

INHOUDSOPGAVE:

- 1.....Inhoudsopgave
- 2.....Van de redactie
- 3.....Gangenstelsel op floppy
- 3.....Lezers schrijven
- 4.....Hoe schrijf ik een ADVENTURE
- 5.....Hoe schrijf ik een ADVENTURE
- 5.....Vragenrubriek
- 6.....Speel en win
- 7.....Oplossing van de maand
- 8.....Oplossing van de maand
- 8.....Puzzel
- 9.....CFORTH voor beginners
- 10.....Datums
- 11.....Datums
- 12.....Datums
- 13.....Datums
- 14.....Vermenigvuldiging
- 15.....BASIC verplaatsen II
- 16.....BASIC verplaatsen II
- 17.....SHAPES in BASIC
- 18.....SHAPES in BASIC

INZENDEN COPY
AAN:Redactie COMX-NEWS
Op Gen Hoes 113
6442 PR Brunssum

DE REDACTIE KAN NIET AANSPRAKELIJK GESTELD WORDEN VOOR DE
OPGENOMEN INZENDINGEN VAN LEZERS.

NIETS UIT DEZE UITGAVE MAG WORDEN VERMENIGVULDIGD OP ENIGERLEI
WIJZE, ANDERS DAN TEN BEHOEVE VAN PERSOONLIJK GEBRUIK, ZONDER DE
VOORAFGAANDE SCHRIFTELIJKE TOESTEMMING VAN HET BESTUUR VAN

COMX GEBRUIKERS ORGANISATIE NEDERLAND.

C.G.O.N. ; Melkersweide 1. 3437 DL Nieuwegein
Giro 16.71.947, Bank 33.10.77.787

VAN DE REDACTIE

Vorige week vroeg mijn buurman mij advies over computers.
Ik heb hem een Commodore geadviseerd.
Ook dit keer ligt een erg gevarieerd COMX NEWS voor U.
Ik begin mij na drie jaar COMX steeds meer te realiseren, dat we
over een bijzondere computer beschikken.
We mogen wel stellen, dat onze computer niet snel stuk te krij-
gen is. Mijn buurman's Commodore al na drie uur.
Als ik met mensen over de COMX spraak, dan vertel ik hen over
zijn makkelijke BASIC, de eenvoudige machinetaal, de
degelijkheid, maar vooral over de 'kick' die je kunt krijgen
met het ZELF DOEN.
De COMX is geen spel-computer; geen Commodore.
Mijn buurman zocht een kant en klare spelcomputer. Vandaar mijn
advies. Mensen met een COMX doen m.i. in de regel meer met hun
computer.
Dat we een bijzondere computer hebben, bewijst weer de inhoud van
dit blad. Zonder de bijzondere inzet van de auteurs waren zeker
enige pagina's leeg gebleven of gevuld met tekeningen.
Een rubriek zult U echter gaan missen.
De gebroeders van Tongeren zijn met hun machinetaal-cursus
gestopt.
Vanaf deze plaats een groot woord van dank aan Marcel en Frank!
Een vol nummer wil niet zeggen dat we copy genoeg hebben.
We blijven blij met elke inzending, op papier, op cassette of
diskette.
Vooral de ervaringen van beginnende computeraars zijn zeer wel-
kom!!!
Het blijkt nog eens, dat geplaatste stukjes onwaarheden bevat-
ten. Althans in de ogen van (sommige) lezers. Voor U een reden
om in de pen te klimmen, en Uw variant op een probleem kenbaar
te maken.
In COMX NEWS nr. 29 (verschijnt in september) willen we een
stukje schrijven over de F&M BASIC en de F&M Eprom.
Dit gaat niet zonder opmerkingen van Uw kant. Laat het ons weten!

Verderop in dit nummer staat een woord-puzzel, gemaakt door
Oscar Nooy.
Er bestaat overigens ook een computerprogramma wat zo'n puzzel
(tot maximaal 15 woorden) zelf ontwerpt en uitprint.
Een listing hiervan kunt U krijgen door een aan Uzelf geadres-
seerde en gefrankeerde!!! enveloppe in een enveloppe naar het
redactieadres te sturen. Doe er dan meteen wat copy bij!
Inzenders van diskette of cassette (met copy!) krijgen deze
retour met het puzzelprogramma erop.
Het blijkt dat het nog onduidelijk is wat de redactie doet.
We zijn alleen bezig met het maken van COMX NEWS.
Het drukken, verzenden, de administratie en de listingservice
gebeurt vanuit het adres op pagina 1 onderaan.
Tot slot wil ik alle lezers in het algemeen en de vaste en losse
auteurs van copy in het bijzonder, een prettige zomervakantie
toewensen.
Ik ga met mijn COMX op vakantie.
Mijn brievenbus niet.

Leo Degenkamp

GANGENSTELSEL OP FLOPPY DISK

Naar aanleiding van de vragen in COMX NEWS 25 & 26, heb ik getracht om het spel GANGENSTELSEL draaiende te krijgen op de floppy disk. Dit ging in het begin niet helemaal zonder problemen, daar het programma met de data veel te lang was voor gebruik op de floppy drive. Echter met veel proberen ben ik er toch uitgekomen. Het resultaat heb ik hieronder in het kort samengevat.

- PLOAD het programma GANGENSTELSEL (C.60)
- Verwijder alle REM-regels. (Een goed programma moet ook zonder de REM-regels nog goed kunnen werken !!)
- Verwijder nu de volgende regels: 110, 120, 4920, 4930, 4940, 4950, 4960, 4970, 4980, 4990 en 5000.
- Typ nu het volgende in:


```
10 DIM S(12,12),I(12,12),J(12,12),P(2),U(5),W(6)
15 PRINT "INSERT DATA-DISK AND PRESS SPACEBAR"
20 IF KEY<>32 GOTO 20
25 DOS OPEN,*1,"GANGENST. DATA"
30 DOS READ,*1,S(12,12),I(12,12),J(12,12)
35 DOS READ,*1,P(2),U(5),W(6)
40 DOS READ,*1,A$(23),D$(23)
45 DOS READ,*1,C$(2),S$(6)
50 DOS CLOSE,*1
60 A$(0)=C$(1): D$(0)=C$(2) : REM DIT IS NIET FOUT !!
4920 REM
```
- Save het programma op floppy, bijv: DOS SAVE,"GANGENSTELSEL"
- DLOAD nu de data van de cassette.
- Typ in:


```
S$(1)=""
C$(1)=A$(0)
C$(2)=D$(0)
DOS CREATE,*1,"GANGENST. DATA"
DOS CLOSE,*1
DOS OPEN,*1,"GANGENST. DATA"
DOS WRITE,*1,S(12,12),I(12,12),J(12,12),P(2),U(5),W(6)
DOS WRITE,*1,A$(23),D$(23),C$(2),S$(6)
DOS CLOSE,*1
```
- Het spel GANGENSTELSEL is nu gereed voor gebruik !!

Door: Erwin Smit

LEZERS SCHRIJVEN

Bas de Paauw tipt de gebruikers van het spel 'SMURFEN'. Het komt wel eens voor dat het programma terugspringt naar het gedeelte waarin de gebruiker gevraagd wordt of hij de instructies wil zien. Als je al een keer een smurf verloren hebt, kun je het beste op deze vraag de 'N' intoetsen, gevolgd door <CR>. De vraag en de instructies zijn door de veranderde SHAPES niet meer leesbaar.

HOE SCHRIJF IK EEN ADVENTURE (deel 3)

In dit deel zal ik de COMMANDO-vertaler uitleggen. Deze vertaler kan maximaal twee woorden per keer herkennen. De commando's mogen ook afgekort worden. Het nadeel is echter, dat als U een B intikt, de vertaler dit als het eerste woord herkent als hij in de M\$ zoekt, en welk met een B begint. Dus boven en beneden kunt U het beste afkorten met BO en BE. Dan volgt nu het programma gedeelte van de commando-vertaler.

```
1 REM COMMANDO-VERTALER
2 CPOS (17,0): PRINT " WAT DOET U ?"
3 INPUT C$
4 IF C$="" GOTO 3
5 N = LEN (C$) : O = N + 1
6 FOR I = 1 TO N
7 IF ASC (MID$(C$,I,1)) = 32 THEN O = I
8 NEXT I
9 P = 0 : Q = 0
10 W$ = MID$ (C$,1,O+1) : R = LEN (W$)
11 FOR I = 1 TO B
12 IF W$ = MID$ (M$(I),1,R) EXIT 14
13 P = I : I = B : NEXT I
14 IF P = 0 GOTO 16
15 PRINT " IK SNAP NIET WAT U BEDOELT " : GOTO 2
16 IF O = N + 1 RETURN
17 W$ = MID$ (C$,C+1,N) : R = LEN (W$)
18 FOR I = (B+1) TO E
19 IF W$ = MID$ (M$(I),1,R) GOTO 21
20 PRINT " IK SNAP HET NIET"
21 Q = I : I = E : NEXT I
22 RETURN
```

Aan C\$ wordt de ingetoetste waarde toegekend, en N bevat het aantal karakters van die string.
O = N + 1.

In regel 7 wordt de plaats van de spatie bepaald. Voor het eerste woord worden alleen de eerste B termen "afgescand", want dit zijn de mogelijke commando's. Voor het vinden van het tweede woord worden alle overige termen bekeken.

De tabel op de volgende pagina toont een voorbeeld van een aantal termen van M\$ en de waarde die aan P en Q worden toegekend. Tevens zien we voor alle termen P met welke waarde van de ARRAY L deze termen overeenstemmen.

Het gebruik van ARRAY L komt nog aan de orde.
(Bekijk alvorens verder te lezen eerst de tabel)

In M\$ worden eerst alle commando's opgenoemd, dan de voorwerpen, en dan de personen.

Met dat gegeven in ons achterhoofd kunnen we snel uitzoeken waar een gegeven waarde van Q betrekking op heeft.

Een voorbeeld:

Bij het commando PAK TOUW krijgt P de waarde 4 en Q de waarde 7. Nu is het aantal commando's (B) gelijk aan 5 en hier volgt weer uit dat het voorwerp dat Q aanduidt voorwerp nummer 7-B is. Dus voorwerp 7-B=7-5=2 !!

Inderdaad, het tweede voorwerp dat wordt opgesomd is het touw. Zo kunnen we ook bepalen om welke persoon het gaat, met Q-B-C. We zullen straks bij de acties en tegenacties zien, dat we van bovenstaande constructies inderdaad gebruik zullen maken.

Tabel 3-1:

Nummer	M\$	P	Q	L(P)
1	NOORD	1		1
2	OOST	2		1
3	ZUID	3		1
4	PAK	4		2
5	LEG	5		3
6	ZWAARD		6	
7	TOUW		7	
8	SLANG		8	
9	HEKS		9	

Reacties en vragen over deze rubriek?

U kunt rechtstreeks schrijven naar:

Cor de Weerd,
Moerweg 680,
2531 BR DEN HAAG.

VRAGENRUBRIEK

Vr.32: Waarom staan bij programma's als GANGENSTELSEL en STRIP-POKER de DATA apart?

Aw.32: DATA apart houden heeft volgens ons alleen dan zin, als ze ook meerdere keren gebruikt (ververst) worden; om bv. bij een spel een nieuw speelveld in te laden, en dan alleen nog, als de geheugenruimte overschreven dreigt te worden.

O.i. hadden genoemde programma's net zo goed de DATA in het programma kunnen opnemen.

Of heeft iemand hier een andere mening over?

Vr.33: Wie is in staat om het programma 'OTHELLO' zodanig te verbeteren, zodat de verbeteringen van Eric van Dorp en Paul Pieters (COMX NEWS 6 en 21) samen gebruikt kunnen worden?

Aw.33: Reacties graag naar de redactie.

SPEEL EN WIN

Deze maand het spel: LASER GUN. Naar mijn mening is dit het beste BASIC (spel)programma dat geschreven is voor de COMX. Het programma zit prima in elkaar, zodat je haast zou vergeten dat het hier om een basic programma gaat. Dit is duidelijk te merken aan de snelheid van het programma, deze is zelfs zonder RUN+ nog groot. Het spel werd gemaakt door Arjen de Ronde.

KORTE BESCHRIJVING

Stelt u zich eens voor dat uw planeet, ergens in de verre uit-
hoeken van het heelal, wordt belaagd door ruimte monsters. Gelukkig hebben de wetenschappers op uw planeet een superdeluxe LASER GUN ontwikkeld, waarmee u uw belagers tot stof kunt omtoveren. Maar omdat de belagers op 9 plekken tegelijk aanvallen, zal het voor u toch nog moeilijk zijn, om ze allemaal te vernietigen. Als een rij met de vijand, uw planeet heeft bereikt, hebt u verloren.

Schieten kunt u door een cijfer van 1 t/m 9 in te typen. De LASER zal dan onder het door de cijfer aangegeven plaats gaan staan en aldaar vuren. Hoe hoger de vijand in de rij zit, des te meer punten krijgt u. Let er in de tussentijd wel op dat een andere rij niet uw planeet bereikt.

SPEELTIPS

De belangrijkste tip, waar eigenlijk het hele spel om draait, is: zorg ervoor dat tenminste een rij vrij blijft, zodat u het moederschap kunt vernietigen. Als u deze namelijk neerschiet, dan zal de vijand de helft van zijn troepen terugtrekken. Afwisselend worden de even en oneven rijen teruggetrokken. Zolang u steeds het moederschap neerschiet is er niets aan de hand. Het moederschap kan echter uw vrije schietgaten afsluiten. Het is daarom van belang om meerdere gaten vrij te houden. Wil dit echter niet meer lukken, dan kunt u alleen nog maar verdedigen en kunt u er van op aan dat het niet meer lang duurt voordat de vijand uw planeet overneemt...

Mocht u niet meer tegen vijand zijn opgewassen dan kunt u nog altijd de SUICIDE (zelfmoord) knop indrukken. Op het toetsenbord is dit toets S. Erg eervol ten onder gaat u dan niet...

AANPASSINGEN IN HET PROGRAMMA

Aan dit programma valt weinig te verbeteren en kan als een prima voorbeeld dienen voor andere basic programmeurs, mijzelf inbegrepen.

TOP ??

Ikzelf haalde bij dit spel een score van 12580. Maar volgens mij moet hier nog veel meer te behalen zijn. Wie heeft er een hogere score ??

Het spel AUTORACE komt de volgende keer aan bod.

Door: Erwin Smit

OPLOSSING VAN DE MAAND FEBRUARI

HET REISPROBLEEM

Hoe reis ik in een zo kort mogelijke tijd door een aantal steden? Dat is iets wat veel mensen zich afvragen. Soms leidt dit zelfs tot een recordpoging in tv-programma's, zoals 'de eerste de beste'. Het probleem staat bekend als het handelsreizigersprobleem. In elk zichzelf respecterend wetenschappelijk blad wordt dit probleem wel eens behandeld. Zo heb ik de oplossing voor ons probleem gevonden in het blad Aarde & Kosmos (no 4/1985, p 226-228). De hieronder staande listing is een speciaal door mij aangepaste versie voor de COMX computers.

```
10 CPOS(0,0):CLS:A=5
20 DIM A(A,A),V(A)
30 FOR I=1 TO A: READ V(I): NEXT
40 DATA 1,2,3,4,5
50 FOR I=1 TO A: FOR J=1 TO I
60 READ A(I,J): A(J,I)=A(I,J)
70 NEXT: NEXT
80 DATA 0,45,0,40,75,0,55,75,65,0
90 DATA 20,40,50,35,0
100 FOR I=1 TO A: READ S$(I): NEXT
110 DATA "A=BRUNSSUM","B","C","D","E"
120 P=1: S1=0
130 PRINT "POGING NR.":P: PRINT
140 S=0: FOR I=1 TO A
150 IF I<A S=S+A(V(I),V(I+1)):GOTO 170
160 S=S+A(V(A),V(1))
170 PRINT S$(V(I))
180 NEXT
190 E=1: PRINT: PRINT "BIJ BOVENSTAANDE POGING ";P;" WAS DE"
200 PRINT "TOTALE REISTIJD: ";S;" MINUTEN": PRINT: WAIT(100)
210 PRINT "-----"
220 IF S1<S IF S1>0 GOTO 260
230 P=P+1: K1=INT(RND(A)+1)
240 L1=INT(RND(A)+1)
250 IF L1=K1 GOTO 240
260 GOSUB 290
270 S1=S: IF S1<T T=S1
280 GOTO 130
290 C=V(K1)
300 V(K1)=V(L1)
310 V(L1)=C
320 RETURN
```

Het programma gaat steeds uit van 'n poging met de kortste reistijd (=uitgangsrij). Daarna worden er RANDOM 2 steden verwisseld. Als nu de reistijd korter is, dan wordt dit de nieuwe uitgangsrij. Is dit niet het geval, dan wordt er een nieuwe poging ondernomen vanuit de oorspronkelijke rij! Uiteraard zou je ook alle mogelijkheden kunnen bekijken. Er is echter met opzet voor deze opbouw gekozen. Bij 5 steden is het

namelijk heel goed toe te doen om alle mogelijkheden te bekijken. Maar veronderstel eens dat je 15 steden hebt ingevoerd (door de DATA regels en de variabele A aan te passen). Als nu elke seconde (!) 1 mogelijkheid zou kunnen worden bekeken, dan zou het nog 1373 jaar duren, voordat de oplossing zou worden gevonden!!! Bij de COMX is zelfs dit nog niet haalbaar! Van daar dat er gekozen is voor een willekeurige methode. Het programma moet je onderbreken met ESCape, als je denkt dat de computer de kortste afstand heeft gevonden. Bij 15 plaatsen kan dit wel UREN duren.

Om meer te weten te komen over dit onderwerp, wordt verwezen naar de eerder genoemde literatuur.

Overigens stond in de opgave (COMX NEWS 24) een fout. Normaal gesproken is de tijd van E naar D hetzelfde als van D naar E. In de opgave was dit echter niet het geval (E->D 35, D->E 40). In het programma heb ik gekozen voor 35. Zou dit de reden zijn dat niemand de oplossing instuurde?!

De oplossing is waarschijnlijk: A->B->E->D->C->A met een tijd van 225 minuten!

Door: Erwin Smit

PUZZEL

```
A B Y S Y C P P L E F Y B O N A R B P Z
B W R T E N U M S L O G C E G W C A D M
O M R I T E A E P A H S K H I L M V E Q
F T A C C R V U N E V C T W F I L E C O
D O P F X A N T I R A E D G T O M N T A
E G J F S I L L Y A L W A I T U O D A T
A T A D R O P L I S T O F U L L P O R P
R R A R O E F E A T E G R O V W E R T L
E S R E B M U N E R R G V E R D A C E P
D E F T R D A G G A A T U A I M E F E E
E R I R E G H E T K S E O L D A A T G E
F N O O G I O A T D C I Z L E O B T E E
U P L E T F N E L A T E R E O U S R H K
M N A P D H E P E W R E A D P N N I B M
W I D H P E E U U M A Z E C A V E S A R
I E D E K O N N L T A M O E D E R P T Z
G O S U P G U T I N R U T E R P L P L R
T H H T S R P E C H Q N L A H O N O T A
H I T I T B Y A H O N F O P L O T O G D
T T A P L L E K A Z K A Q S E U K K H E
```

Wat moet je doen ?

In deze letterbrei zijn 25 COMX-BASIC commando's verborgen. Deze commando's kunnen horizontaal, vertikaal, diagonaal, van links naar rechts en van rechts naar links staan. Let op!! Er staan ook andere woorden in, alleen de COMX-BASIC-commando's zijn gevraagd. DIM is al gegeven, zodat je er nog maar 24 hoeft te zoeken. Veel plezier!

CFORTH VOOR BEGINNERS

Er is kritiek dat COMX NEWS weinig te bieden zou hebben voor de prille beginner in bijvoorbeeld FORTH of machinetaal. Wanneer dit waar is, dan zijn we daarin niet verschillend van veel andere computerbladen. De HCC Nieuwsbrief, het maandblad van de Hobby Computer Club, behoeft je ook niet meer voor je plezier in huis te hebben. Het is nu eenmaal onvermijdelijk dat beginners langzamerhand, sommigen zelfs zeer snel, tot gevorderden worden. Daarmee verandert de interessesfeer; men wil moeilijker dingen doen dan alleen maar spelletjes. Met als gevolg dat de argeloze binnenkomer gekonfronteerd wordt met een computer-ritueel dat alleen nog door de hogepriesters kan worden begrepen. Maar, ben je wel eens bij een tennisclub geweest? Worden beginners daar zoveel schappelijker behandeld?

Het is misschien goed om, zo vlak voor de vakantie, de zinnen even te verzetten. Daarom heb ik besloten om mij het lot van de absolute beginner voor EEN keer aan te trekken. Want COMX NEWS moet ook voor mezelf leuk blijven.

Het punt waar in de computerbiz alles om draait is: waar haal je de nodige informatie vandaan. Ironisch genoeg is, MET NAME in de informatica, het vinden van de juiste informatie meestal een uiterst moeizame aangelegenheid. Waarom dat zo is, daar kan ik uren over doorgaan. Twee meter handleiding voor EEN regel code is niet ongewoon. Niet bepaald een (werk)situatie om vrede mee te hebben.

Waar haal je de juiste informatie vandaan. Het antwoord is niet bepaald modern: in de eerste plaats uit een GOED boek. Daarbij geldt de regel: beter EEN boek dan een hele stapel. Tussen twee haakjes: boeken zijn de oudste (en de beste) ROMs aller tijden. Helaas is het meestal nodig om Engels te kennen, want het meeste op computergebied is van Amerikaans origine. Hieronder volgt dan wat ik zelf heb stukgelezen. Anderen mogen wat mij betreft de lijst aanvullen, maar denk aan de twee meter handleiding!

1. Tom Swan; Programmer's Guide to the 1802; Hayden Book Company U.S. (1981). In Nederland verkrijgbaar bij: Nanton Press, f 26,-; of bij de 1802 gebruikersgroep (adres vast wel bij de redactie). Dit boek is een echte aanrader wanneer je machinetaal wilt toepassen in FORTH of BASIC.

2. Alan Winfield; The Complete Forth; Sigma Technical Press U.K. In het Nederlands vertaald onder de titel "Flitsend FORTH" (?). Uit dit boek heb ik FORTH geleerd. En naar aanleiding daarvan (niet omgekeerd!) kocht ik destijds een COMX-35.

3. Beslist niet te versmaden zijn de boeken over machinetaal, de technische gegevens van de COMX, en het inmiddels vertaalde "Introduction to C Forth programming" van Ken Tracton.

Maar dan praten we over literatuur die eigenlijk toch niet meer voor beginners is. Overigens zou ik het CFORTH boek WEL kopen, en wat je niet snapt ... gewoon overslaan !!

Han de Bruijn
Kempenaar 43
2991 PH Barendrecht

DATUMS VERWERKEN

In veel programma's worden op de een of andere manier datums (dit keer geen DATA. -red.) verwerkt.

Het probleem voor velen is nu: "hoe verwerk je nu op een correcte manier datums?".

In dit artikel gaan we hier uitgebreid op in.

Ik beweer niet dat het DE manier is om datums te verwerken, noch dat het de snelste of enige methode is.

Kortom het is een manier waarop we DATUMS kunnen verwerken, niet meer, niet minder.

Op de een of andere manier moeten we de computer een DATUM aanbieden. Dit kan op verschillende manieren.

Persoonlijk doe ik het altijd op de meest voorkomende manier.

En die vorm noem ik dan een 'NORM' datum.

In de variabele D komt het dagnummer.

In de variabele M komt het maandnummer.

In de variabele J komt het jaartal.

Toch vraag ik het nooit aan op die manier, nee, in mijn programma's vraag ik ALLE invoer aan in K\$.

Onverschillig wat voor invoer het ook is, het komt altijd eerst in K\$. Waarom?

Wel, na de invoer door de gebruiker moeten we in veel gevallen een controle op de invoer doen plaats vinden.

En omdat alles in K\$ staat kunnen we met een paar standaardroutines volstaan.

Laten we eens zien hoe dat in de praktijk gaat.

Op het scherm zien we het volgende:

VOER DATUM IN <DDMMJJ>:

Na de invoer van de gebruiker staat het ingevoerde in K\$. Door de aanduiding <DDMMJJ> is het de meeste mensen wel duidelijk dat verwacht wordt dat de eerste twee cijfers het dagnummer is, de volgende twee het maandnummer en de laatste twee het jaarnummer. Maar wat men ook intikt, na <CR> volgt nooit een 'ERR CODE' bericht. Nu gaan we controleren of het ingetoetste een datum kan zijn. De eerste controle is de lengte. We weten dat we zes cijfers vragen, dus is de lengte anders dan 6 dan hebben we zowiezo te maken met een invoer fout. In BASIC ziet dat er zo uit:

```
... IF LEN (K$) <> 6 THEN ....
```

Komt K\$ door die controle dan gaan we verder, anders stellen we de vraag "VOER DATUM IN....." opnieuw.

De volgende stap is logisch; we weten dat we naar zes cijfers vroegen.

We weten ook dat we zes tekens binnen gekregen hebben.

Nu gaan we dus kijken of het ook inderdaad cijfers zijn.

Een manier waarop we kunnen controleren of er alleen maar cijfers zijn is:

K = FVAL (K\$) : X\$ = STR\$ (K)

Wanneer X\$ en K\$ nu aan elkaar gelijk zijn, dan zijn er alleen cijfers tijdens de invoer gebruikt.
Maar let op!
Er kunnen twee dingen nog mis zijn, het getal kan negatief zijn of het getal bevat een decimale punt.
Kontroleren dus.

IF K = INT (K) IF K > 0

Dat is de BASIC regel.
We zijn nu weer een grote stap verder.
Maar nog weten we niet zeker of het ingevoerde nu wel een goede datum is.
Immers de dagen mogen niet meer zijn dan 31, de maanden niet meer dan 12 en DAN NOG is er kans op fouten.
Wat we nu gaan doen is het volgende:
We maken van de datum een UNIEK getal
Daarna maken we van dit UNIEKE getal weer een datum.

Wanneer de laatste datum gelijk is aan de ingevoerde, dan is de datum correct.
D.w.z: het is een datum, of het dan inderdaad de datum van vandaag is..?

Voor ik nu de formule geef eerst een korte uitleg van het principe.

3 + 4 = 7, dus 7 - 4 = 3

Wanneer ik nu uitrekenen dat 3 + 4 = 6 en de computer dit bovenstaande laat kontroleren, dan zijn er verschillen in uitkomst, dus is mijn antwoord fout!!

We gebruiken de volgende variabelen:

* J = (J)aartal (alleen laatste twee cijfers)
* M = (M)aand nummer (jan = 1, feb. = 2 etc..)
* D = (D)ag nummer (1,2,3.....31)
* G = (G)etal (dat we gaan uitrekenen)

De formule om van D, M en J een UNIEK getal te maken:

G = INT (J*365.25) + INT (M*300.65) + D

Opzich zelf nog wel vrij simpel, maar om van G weer een datum te maken is iets moeilijker.
De formule:

J = INT ((G - 122.1) / 365.25)
G = INT (365.25 * J)
M = INT ((G - 0.1) / 30.6)
D = G - INT (30.6 * M)

Indien M > (groter dan) 13 : J = J + 1 : M = M - 13
Anders geldt: M = M - 1
De nieuwe datum is nu: G = D + M + J * 1000

Ik geef toe dat het moeilijk klinkt, maar het werkt perfect.
Wanneer de op deze manier verkregen datum anders is dan de datum in K\$, dan is K\$ fout!!

Voor sommigen zal de formule niet helemaal duidelijk zijn.
Laat die mensen dan bedenken dat het wel werkt, en dat is het belangrijkste.
Anderen zullen denken waarom zo ingewikkeld.
Wel, dat heeft een andere oorzaak.
Met 'G' kunnen we namelijk rekenen.
Probeer het volgende maar eens:

Vandaag is het 1 juni 1987; welke datum is het over 87 dagen?

Handmatig een lastige klus.
Voor ons fenomeen G niet!
We rekenen 'G' uit aan de hand van de ingevoerde datum (010687);
we tellen daar 87 bij op en via de formule laten we 'G' weer veranderen in een datum, en zie daar ons antwoord!

Ook kunnen we de dag van de week nu vrij simpel bepalen.
Van de oudste datum die ons bekend is, waarvan we de DAG weten berekenen we 'G'.
Doe dit bij voorkeur op een zondag.
In ons programma benoemen we:
D\$(1)="ZONDAG" : D\$(2)="MAANDAG" etc.

Stel dat we de oudst bekende zondag G1 noemen dan berekenen we de dag als volgt:

K = G - G1
F = MOD (K , 7) : F = F + 1

De dag is dan D\$(F) !!!!!!!!!!!

Kortom, datums? We kunnen er nu mee doen wat we willen!
Of niet?
Hoe zit het met Pasen, Pinksteren etc.?
Wel, ook hier is een oplossing voor.
Feestdagen zijn er in twee soorten. nl.: Die met een vaste datum zoals Kerstmis, Nieuwjaarsdag, of dagen met telkens een andere datum zoals Pasen, Pinksteren, Carnaval, Hemelvaart etc.
Die laatste categorie heeft als eigenschap dat de onderlinge afstand in dagen gerekend gelijk blijft.
Wanneer we dus een datum hebben, dan is de rest uitrekenen via de methode zoals die hiervoor is uitgelegd kinderspel.

Zelf een formule vinden voor het berekenen van de Paasdatum is niet simpel, maar dat hoeft ook niet.
Dat heeft aan het begin van de vorige eeuw de wiskundige Karl Friedrich Gauss al voor ons gedaan.

De formule is correct vanaf 1583 en werkt met een vier (4) cijferig jaartal.

Bereken de volgende waarden:

K is het gehele deel van jaartal / 100
 P is het gehele deel van (8 * K + 13) / 25
 G is het gehele deel van K / 4
 M is de rest van de deling (15 + K - P - G) door 30
 N is de rest van de deling van (4 + K - G) door 7
 A is de rest van de deling van het jaartal door 19
 B is de rest van de deling van jaartal door 4
 C is de rest van de deling van jaartal door 7
 D is de rest van de deling van (19 * A + M) door 7

1ste Paasdag valt dan op D + E + 22 maart of D + E - 9 april

Maar als D=29 en E=6 dan wordt het 19 april.

En als D=25 en E=6 en de rest van de deling van 11 * M + 11 door 30 kleiner is dan B dan is 18 april de eerste paasdag.

Wel, tot zover het verhaal over datums.

Als slot wil ik nog opmerken, dat er in dit artikel geen listings zijn opgenomen.

De reden is simpel: Voor mij en voor velen met mij is de charme van de computer het zelf proberen.

Zelf een programma maken.

Mijn bedoeling is dan ook dat U aan de hand van het bovenstaande ZELF probeert om het een en ander aan de praat te krijgen.

Achtergrond informatie heeft U nu voldoende, en waar vindt U de informatie:

Juist, in COMX NEWS.

Door: A. Mol

Naschrift van de redactie:

Als alles goed gaat, zult U in de toekomst meer van dhr. Mol in COMX NEWS kunnen lezen.

Hij heeft ons beloofd, nog enige teksten op papier (cassette) te zetten.

Na Erwin Smit, Han de Bruijn en Cor de Weerd verwelkomen we een vierde vaste rubriekschrijver in COMX NEWS (naast de redactie) Het kan zijn, dat de formule onderaan pagina 11 niet geheel correct is. Overtypfouten hebben ons dan parten gespeeld. De formule voor G zou ook deze kunnen (?) zijn:

$$G = \text{INT} (J * 365.25) + \text{INT} (M * 30.65) + D$$

Erg van belang vinden we dit niet.

Lees dan nog eens de laatste alinea van dhr. Mol zijn betoog!!

Juist, ja, zelf doen!

Hieronder voor de liefhebbers een variant op het vermenigvuldigingsprobleem uit COMX NEWS 23.

Deze oplossing werd ingestuurd door Dhr. G. de Weert uit Venray.

```

10 CPOS (0,0): CLS:H=20:V=5:Q=1:U=0:W=0
20 I=0:J=0
30 DIM F(9): DIM G(6): DIM P(6): DIM Q(9)
40 FOR N=1 TO 9:F(N)=0: CPOS (N+1,1): PRINT N;"-"
50 CPOS (N+1,3): PRINT F(N): NEXT
60 G(1)=179+I:G(2)=223+J
70 E=MOD(G(2),100):X=(G(2)-E)/100:D=MOD(E,10):Y=(E-D)/10:Z=D
80 IF Y=ZJ=J+1: GOTO 60
90 IF Y=0J=J+10: GOTO 60
100 IF Z=0J=J+1: GOTO 60
110 G(6)=G(2)*G(1): IF G(6)>99999I=I+1:J=0: GOTO 60
120 G(6)=G(2)*G(1)
130 CPOS (10,10): PRINT G(2),G(1),G(6)
140 CPOS (10,15): PRINT "*": CPOS (10,23): PRINT "="
150 G(3)=G(1)*Z: IF G(3)>999J=J+1: GOTO 60
160 G(4)=G(1)*Y: IF G(4)>999J=J+1: GOTO 60
170 G(5)=G(1)*X: IF G(5)>999J=J+1: GOTO 60
180 CPOS (12,10): PRINT G(3),G(4),G(5)
190 CPOS (12,3): PRINT Q;" "
200 FOR K=6 TO 1 STEP -1:F=G(K)
210 IF K=6E=F:T=0: GOTO 240
220 E=100*F:T=T-2
230 CPOS (14,3): PRINT T;" "
240 D=MOD(E,10000):P(1)=(E-D)/10000
250 E=MOD(D,1000):P(2)=(D-E)/1000
260 D=MOD(E,100):P(3)=(E-D)/100
270 E=MOD(D,10):P(4)=(D-E)/10
280 P(5)=E
290 FOR L=1 TO 5:U=P(L)
300 IF U=0T=T+1: CPOS (14,3): PRINT T;" "
310 IF T>2 EXIT 380
320 FOR N=1 TO 9:Q(N)=U
330 IF U=NF(N)=F(N)+1: CPOS (N+1,3): PRINT F(N)
340 IF F(N)>2 EXIT 380
350 NEXT N
360 NEXT L
370 NEXT K
380 FOR N=1 TO 9
390 IF F(N)=2Q=Q+1: CPOS (12,3): PRINT Q
400 F(N)=0
410 NEXT
420 IF Q=9 GOTO 460
430 Q=0
440 FOR N=1 TO 9:F(N)=0: CPOS (N+1,3): PRINT F(N);" " : NEXT
450 J=J+1: CPOS (14,2): PRINT " " : GOTO 60
460 CPOS (0,0): CLS:V=5:H=20
470 CPOS (V,H): PRINT G(1)
480 CPOS (V+1,H): PRINT G(2)
490 CPOS (V+2,H): PRINT "----";" " ; "*"
500 CPOS (V+3,H): PRINT G(3)
510 CPOS (V+4,H-1): PRINT G(4)
520 CPOS (V+5,H-2): PRINT G(5)
530 CPOS (V+6,H-2): PRINT "-----"
540 CPOS (V+7,H-2): PRINT G(6)

```


BASIC VERPLAATSEN II

In COMX NEWS nr. 24 hebben F&M v.Tongeren beschreven hoe we een "ruimte" vrij kunnen maken in het geheugen op een plaats beneden een bestaand basic-programma, zodanig dat we alsnog een stuk machinetaal kunnen toevoegen.
De beschreven methode (veruit de gemakkelijkste) geeft echter problemen als we te maken hebben met een lang (groter dan 19.4 K) basic-programma.
De reden, dat de eerder beschreven methode dan tot mislukken is gedoemd zit simpelweg in het feit, dat het (lange) basicprogramma het monitor programma overschrijft (dit start nl. op @9000).

Als voorbeeld zullen we eens het spel-programma YAHTZEE bekijken. De versie die ik bezit eindigt op adres @AD05. (uit te lezen met de opdracht EOD.)
Ik heb het hoofdscherm van dit programma in machinetaal gezet, hetgeen het spelen van het spel een stuk sneller maakt en daarmee aangenamer.
Ruimte maken voor machinetaal m.b.v. de handige F&M MONITOR was niet mogelijk omdat de programma's in elkaars geheugen gebied terecht komen.
In zulk geval (en niet alleen dan) kunnen we gebruik maken van de de programma's die zich in de PRINTER INTERFACE kaart bevinden.

We zullen de uit te voeren handelingen eens op de voet volgen:

Laten we samen besluiten om het YAHTZEE-programma 6 blokken (van 256 bytes = @0600) omhoog te verplaatsen.

OPMERKING:

- 1) Hebt U een F&M EPROM in de expansiebox, dan dient u deze uit te schakelen met POKE (@43F8,1)
- 2) Indien de printer interface kaart via de expansiebox is aangesloten dient U de kaart eerst in te schakelen met CARD Sn (n = slotnr.)

We gaan eerst een blik werpen op de bekende programma pointers.

```
Type in : CALL (@C109) Hiermee roepen we een geheugendump aan.
Resultaat: START:
Type in : 4281
Resultaat: END:
Type in : 4299
Resultaat: 4281 44 0C AC FC 03 2A D0 00
          4289 B9 D4 14 09 D5 D4 14 06
          4291 D5 AD 05 AD 05 FF FF 00
          4299 AD 05 0C 00 00 44 D4 2C
De gezochte pointers zijn onderstreept.
```

```
Type in : CALL (@C10C,@ADFF,@B3FF)
Op adres @C10C roepenwe het programma aan waarmee we
het Basic programma gaan verplaatsen.
```

@ADFF is het einde van het blok waarin zich EOD bevindt
@B2FF is @ADFF verhoogd met het te verplaatsen gebied.
Dus in ons geval @ADFF + @0600

```
Resultaat: SOURCE = ADFF DEST = B3FF COPY (Y/N) ?
Type in : Y
Resultaat: SOURCE = ACFF DEST = B2FF COPY (Y/N) ?
Type in : Y totdat:
          SOURCE = 43FF DEST = 49FF COPY (Y/N) ?
Type in : N
Resultaat: READY
```

Het programma is nu in zijn geheel @0600 Bytes omhoog gezet in het geheugen.

Zoals we weten moeten nu nog de pointers worden aangepast aan de nieuwe positie van het programma.

Ook hiervoor vinden we een handig programma in de PRINTERCARD :

```
Type in : CALL (@C10F)
Resultaat: MODIFY ADDRESS:
Type in : 4281
Resultaat: 4281 44 --> type in 4A (is #44 + #6)
          4282 0C --> type spatiebalk
          4283 AC --> type in B2 (is #AC + #6)
          4284 FC --> type nu spatiebalk totdat:
          4292 AD --> type in B3 (is #AD + #6)
          4293 --> type spatiebalk
          4294 AD --> type in B3 (is #AD + #6)
          4295 --> type nu spatiebalk totdat:
          4299 AD --> type in B3 (is #AD + #6)
          429A --> type <ESC> en <CR>
          ERR CODE 0
          READY
```

Zo, het programma is nu gereed om in het geheugengebied van @4400 tot @49FF machinetaal toe te voegen.

SAVE echter eerst het verplaatste basicprogramma!!

Door: J Schiffelers

SHAPES IN BASIC

Uit berichten aan de redactie blijkt dat er veel mensen zijn, die niet weten hoe ze de ingebouwde tekens van onze COMX-computer kunnen veranderen. Ja er zijn zelfs mensen die niet eens weten dat je de ingebouwde tekens kunt veranderen. Ook zijn er mensen die wel weten hoe zij shapes kunnen maken, maar niet weten hoe ze dit in machinetaal kunnen doen. Voor al deze mensen is het nu volgende verhaal bedoeld.

HET BEGIN.

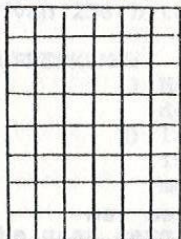
De eerste vraag is 'wat zijn shapes?'. Het antwoord op deze vraag is niet moeilijk; shapes zijn zelf gedefinieerde (=vast-gestelde) karakters, die in plaats komen van een ander karakter, bv. A of SHIFT-W of CTRL-P of CHR\$(79). Als je bv. de A vervangt door een zelfgemaakt karakter, dan krijg je als je de A intoetst, dat zelfgemaakte karakter op het scherm. Zelfs alle A's op het scherm zijn veranderd.

typ in SHAPE (65,"009EADBFBFAA95BF9E") (je hoeft dit nog niet te begrijpen.)

Alle A's zijn veranderd in een doodshoofd, en als je de A intypt, krijg je ook een doodshoofd te zien.

DE OPBOUW.

Elk karakter of shape bestaat uit 9x6 punten; 9 rijen,



6 kolommen (zie hiernaast)

Wil je een karakter veranderen, dan moet je deze eerst op (ruitjes) papier tekenen, waarbij een matrix van 9x6 hokjes het karakter voorstelt. Per rij heb je de keuze uit twee kleuren: de achtergrondkleur en een van de vier kleuren uit tabel 1.

DE SHAPE-CODE

Als een dot (=punt) de achtergrondkleur moet krijgen, dan moet je in dat hokje een 0 zetten, anders een 1. Om nu de shape code te vinden gaan we elke rij apart bekijken. We kiezen met behulp van tabel 1 een kleur uit, en zetten de code die achter de kleur staat, voor de rij waar we mee bezig zijn. Op die rij staan nu acht enen en/of nullen. Deze verdelen we in twee groepen van vier stuks. Met behulp van tabel 2 zetten we deze twee groepen van vier om in twee hexadecimale getallen. bv. 101110 met kleur groen/geel wordt 10101110 of 1010 1110. Volgens tabel 2 is 1010 gelijk aan A en 1110 gelijk aan E. De code voor deze regel is AE. Zo ga je elk van de 9 regels na, en schrijf je de shape-code op.

TABEL 1

Kleur	Code
Zwart/rood	0 0
Blauw/magenta	0 1
Groen/geel	1 0
Cyaan/Wit	1 1

TABEL 2

Binair	Hex	Binair	Hex
0 0 0 0	0	1 0 0 0	8
0 0 0 1	1	1 0 0 1	9
0 0 1 0	2	1 0 1 0	A
0 0 1 1	3	1 0 1 1	B
0 1 0 0	4	1 1 0 0	C
0 1 0 1	5	1 1 0 1	D
0 1 1 0	6	1 1 1 0	E
0 1 1 1	7	1 1 1 1	F

SHAPE IN BASIC

Nu je de shape hebt getekend en berekend, zou je hem natuurlijk graag in de computer stoppen. Dit kan je doen met het commando SHAPE. Hoe werkt dit commando nu?

Heel eenvoudig: de syntax is SHAPE (ASCII-code, "Hex-code").

Met andere woorden, je typt eerst 'SHAPE (' en dan de ASCII-code van het karakter dat je wilt veranderen. De volledige lijst staat in de engelse manuel. Vervolgens typ je een komma [,] en een quote ["] daarna de hexcodes van alle negen regels achter elkaar, en dit alles sluit je af met een quote en een haakje-sluiten ["]]. Je kunt het SHAPE-commando zowel in de direct- als in de indirect mode gebruiken. Ofwel met, of zonder een regelnummer ervoor.

Nb.: Zwart, blauw, groen en cyaan zijn de uitvoer kleuren; dwz. wat de computer op het scherm zet.

Rood, magenta, geel en wit zijn de invoer kleuren, welke te zien zijn bij het intypen van tekst.

Als je bij de ASCII-waarde 128 optelt, krijg je WEL de invoer kleuren in de uitvoer.

Je hebt dus geen 256, maar 'slechts' 128 SHAPES ter beschikking.

Je kunt een SHAPE ook weer opnieuw "SHAPEN".

Als laatste opmerking nog deze:

Let op met ASCII-codes die ook 'iets' met een printeraansturing 'doen'.

Het is dan mogelijk dat een eventueel aangesloten printer rare dingen gaat doen.

SHAPE IN MACHINETAAL

Ik verwijs hier naar de routine van F&M van TONGEREN in COMX NEWS nr. 8. In het komende nummer gaan we hier uitgebreid op in.