

J. Uges

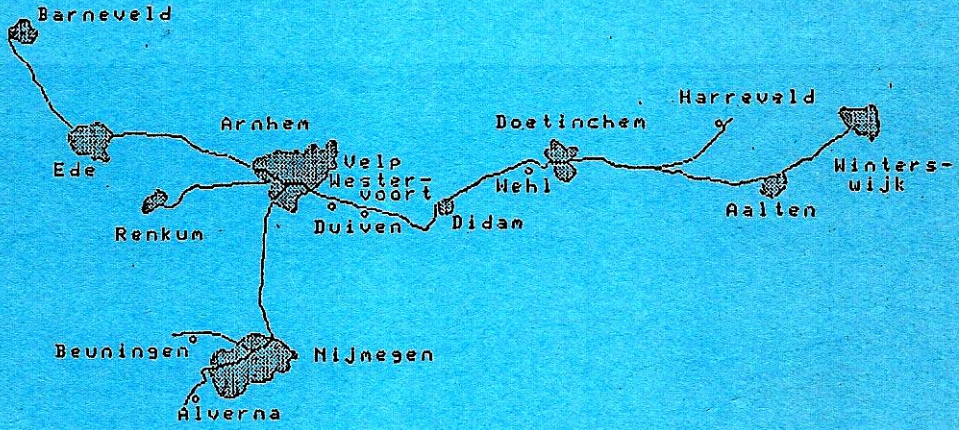
# Tempus Fugit

knipselkrant van de Acorn Computerclub

Regio Arnhem



REDAKTIE : G. Bouwman



juni 1986

## ELECTRON GEBRUIKERS OPGELET!!!

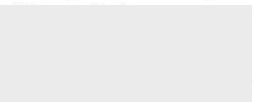
Als lid van een groep van Electron-gebruikers (buiten de Atom Club) beschik ik over een grote hoeveelheid software voor de Electron. Daar ik ook lid ben van de Atom Club -waarvan een aantal leden ook een Electron hebben- willen wij graag onze software en drukwerk uitwisselen met dat van die Electron- en Atomisten.

Omdat de programma's eigendom zijn van onze groep en niet van elk lid afzonderlijk, zijn we tot de volgende (algemene) formule wat betreft de uitwisseling gekomen: voor elk programma van U, krijgt U een programma van ons. Uitzonderingen i.v.m. de waarde (=prijs) zijn natuurlijk mogelijk.

Onze belangstelling gaat vooral uit naar de meer serieuze programma's.

Heeft U interesse? Neemt U dan even contact op met mij.

Een kleine greep uit ons archief: zeer geavanceerde spelen zoals 3D-OVERDRIVE, ARABIAN NIGHT? en tientallen andere spelen. Verder hebben we nog programmeertalen enz. Software het liefst op cassette, maar op disk (Plus-3) en unlocked(!) is niet erg bezwaarlijk.



## Het regula-falsi programma.

Het onderstaande programma is een vertaling van een VAX-Pascal programma in Atom-basic. Ik hoop hiermee een aanzet te geven tot het opzetten van een bibliotheek van procedure's voor wiskundig en natuurkundig gebruik van de Atom. Het hierbij afgedrukte programma bestaat voor het grootste deel uit een algemeen bruikbare procedure, die vanuit het hoofdprogramma aangeroepen wordt. De procedure berekent van de functie op regel 360 het nulpunt in het interval dat aangegeven wordt bij de aanroep. Bij de aanroep dient men tevens de gewenste nauwkeurigheid op te geven. Bij terugkomst retourneert de subroutine het nulpunt, eventueel een foutcode, de bereikte nauwkeurigheid en het aantal stappen benodigd voor de iteratie. De subroutine dient als volgt aangeroepen te worden:

```
REGFAL (L'ST'WAARDE,R'ST'WAARDE,PRECISIE,UITKOMST,ERRORCODE,  
#STAPPEN)
```

### Verklaring der parameters:

L'ST'WAARDE: Invoerparameter, floating-point. Linker punt van het interval waarbinnen het nulpunt geschat wordt

R'ST'WAARDE: Invoerparameter, floating-point. Rechterpunt van het interval waarbinnen het nulpunt geschat wordt

PRECISIE: In- en uitvoerparameter, floating-point. Bij binnenkomst de maximale onnauwkeurigheid. Bij uitvoer de werkelijk gevonden nauwkeurigheid

UITKOMST: Uitvoerparameter, floating-point.

ERRORCODE: Uitvoerparameter, integer. Bij een opgetreden fout wordt deze parameter ongelijk aan 0 gemaakt. Gedetekteerde fouten:

- 0 = geen fout, uitkomst is geldige waarde
- 1 = L'ST'WAARDE > R'ST'WAARDE
- 2 = Funktiewaarden interval verschillen niet van teken, proces ongeldig
- 3 = Meer als 50 iteraties nodig, te langzame konvergentie.

#STAPPEN: Uitvoerparameter, integer, aantal iteraties dat nodig was om de uitkomst te bereiken

Bij de ontwikkeling ben ik op enige problemen gestoken, zo pikt P-Charme geen floatingpoint funkties, hetgeen de gestructureerdheid van het programma danig verstoort en is de XIF-THEN-ELSE konstruktie niet mooi genoeg voor een duidelijk programma, omdat achter else slechts een regel past. Gebruik van XELSE en XEND zou dit kunnen oplossen?

Een ander probleem bleek dat P-charge slechts 9 variabelen in een procedure-deklaratie pikt. Vandaar dat dit programma beslag legt op de floatingpoint-variabelen: XF, ZX, XY, XM, XN. Misschien redenen voor een nieuwe versie van P-charge?

Ik ben van plan om op soortgelijke wijze een procedure te schrijven voor het oplossen van differentiaalvergelijkingen. Wellicht dat anderen ook nuttige programma's willen of hebben geschreven. Wel. laat dan van je horen! Misschien tot een volgende keer en veel plezier met deze procedure.

```

10 PROGRAM REGULA-FALSI
20
30 G,b
40(*****
50(*Programma: Regula-falsi *)
60(*Interface: Dit programma bepaalt een nulpunt van een *)
70(* gegeven functie. Het vraagt om startwaarden*)
80(* S en E, vraagt vervolgens naar de nauwkeu- *)
90(* righeid. Hierna wordt mbv een Regula-falsi *)
100(* proceduren het nulpunt berekend. Het pro- *)
110(* gramma komt terug met een uitkomst of een *)
120(* foutmelding. In het laatste geval wordt *)
130(* naar nieuwe startwaarden gevraagd, waarna *)
140(* de procedure opnieuw opgeroepen wordt *)
150(*Auteur: Alex Vermeulen, Wageningen (08370-21007) *)
160(*Opm.: Oorspronkelijk geschreven in VAX-Pascal, *)
170(* met wat moeite omschreven in gestruk- *)
180(* teerd ATOM-basis, *)
190(*Nodig: P-charme *)
200(* F-point *)
210(* enig wiskundig begrip *)
220(* Datum: 1 mei 1986 *)
230(*****
240
250 VAR XS, (* linker startwaarde *)
260 ZE (* rechter startwaarde *)
270 XP, (* precisie *)
280 XU: REAL; (* nulpunt *)
290 E, (* error-code *)
300 N: INTEGER; (* aantal iteraties *)
310
320 FUNCTION F (XX: REAL):REAL;
330 !Helods pakt P-charme geen floating-point func!
340
350 BEGIN
360 F:=XX*XX - 3*XX + 2;R.
370 END;
380
390(*****
400(*Procedure: REGFAL *)
410(*Interface: De startwaarden XS en ZE leveren een eerste*)
420(* schatting op. Is de relatieve fout kleiner *)
430(* dan de opgegeven XP dan wordt de benadering*)
440(* doorgegeven in XU en de exakte waarde van *)
450(* de relatieve fout in XP. In N komt het aan-*)
460(* tal iteraties dat nodig was en in geval van*)
470(* een fout komt de errorcode in E *)
480(* Auteur: Alex Vermeulen *)
490
500
510 !!! Gebruikt: F, XS, ZE, XP, XU, E, N !!!
520PROC REGFAL (XS,ZE:XP,XU,E,N):XU,XP,N

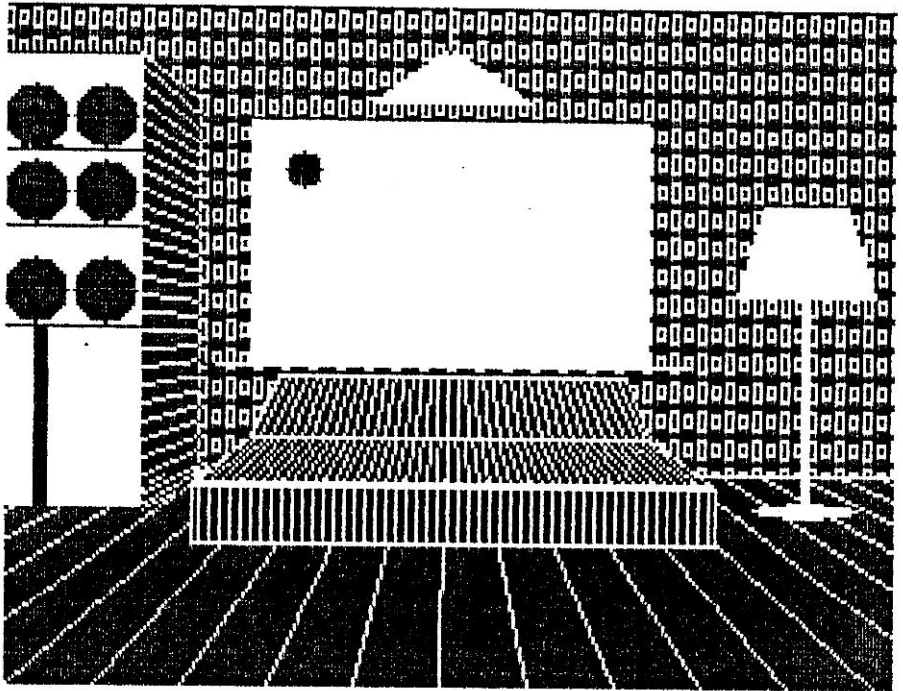
```

```

530
540REM(*1 Initialisatie van G, ZE, E en N *)
550 G=1
560 E=0
570 N=0
580 ZO=ZS
590 WHILE G
600
610REM(*2 Bereken een benadering voor ZS en ZE van ZU met *)
620REM(* precisie XP. Pas desgewenst E of G aan. *)
630 XIF ZS<ZE THEN
640
650REM (*2.1 Bereken de funktiewaarden ZL en ZR voor ZS *)
660REM (* ZE en pas E aan bij fout *)
670
680 XX=ZS;GOS.F;ZL=XF
690 XX=ZE;GOS.F;ZR=XF
700 XIF ZL*ZR(<=0 THEN
710
720REM (*2.2 Bereken een nieuwe benadering ZY en zijn *)
730REM (* funktiewaarde ZN; bepaal ZR *)
740 N=N-1
750 ZY=(ZR*ZS-ZL*ZE)/(ZR-ZL)
760 ZN=ABS(ZO-ZY)
770 ZO=ZY
780 ZX=XV;GOS.F;ZN=XF
790
800 (*2.3 Geef afhankelijk van de tekens de waarde *)
810 (* van ZY aan ZS of ZE; fout?:pas E aan *)
820 FIF ZN=0;ZN=0;G=0;G.e
830 XIF ZN*ZL<0 THEN
840 ZE=ZY
850 ELSE:FIF ZN*ZR<0;ZS=ZY
860 XIF ZN<0
870 ZP=ZN
880 ZU=XV
890 G=0
900 ELSE
910 XIF G=0
920 ZP=ZN
930 ZU=XV
940 ELSE
950 XIF N>50 THEN
960 G=0
970 E=3
980 ELSE
990 ELSE G=0;E=2
1000 ELSE G=0;E=1
1010 WEND
1020 PEND
1030
1040 E=0
1050 FIN."Precisie"XP
1060 DO
1070 IF E<>0;P."foutkods "E
1080 FIN."Linker waarde"ZS

```

```
1070 FIN."Rechterwaarde"%E
1100 REGFAL(ZS,ZE,ZP,ZU,E,N)
1110 UNTIL E=0
1120 FP.'"UITKOMST: "ZU'
1130 FP.'"Precisie: "ZP'
1140 P.'"In "N" stappen"'
1150 END
```



## Kasboek

Met dit kasboek is het mogelijk om uw inkomsten en uitgaven bij te houden (dus ook het bekende huishoudboekje.)

Ik zelf moest een kasboek bijhouden voor een zakelijke toepassing maar ik zag dat geschrijf en rekenen niet zo zitten en aangezien ik toch een ATOM heb kan deze dat naar mijn mening sneller en nauwkeuriger. En zie hier het resultaat, een kasboek waarin je niets meer hoeft uit te rekenen en alles voor je gedaan wordt. (gemak dient de mens)

**Voordelen:**

- 1 geen rekenen meer.
- 2 Menugestuurd, dus overzichtelijk.
- 3 schrijven overbodig. (wel een printer nodig.)
- 4 printeruitdraai volgens "het boekje"
- 5 keuze tussen maandafluiting of openlaten
- 6 data onafhankelijk van programma (apart gasaved)

**Nadelen:**

1. Gegevens moeten wel ingetyped worden. (inpraten wil niet bij mij.)
2. Je moet weer achter de ATOM gaan zitten. (voor zover dit een nadeel is.)

Het programma gebruikt P-charme en loopt van #2900 tot #46FA, een 16K-kaart is dan ook vereist. De kasboekgegevens komen te staan vanaf #8200.

Elke post wordt opgeslagen als een record, hierdoor is het mogelijk in het kasboek te zoeken op omschrijving of datum.

Ook kan er gezocht worden op een zoekbegrip d.m.v. de ^ en het ?

De ^ maakt uit tot hoever er in het zoekargument gezocht wordt.

Staat er een ? dan wordt die plaats overgeslagen bij het zoeken. Laten we als voorbeeld eens de naam JANSEN nemen.

JA^ laat dus alle posten zien die met JA beginnen

JA?S?N laat dus alle posten zien die de letters J,A,S,N bevatten.

Natuurlijk kunt u deze twee ook combineren.

Genoeg theorie nu, tijd voor een voorbeeld, dat zegt soms meer als duizend woorden.

Laten we eens een gefingeerd huishoudboekje bijhouden.

Dit zijn de inkomsten:

01-3 loon f.1600,53 (Let op datum, 1 cijfer dan een 0 ervoor.)

10-3 ontvangen P.Pietersen f.500.00

15-3 terug elektriciteit f.130.25

En nu de uitgaven:

02-3 gas f150.00

05-3 boodschappen f.200.00

15-3 elektriciteit f.185.30

20-3 telefoon f.50.75

25-3 boodschappen f.175.53

De printercode's zijn voor een Star-Gemini 10x, deze staan in de procedure uitdraai. Ik denk dat als je aan de slag gaat zich alles verder wel wijst.

O, ja een ding nog, neem altijd met keuze 1 als je gaat beginnen, het kasboek moet nl. eerst geïnitialiseerd worden.

Voor op-en/of aanmerkingen, verbeteringen enz. houd ik mij graag aanbevolen.

---

G.Hammink  
Bildersstraat 22  
6717NC Ede  
08380-35267

```

-10 PROGRAM KASBOEK V1.0
20
30 DATA"NIUW BESTAND","AFDRUKKEN","TOEVOEGEN"
40 DATA"VERWIJDEREN","OVERZICHT"
50 DATA"LADEN","OPSLAAN"
60 DATA"DATUM" ..>"
70 DATA"OMSCHRIJVING .....">"
80 DATA"DEBET/CREDIT ..>","BEDRAG .....">"
90 DATA"OMSCHRIJVING","DATUM","HELE KASBOEK OP SCHERM"
100 DATA"PRINTERUITDRAAI"
110
120 PROC ADRES
130 X=(#B2B5-?#B20B)+N1?#B20B
140 PEND
150
160 PROC FIX(ZX),Ze,Z
170 REM in ZX:uit ZX MET 2cijfers achter de komma
180 Ze=e;Z=Z(100|ABSZX+0.51)-ABSZX|100;IFZ>99P.ZX+SNGZX,00"
190 FIFZX(OFIFZX)-100 DO P." ";U.C.)=e-Z;e=0
200 P.ZX."";e=0;IFZ<10;P.0
210 P.;e=Ze
220 PEND
230
240 PROC TOETS
250 HT.0;VT.15
260 P."DRUK OP EEN TOETS "
270 LT.#FFES
280 PEND
290
300 PROC VERGELIJK,I,L
310 V=0;L=LEN(A)
320 FOR I=0TO L;IF(A?I=X?I)+(A?I=63);E.340
330 V=(A?I<94)X(1-2*(A?I<X?I));I=L
340 NEXT I
350 PEND
360
370 PROC AKKOORD
380 DO
390 VT.14;IN."AKKOORD"%A
400 UNTIL ?A=74OR?A=7B
410 PEND
420
430 PROC DRUKAF,I
440 IF ?#B202=0 P."bestand leeg **$7;PEND
450 ADRES;IF ?Y=#FF PEND
460 FOR I=0TO?#B204
470 P.$X " "
480 X=X+LEN(X)+1
490 NEXT I
500 P.'
510 PEND
520
530 PROC UITDRAAI,I,J,D
540REM printercode's voor star gemini-10X
550 IF ?#B202=0 P.$12"bestand leeg **$7;TOETS;PEND
560 P.$12
570 HT.5;P."KOPTEKST J/N"
580 DO
590 KEY E
600 UNTIL E=74 OR E=7B
610 IF E=74 IN."TEKST"%C
620 P.$12$21$2
630 P.$27$6B$40$0
640 P.$27$69
650 P.$9;P.$C"
660 P.$27$70
570 PROC LEEG,I
580 P.$12;?#E1=0
1990 P." " " HET EVENTUEEL AANWEZIGE" " BESTAND WORDT "
2000 P."GEWIST!!";AKKOORD;IF ?A=7B PEND
2010 ?#B202=0;#B20C=00
2020 !#B20C=00
2030 ZA=0;ZB=0;ZC=0;ZD=0;ZE=0
2040 Z!#B210=0;Z!#B215=0;Z!#B21A=0
2050 FOR I=#B220TO#B27F;?I=0;N.
2060 FOR I=#B285TO#B69STEP?#B20B
2070 ?I=#FF
2080 NEXT I
2090 P.$12
2100 IN."VOOR WELKE MAAND WILT U BOEKEN"%"C
2110 PEND
2120
2130 PROC ZIEKNAAM,I,B,F
2140 P.$12"zoeken op postnaam";F=FALSE
2150 VT.5;IN."ONSCHRIJVING"%A
2160 IF #A=" " PEND
2170 K=?#B207;SPATIESA
2180 IF?#B202=0 P.$12$7"bestand leeg";TOETS
2190 VT.9;P."wacht"
2200 F.I=ITD?#B202
2210 N=#B21F?I
2220 ADRES;X=X+LEN(X)+1
2230 VERGELIJK
2240 IFV=OF=TRUE;P.$12;DRUKAF;TOETS
2250 KEY B
2260 IF B=13 I=1000;PEND
2270 N.I
2280 XIF F=FALSE P.$12;VT.6;P."DEZE NAAM KOMT NIET"
2290 P."HET BESTAND VOOR!!"
2300 TOETS
2310 ELSE
2320 PEND
2330
2340 PROC ZOEKDATUM,I,B,F
2350 P.$12"zoeken op datum";F=FALSE
2360 VT.5;IN."DATUM"%A
2370 IF #A=" " PEND
2380 K=?#B206;SPATIESA
2390 IF?#B202=0 P.$12$7"bestand leeg";TOETS;PEND
2400 VT.11;P."wacht"
2410 F.I=ITD?#B202
2420 N=#B21F?I
2430 ADRES;VERGELIJK
2440 IF V=OF=TRUE;P.$12;DRUKAF;TOETS
2450 KEY B
2460 IF B=13 I=1000;PEND
2470 N.I
2480 XIF F=FALSE P.$12;VT.6;P."DEZE DATUM KOMT NIET"
2490 P."HET BESTAND VOOR!!"
2500 TOETS
2510 ELSE
2520 PEND
2530
2540 PROC AFDRUKLIJST,I,B
2550 P.$12;?#E1=0;e=0
2560 FOR I=1TO?#B202
2570 N=#B21F?I
2580 IF I=0;L.I.#FFES
2590 DRUKAF
2600 KEY B
2610 IF B=13 I=1000;PEND
2620 NEXT I

```



