940 IF (A IAND 8)<>0 THEN COUL=ABS(COUL-1):WAIT TIME 11
950 IF (A IAND 1)<>0 AND LIGN<#BFEF THEN LIGN=LIGN+134:WAIT TIME 11
960 IF (A IAND 4)<>0 AND LIGN>#B3E5 THEN LIGN=LIGN-134:WAIT TIME 11
970 POKE LIGN,(#C0 IOR COUL)
980 WAIT TIME 1:GOTO 910
997 REM *****
998 REM * CURSEUR *
999 REM *****
1000 WAIT TIME 17:MODE 0:PRINT CHR$(12)
1010 COLORT 9 12 12 1
1020 POKE #75,#5F
1030 V=20:H=0
1050 IF (PEEK(#FD00) IAND #30)<>0 THEN 10
1060 A=(PDL(0) IAND #F0)/16
1070 IF (A IAND 1)<>0 AND V<23 THEN V=V+1
1080 IF (A IAND 2)<>0 AND H<59 THEN H=H+1
1090 IF (A IAND 4)<>0 AND V>0 THEN V=V-1
1100 IF (A IAND 8)<>0 AND H>0 THEN H=H-1
1110 CURSOR H,V
1120 A=GETC:WAIT TIME 1
1130 IF A<>0 AND H<59 THEN PRINT CHR$(A);:H=H+1
1140 GOTO 1050
1998 REM
1999 REM
2000 REM *****
2001 REM * F I N *
2002 REM *****
2010 REM .....C.DUFOUR

```



## Inleiding.

Lineaire regressie is een rekenmethode uit de statistiek, waarmee men de rechte lijn bepaalt, die het best "past" bij een aantal punten in een grafiek. Als je deze methode wilt gebruiken, moet je van te voren wel weten, denken of hopen dat er tussen de variabelen die in de grafiek zijn uitgezet een lineair verband is. Zo'n verband komt vaak voor.

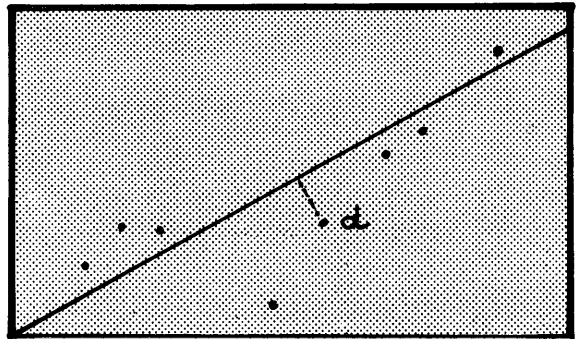
Een paar voorbeelden:

1. Bij het maken van een niet al te lange wandeling is er een lineair verband tussen de looptijd en de afgelegde afstand: in twee maal zo veel tijd kom je twee maal zo ver.
2. In de natuurkunde is het verband tussen de elektrische spanning en de stroomsterkte bij een weerstand lineair.
3. In de scheikunde bestaat er binnen zekere grenzen vaak een lineair verband tussen reactiesnelheid en de concentratie van de reagerende stoffen.
4. In de economie: gedurende een niet te lange periode kan de consumptie een lineair verband hebben met de tijd.
5. Wie spaarzaam en regelmatig leeft, merkt dat zijn spaargeld lineair met de tijd toeneemt.

Men bepaalt een rechte lijn, dat wil het volgende zeggen: een rechte lijn wordt gegeven door het functievoorschrift  $y = ax + b$ , waarin  $a$  de richtings-coëfficiënt (rico) is en  $b$  het snijpunt met de  $y$ -as. Met lineaire regressie kun je voor een bepaalde situatie  $a$  en  $b$  berekenen van de lijn waar de gegeven punten gemiddeld het dichtst bij liggen. Dit is niet de plaats om diep op de theorie in te gaan, toch wil ik er iets over zeggen.

## Het principe.

In onderstaande figuur is een aantal punten in een diagram getekend plus een willekeurige rechte lijn.



Voor elk punt kun je de verticale afstand  $d$  tot aan de lijn bepalen. De best passende lijn loopt nu zo, dat de som van de kwadraten van al deze afstanden minimaal is (kleiner dan voor elke andere lijn).

## Het nut.

Wat is nu het nut van het berekenen van zo'n lijn? Welnu, dat kan verschillend zijn: in de natuur- en scheikunde hebben de rico en het snijpunt met de  $y$ -as vaak een zekere betekenis. Zet men elektrische spanning verticaal uit tegen horizontaal de stroomsterkte, dan is de best passende lijn er een die door de oorsprong gaat (tenzij je meetfouten maakte, hetgeen je dan gelijk weet) en waarvan de rico de weerstand in ohm voorstelt. Ook kun je de methode gebruiken om een toekomstverwachting uit te spreken (weet je  $a$  en  $b$ , dan kun je bij elke  $x$  de bijbehorende  $y$  uitrekenen). Stel dat de persoon uit het vijfde voorbeeld in het bezit is van een elektronisch apparaat van een zeer bekend Amerikaans merk, dat met computer de eerste drie letters gemeenschappelijk heeft (een van de weinige overeenkomsten). Stel dat deze persoon er achter komt, dat programmeren meer is dan POKE-n. Dan kan hij met lineaire regressie een nauwkeurige schatting maken wanneer hij genoeg geld heeft om een DAI te kopen.

## Correlatie.

Als je het verband tussen twee variabelen gaat onderzoeken weet je soms vooraf niet of er wel sprake van een lineair verband is. Stel je hebt onderzoek verricht naar het verband tussen twee variabelen P en Q waarvoor geldt: Q hangt van P af, zodat een verandering van P er de oorzaak van is dat Q verandert (P is dan de onafhankelijk variabele en Q is de afhankelijk variabele; je hoort dan P horizontaal en Q verticaal uit te zetten). In de regressietheorie is de zogenaamde correlatiecoëfficiënt  $r$  zo gedefinieerd, dat hij een maat is voor hoe goed (of hoe slecht) de berekende lijn bij de punten past. Deze  $r$  ligt altijd tussen  $-1$  en  $+1$ . Als  $r=0$ , dan is er helemaal geen verband tussen P en Q, dus ook geen lineair verband. Als  $r$  een getal is, dat in de buurt van  $0$  ligt, dan hangt Q wel van P af, maar ook van andere factoren. Of Q hangt wel uitsluitend van P af, maar het verband is niet lineair. Welke van deze twee mogelijkheden het is, kun je zelf in de grafiek zien. Als  $r=-1$  of  $r=+1$  dan is het zo dat Q alleen maar van P afhangt en het verband is volmaakt lineair (er is een lijn waar alle punten op liggen). Tevens bevatten de waarden van P en Q dan geen meetfouten. Overigens:  $r$  is negatief als de rico van de lijn dat is.

## Een truc.

Soms weet je vooraf wel zeker dat je met een lineair verband te maken hebt, en ook dat de gezochte lijn door de oorsprong van het assenstelsel hoort te gaan, bijvoorbeeld bij het verband tussen spanning en stroomsterkte. Als je in zo'n geval de parameters van de best passende lijn hebt berekend en deze blijkt niet door de oorsprong te gaan, wat doe je dan? Dat ligt aan de oorzaak, en die kan verschillend zijn: zit er in de meetwaarden een systematische fout, dan rest je niets anders dan die fout op te sporen en nogmaals te meten, maar dan goed. Ook kan de oorzaak liggen in een (te) klein aantal punten. Daardoor is er een statistische onzekerheid dat de berekende lijn ook echt de beste is.

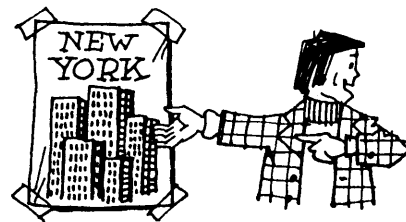
In dat geval is er een truc om een nieuwe lijn te vinden die wel aan de eis  $b=0$  voldoet. Een "smerige" truc, dat wel: voeg aan je punten een zeer groot aantal malen de oorsprong toe. Dit heeft een duidelijk nadeel: vergroting van het aantal punten betekent in principe vergroting van de zekerheid dat de gevonden lijn het verband juist weergeeft, maar in dit geval is dat natuurlijk slechts een schijnzekerheid. Dit betekent dat door deze truc de correlatiecoëfficiënt kunstmatig dichter naar plus of min 1 wordt gebracht. Kun je met vrij grote zekerheid stellen dat twee variabelen lineair samenhangen als  $r > 0.90$ , na toepassing van de truc zul je een waarde boven  $0.99$  moeten vinden, anders was de truc zeker niet verantwoord.

## Een programma.

Ik hoop dat het bovenstaande u in staat stelt om te beoordelen of lineaire regressie een rekentechniek is waar u misschien eens nuttig gebruik van kunt maken. Binnenkort is een programma beschikbaar waarmee u regressielijnen die naar keuze wel of niet door de oorsprong gaan kunt berekenen, en dat tevens lijn en punten grafisch weergeeft, voor een gemakkelijke interpretatie van de resultaten.

Aad de Bruijn  
Heerenlaan 14  
NL-3218 VL Heenvliet.

```
10 REM NEW-YORK: AUTHEUR:Sebastien DUBOURG 1983
100 MODE 0: CLEAR 1000: COLORG 9 14 3 0: MODE 5
110 Y=YMAX
120 DRAW 0, Y XMAX, Y 0
130 PAS=0.2*(YMAX-Y+1.0)+2.0
140 Y=Y-PAS
150 IF Y>30.0 GOTO 120
160 RESTORE: REM DESSIN DES IMMEUBLES
170 FOR I=1.0 TO 17.0
180 READ X1, Y1, X2, Y2
190 FILL X1, Y1 X2, Y2 0
200 NEXT
250 DATA 0, 20, 335, 40
260 DATA 40, 40, 60, 170, 45, 170, 55, 180, 49, 180, 51, 190
270 DATA 0, 40, 50, 120
280 DATA 80, 40, 130, 135, 85, 135, 125, 145, 95, 145, 115, 150
290 DATA 70, 40, 100, 100
300 DATA 130, 40, 145, 55, 185, 40, 205, 110
310 DATA 145, 40, 185, 170
320 DATA 205, 40, 275, 60, 220, 60, 255, 70, 230, 70, 250, 151
330 DATA 275, 40, 335, 65, 300, 65, 335, 75
340 REM DESSIN DE LA LUNE
350 FOR I=0.0 TO 2.0*PI STEP PI/35.0
360 DRAW 300, 200 300.5+10*COS(I), 200.5+10*SIN(I) 14
370 NEXT
380 REM LES ENTREPOTS
390 FOR I=1.0 TO 16.0
400 READ X1, Y1, X2, Y2
410 DRAW X1, Y1 X2, Y2 9
420 NEXT
450 DATA 175, 25, 180, 25, 180, 25, 180, 35, 190, 35, 190, 25
460 DATA 180, 35, 190, 35, 130, 20, 130, 25, 200, 20, 200, 25
470 DATA 130, 25, 145, 25, 190, 25, 200, 25
480 REM ENTREPOT 2
490 DATA 245, 20, 245, 40, 245, 40, 260, 40, 260, 40, 260, 35
500 DATA 260, 35, 275, 35, 275, 35, 275, 25, 295, 25, 315, 25
510 DATA 315, 25, 315, 32, 315, 32, 335, 32
520 REM TOITS
530 N=10.0: MAX=30.0: X=145.0: Y=24.0: GOSUB 600
540 N=20.0: MAX=20.0: X=275.0: Y=22.0: GOSUB 600
550 GOTO 700
600 FOR I=0.0 TO MAX
610 A=A+SGN(N/2.0-(I MOD (N+1.0)))
620 DOT X+I, Y+A 9
630 NEXT
640 RETURN
```





```

700 REM LUMIERES A REFLETS
710 Y=24.0:C=14.0
715 FOR X=20.0 TO 120.0 STEP 20.0
720 GOSUB 800:NEXT
730 X=215.0:GOSUB 800
740 X=230.0:C=3.0:GOSUB 800
750 X=185.0:Y=35.0:GOSUB 800
760 GOTO 900
800 REM LUMIERE
810 DOT X,Y C
820 DOT X-1,Y C:DOT X+1,Y C
830 DOT X,Y-1 C:DOT X,Y+1 C
840 REM REFLETS
850 FOR J=15.0 TO 0.0 STEP -1.0
860 A=A+SGN(1.5-J MOD 4.0)
870 DOT X+A,J C
880 NEXT
890 RETURN
900 REM INIT LUMIERES
910 FLAG=0.0:FOR I=1.0 TO 1100.0
930 GOTO 1000
940 REM
950 FOR I=1.0 TO 5.0
960 FLAG=1.0
1000 A=RND(11.0)+1.0:ON A GOSUB 3000,3500,4000,4500,5000,5500,6000,6500,7000,
7500,7600
1010 NEXT
1020 A=RND(11.0)+1.0:FLAG=0.0:ON A GOSUB 3000,3500,4000,4500,5000,5500,6000,6
500,7000,7500,7600
1030 GOTO 950
3000 C=14.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
3010 X1=0.0:X2=47.0
3020 Y1=41.0:Y2=120.0
3030 GOSUB 8000
3035 GOSUB 8500
3040 RETURN
3500 REM 2
3510 C=6.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
3520 Y=RND(128.0)+41.0
3530 IF Y>120.0 THEN X=RND(18.0)+41.0:GOTO 3600
3540 X=RND(8.0)+51.0
3600 DOT X,Y C
3610 RETURN
4000 REM 3
4010 C=13.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
4020 Y=RND(59.0)+40.0
4030 X=RND(8.0)+71.0
4040 DOT X,Y C
4050 RETURN
4500 REM 4
4510 C=8.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
4520 Y=RND(107.0)+41.0
4530 X1=81.0:X2=128.0
4540 IF Y>145.0 THEN X1=96.0:X2=113.0:GOTO 4600
4550 IF Y>135.0 THEN X1=86.0:X2=123.0
4600 GOSUB 9000
4610 RETURN

```

```

5000 REM 5
5010 C=3.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
5020 X1=110.0:X2=142.0:Y1=31.0:Y2=54.0
5030 GOSUB 8000:GOSUB 8500
5040 RETURN
5500 REM 6
5510 C=15.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
5520 Y1=41.0:Y2=169.0:X1=146.0:X2=182.0
5530 GOSUB 8000:GOSUB 8500
5540 RETURN
6000 REM 7
6010 C=7.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
6020 Y=RND(69.0)+40.0
6030 X1=186.0:X2=203.0:GOSUB 9000
6040 RETURN
6500 REM 8
6510 C=10.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
6520 Y=RND(109.0)+41.0
6530 IF Y>70.0 THEN X=RND(18.0)+231.0:GOTO 6600
6540 IF Y>60.0 THEN X=RND(33.0)+221.0:GOTO 6600
6550 X=RND(68.0)+206.0
6600 DOT X,Y C
6610 RETURN
7000 REM 9
7010 C=11.0:IF FLAG=1.0 THEN C=0.0:FLAG=0.0
7020 Y1=41.0:Y2=74.0:GOSUB 8000
7030 X1=276.0:X2=333.0
7040 IF Y>63.0 THEN X1=301.0:X2=333.0
7050 GOSUB 8500
7060 RETURN
7500 REM 10
7510 C=INT(RND(15.0))+1.0
7520 IF FLAG=1.0 THEN C=0.0:FLAG=0.0
7530 X=RND(130.0)
7540 Y=RND(14.0)+26.0
7550 DOT X,Y C
7560 RETURN
7600 C=INT(RND(15.0))+1.0
7610 IF FLAG=1.0 THEN C=0.0:FLAG=0.0
7620 X=RND(43.0)+201.0
7630 Y=RND(19.0)+26.0
7640 DOT X,Y C
7650 RETURN
8000 REM 2 X 3
8010 Y=INT(RND((Y2-Y1)/2.0))*2.0+Y1
8020 Y=INT(Y)
8030 RETURN
8500 X=INT(RND((X2-X1)/3.0))*3.0+X1
8510 X=INT(X)
8520 FILL X,Y-1 X+2,Y C
8530 RETURN
9000 REM 2X1
9500 X=INT(INT(RND((X2-X1)/2.0))*2.0+X1)
9520 DRAW X,Y X+1,Y C
9530 RETURN

```

SPL V1.1 PAGE 1

ON ERROR GOTO V2.0

```

1          TITL      'ON ERROR GOTO V2.0'
2          ;AUTHEUR:LOOIJJE DAInamic 83-16
3          ;REV. :Sebastien DUBOURG 27/8/1983
4          ;to activate
5          ;*UT <RETURN>
6          ;>V5 C6FD-300 <CURSOR LEFT>
7          ;>B
8          ;*POKE 6,LINENUMBER IAND #FF:POKE 7,LINENUMBER SHR 8
9          ;
10         ;To deactivate
11         ;POKE 6,0:POKE 7,0
12         ;Explanations:see DAInamic 83-15 & 83-16:Looije
13         ;-----
14         PUT      "H"
15         PUT      "h"
16         ;
17         ERRLIN  EQU      6H
18         ;
19         ORG      300H
20         PUSH PSW
21         ;
22         LHL D    ERRLIN      ;Is ON ERROR GOTO
23         MOV A,H      ;activate ?
24         OR A,L
25         JZ         OUT      ;If no exit
26         ;
27         LXI H    0AH      ;Offset to find
28         DAD SP      ;original caller
29         ;
30         MOV A,M      ;Is it an error message
31         CPI      53H      ;being printed ?
32         JNZ      OUT      ;)
33         INX H      ;)
34         MOV A,M      ;)If no exit
35         CPI      ODAH     ;)
36         JNZ      OUT      ;)
37         INX H
38         MOV A,M
39         CPI      40H
40         JZ         ONERR
41         ;
42         OUT      POP PSW      ;Restore CPU registers
43         JMP      OC6FDH     ;& cont the RST 4
44         ;
45         ONERR   PUSH B
46         LHL D    ERRLIN
47         CALL    OCAF6H     ;FIND this line in BASIC
48         JNC     UNDEFL
49         ;
50         MOV B,H      ;BC=START LINE
51         MOV C,L
52         LXI H    100H
53         MVI A    15H      ;CLEAR GOS/NEXT
54         LOOP   MVI M    OH
55         INX H
56         DCR A
57         JNZ     LOOP
58         ;

```

```

59          DI
60          STA      131H          ;output to screen
61          LDA      40H
62          ANI      3FH
63          STA      40H
64          STA      OFD06H
65          EI
66          POP PSW
67          STA      NMBR
68          LXI SP   OF900H
69          JMP      OC892H
70          ;
71 UNDEFN LXI H      0H
72          SHLD     ERRLIN
73          MVI A    4H
74          JMP      OD9F5H
75          ;
76 NMBR     DS      1H
77          ;
78          END

```

SPL V1.1 PAGE 1

```

1          TITL     'UT LOADER'
2          ;*****
3          ;*              UT LOADER              *
4          ;*              =====              *
5          ;* FUNCTION                                *
6          ;* ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' ' *
7          ;* -LOAD THE FIRST UT FILES ON TAPE      *
8          ;*                                         *
9          ;* IN A BASIC PGM USE CALLM#300          *
10         ;*                                         COPYRIGHT:*
11         ;* SEBASTIEN DUBOURG & DAINAMIC janv 1982 *
12         ;*****
13 CURRNT EQU      256D
14 ROPEN  EQU      718D
15 RBKL   EQU      721D
16 RCLO   EQU      724D
17 ERRLD  EQU      53928D
18 CRLF   EQU      56670D
19 POPRET EQU      49485D
20 ENTRY  ORG      768D
21          PUSH PSW          ;SAVE
22          PUSH B
23          PUSH D
24          PUSH H
25          LXI H      OD
26          SHLD     CURRNT
27          LXI B      12799D          ;NO NAME T1
28          CALL     ROPEN          ;MOTOR
29          LXI H      TEMP          ;BGN SAVE AREA
30          LXI D      TEMP+2D
31          CALL     RBKL          ;LOAD
32          LXI D      45056D
33          LHLD     TEMP
34          CC       RBKL          ;IF NO ERROR
35          CALL     RCLO          ;STOP MOTOR
36          JNC     ERRLD          ;ERROR
37          CALL     CRLF
38          JMP      POPRET
39 TEMP    DW      OD
40          DW      OD
41          END

```

## DAI

## TOUT SAVOIR SUR LES FICHIERS

Les heureux possesseurs d'un Dai agrémenté d'un système de disquettes Ken-Dos doivent absolument avoir cet utilitaire dans leur boîte à outils. En effet, le programme Ken-Zapper liste en clair les attributs des fichiers disque, leur localisation en pistes et secteurs, et même le code de verrouillage, s'il y en a un. Pratique, pour les étourdis, mais à ne pas laisser traîner entre toutes les mains !

Pour fonctionner correctement, ce programme ne doit être lancé qu'après s'être assuré que le tampon de DIRECTORY se trouve bien à sa place habituelle en mémoire. Pour plus de sûreté, faire

"BUF" avant toute chose. Ensuite, lancer (RUN) Ken-Zapper. Il demande d'appeler le DIRECTORY désiré, puis de le relancer par RUN 100. C'est tout. Le programme fait le reste.

Comme j'étais pressé, quand je l'ai écrit, je ne me suis pas trop embarrassé de finasseries. C'est ainsi que le numéro du fichier testé (ligne 180) n'est juste que pour le premier DIRECTORY, et que le STATUS n'est décodé en clair que pour certains états (lignes 260 à 280). Vous pouvez ajouter ce qui manque.

Outre le code de protection (pour les étourdis qui bloquent un fichier, puis

**Ken-Zapper**

Programme pour Dai et Ken-Dos

Auteur Alain Mariatte

Copyright LIST et l'auteur

```

10  MODE 0:COLORT 0 0 0:PRINT CHR$(12);CHR$(13)
20  PRINT "          KEN      Z  A  P  P  E  R"
30  PRINT "          *****"
40  PRINT
50  PRINT :PRINT
60  PRINT "CHARGER LE ZAPPER. APPELER LE BLOCK DE DIRECTORY"
70  PRINT "DESIRE (ex: DIR2) PUIS <RUN 100>. RECOMMENCER AUTANT"
80  CURSOR 0,1:STOP
100  GOSUB 750
110  GOSUB 130
120  END
130  REM DECODE 1 SECTOR DIR
140  REM
150  FOR I=#AF50 TO #B350 STEP 32
160  IF PEEK(I+26)=0 THEN RETURN:REM SECTOR=0 :FICHIER NON CREE
170  FN=PEEK(I)
180  PRINT :PRINT "FILE NUMBER          :";FN-#20
190  P=I
200  P=P+1:L=PEEK(P):P=P+1
210  N$="":FOR K=P TO P+L-1:N$=N$+CHR$(PEEK(K)):NEXT
220  PRINT "FILE NAME          : ";N$
230  P=P+16:ST=PEEK(P)
240  PRINT "STATUS          : ";
250  IF ST<>0 AND ST<>1 AND ST<>#10 THEN PRINT "#";HEX$(ST):GOTO 290:REM #28 DN
A ASS.
260  IF ST=0 THEN PRINT "OPENED":GOTO 290
270  IF ST=1 THEN PRINT "LOCKED":GOTO 290
280  IF ST=#10 THEN PRINT "DELETED":GOTO 290
290  P=P+17:T=PEEK(P)
300  PRINT "FILE TYPE          : ";
310  IF T=0 THEN PRINT "BASIC":GOTO 400
320  IF T=1 THEN PRINT "BINAIRE":GOTO 400
330  IF T=2 THEN PRINT "FPT ARRAY":GOTO 400
340  IF T=3 THEN PRINT "SRC DNA <or new SPL source>":GOTO 400
350  IF T=#12 THEN PRINT "INTEGER ARRAY":GOTO 400
360  IF T=#22 THEN PRINT "ALPHANUM.ARRAY":GOTO 400
370  IF T=#F6 THEN PRINT "DBS <or SPL source >":GOTO 400
380  IF T=#F4 THEN PRINT "RND <or PASCAL source>":GOTO 400

```

LIST - PAGE 65

```

390 PRINT
400 IF T IAND 2=2 OR T IAND #12=#12 OR T IAND #22=#22 THEN 430
410 P=I+18:GOSUB 740
420 PRINT "START ADDRESS          : #";HEX$(AD)
430 P=I+20:GOSUB 740
440 PRINT "FILE LENGTH            : #";HEX$(AD)
450 P=I+22:GOSUB 740
460 IF T=0 THEN PRINT "LENGTH OF TEXTBUFFER      : #";HEX$(AD)
470 IF T=1 AND AD=0 THEN PRINT "NO EXECUTION ADDRESS"
480 IF T=1 AND AD<>0 THEN PRINT "EXECUTION ADDRESS      : #";HEX$(AD)
490 P=I+24:LO=PEEK(P):P=I+25:HI=PEEK(P)
500 IF LO+HI=0 THEN 530
510 CO=256*(HI-#D)+LO-#D
520 PRINT "LOCK CODE OF FILE          :";CO
530 P=I+28:BL=PEEK(P):P=I+29:BH=PEEK(P):GOSUB 820
540 PRINT "CREATED                    :";JJ#+MM#+AA#
550 P=I+30:BL=PEEK(P):P=I+31:BH=PEEK(P):GOSUB 820
560 PRINT "UPDATED                     :";JJ#+MM#+AA#
570 P=I+26:SE=PEEK(P)
580 PRINT "NUMBER OF SECTORS          :";SE
590 STCOUNT=#AD90+1:EFAM=#AF4F:C=0:TTR=0:AL=0:TR=0
600 PRINT "LOCATION ON DISK :";
610 FOR J=STCOUNT TO EFAM:C=C+1:NF=PEEK(J)
620 IF NF<>FN THEN 660
630 TR=C/5:SEC=(C MOD 5)+1:IF TR<10 THEN AL=1
640 IF TTR<TR THEN PRINT :PRINT "track :";TR;" sector (s) ";SPC(AL);"
;:TTR=TR
650 PRINT SEC;" - ";
660 AL=0:NEXT:IF TR=0 THEN PRINT ">>>ACTUALLY KILLED<<<<"
670 PRINT
680 G=GETC
690 IF G=ASC("S") THEN RETURN
700 IF G=0 THEN 730
710 G=GETC:IF G=0 THEN 710
720 IF G=#20 THEN 730:GOTO 690
730 PRINT :NEXT I
740 L=PEEK(P):H=PEEK(P+1):AD=L+256*H:RETURN
750 REM FENETRE DE SCROLLING
760 REM
770 POKE #BA,#69:POKE #8B,#BF
780 POKE #8C,#E5:POKE #8D,#B3
790 POKE #84,#D5:POKE #85,#B3
800 POKE #8E,#D5:POKE #8F,#B3
810 RETURN
820 REM DECODAGE DE LA DATE
830 REM
840 BJ=BL IAND 31:REM 5 BITS
850 BM=BH IAND 15:REM 4 BITS LOW NIBBLE
860 AA=BH IAND 240:REM BH HIGH NIBBLE
870 AL=AA SHR 4:REM LOW PART OF YEAR
880 AT=BL IAND 192:REM BITS 7&8
890 IF AT=0 THEN BA=80+AL:REM 80-95
900 IF AT=64 THEN BA=96+AL:REM 96-99
910 IF AT=128 THEN BA=AL:REM 00-15
920 IF AT=192 THEN BA=16+AL:REM 16-31
930 BBJ#=STR$(BJ):BBM#=STR$(BM):BBA#=STR$(BA)
940 BJ#=MID$(BBJ#,1,1):IF BJ>9 THEN BJ#=MID$(BBJ#,1,2)
950 BM#=MID$(BBM#,1,1):IF BM>9 THEN BM#=MID$(BBM#,1,2)
960 BA#=MID$(BBA#,1,1):IF BA>9 THEN BA#=MID$(BBA#,1,2)
970 JJ#"0"+BJ#:IF BJ>=10 THEN JJ#=BJ#
980 MM#"0"+BM#:IF BM>=10 THEN MM#=BM#
990 AA#"0"+BA#:IF BA>=10 THEN AA#=BA#
1000 RETURN
*
```

### Exemple d'utilisation de Ken-Zapper

```

FILE NUMBER          : 2
FILE NAME            : DNA
STATUS               : #2A
FILE TYPE            : BINAIRE
START ADDRESS       : #1100
FILE LENGTH         : #1F00
EXECUTION ADDRESS   : #1100
CREATED             : 190984
UPDATED             : 190984
NUMBER OF SECTORS   : 8
LOCATION ON DISK :
track : 3 sector(s) : 2 - 3 - 4 - 5 -
track : 4 sector(s) : 1 - 2 - 3 - 4 -
```

```

FILE NUMBER          : 1
FILE NAME            : AUTOEXEC.HELLO
STATUS               : #2
FILE TYPE            : BASIC
START ADDRESS       : #3EC
FILE LENGTH         : #1C
LENGTH OF TEXTBUFFER : #1B
CREATED             : 100884
UPDATED             : 100884
NUMBER OF SECTORS   : 1
LOCATION ON DISK :
track : 3 sector(s) : 1 -
```

oublie le mot de passe ensuite !), le programme donne les renseignements les plus importants concernant les fichiers : l'adresse de début d'implantation, la longueur, l'adresse d'exécution (si c'est un programme binaire), ou la longueur du tampon texte (si c'est du Basic). Ceci permettra de récupérer un fichier accidenté et de le replacer où il faut.



Mais, pour récupérer quoi que ce soit, encore faut-il savoir où chercher ! Rassurez-vous, Ken-Zapper veille et vous aide : les derniers renseignements fournis concernent le nombre de secteurs utilisés par le fichier, et leur localisation, exprimée en pistes et secteurs. Au passage, cet utilitaire montre immédiatement les fichiers dont les secteurs sont disjoints, par suite de nombreuses réécritures sur le disque. Vous saurez donc quand il est nécessaire de COMPACTer un disque, pour remettre tout cela en ordre.

Pour ce qui est de la technique de programmation, le programme se contente d'exploiter ce que dit le mode d'emploi du Ken-Dos. Il décode les 32 octets de DIRECTORY alloués à chaque fichier, à partir de l'adresse AF50H, et la File Allocation Map (à partir de AD50H). Il exploite aussi ce que le mode d'emploi ne dit pas ! Pour la curiosité, regardez donc dans la liste comment le LOCK CODE est constitué (lignes 490 à 520), et comment la DATE est composée (lignes 820 à 990). Pas évident, n'est-ce pas ?

Au fait, savez-vous que vers l'an 2035, les dates inscrites sur les fichiers seront fausses ? Dépêchez-vous d'utiliser votre Ken-Dos !

Alain MARIATTE