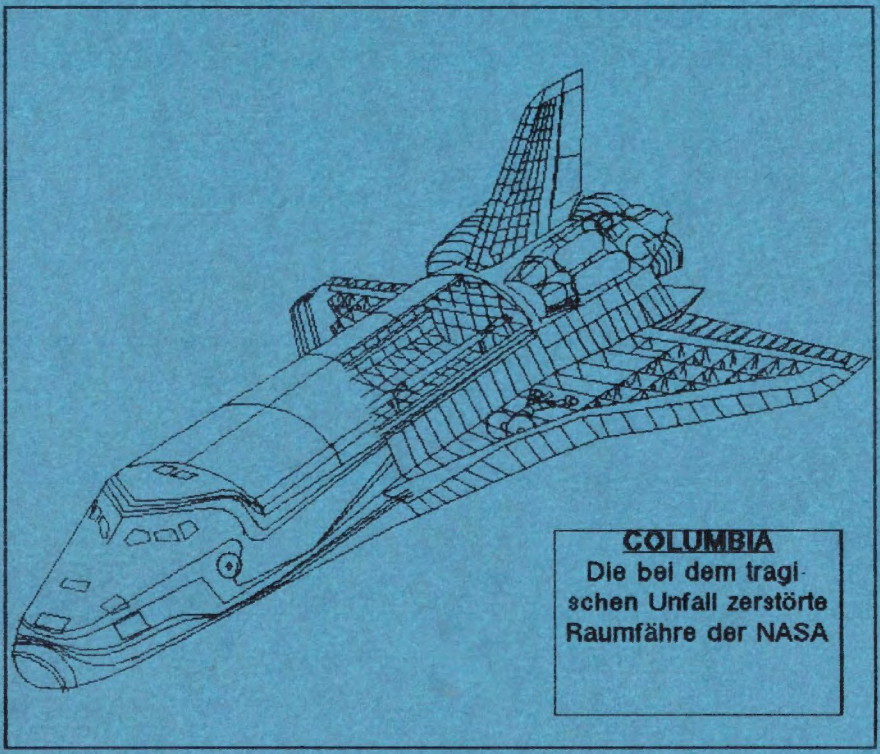
ti news

TI-COMPUTERCLUB • AACHEN E.V. • Nummer 1/1988



COLUMBIA
Die bei dem tragischen Unfall zerstörte Raumfähre der NASA

Die Vereinszeitschrift des
TI-Computer e.V. Aachen

IMPRESSUM

Herausgeber: TI-COMPUTER e.V. Aachen

Postanschrift: Alfred Scheller, Alte Vaalser Str. 65,
5100 Aachen, Tel 0241/83325
Ulrich Prümmer, Schulweg 23,
5108 Monschau/Kal, Tel. 02472/1663

Redaktion: Stefan Böhmer, Manfred Straatmann
Alfred Scheller, Ulrich Prümmer

Titelbild:

Gestaltung: Ulrich Prümmer

Verlag: CPV Computer Programm Verlag

Verantwortlich für den Inhalt der Artikel sind die Autoren

Autoren: Alfred Scheller, Michael Brod, Manfred Straatmann
Rolf Buchholz, Peter Voigt, Günther Faust,
Stefan Böhmer

TI News ist das Vereinsorgan des TI-COMPUTER e.V. mit Sitz Aachen

Für unaufgefordert eingesandte Manuskripte oder Datenträger übernehmen wir keine Haftung. Beiträge werden gern entgegengenommen, ohne Garantie auf Veröffentlichung.
Privatanzeigen sind kostenlos. Kommerzielle Anzeigen auf Anfrage. Die Zeitschrift, sowie der Versand, ist für unsere Mitglieder kostenlos. Nichtmitglieder erhalten diese Zeitschrift gegen Voreinsendung von 3,00 DM plus 1,50 DM für den Versand in Bar oder Briefmarken.

VORWORT

Nach langem Warten liegt nun die Weihnachtsausgabe der TI NEWS vor Ihnen. Immer sind es (noch) dieselben die auch zu dieser Ausgabe ihren Teil beigetragen haben. Es wird, wie in den anderen Ausgaben immer wieder um Ihre Mithilfe gebeten. Diese Zeitschrift wird mit den Beitraegen und Programmen der Leser steigen oder fallen. Es soll Ihre Zeitschrift, ein Medium fuer USER sein. Deshalb fuehlt Euch angesprochen und aufgerufen mitzumachen!!!! Da die Zahl der TI USER dauernd abnimmt wollen wir versuchen mit einer neuen Ueber regionalen Ausgabe die verbliebenen Freunde des TI weiter mit Information zu versorgen. Die Existenz der weiteren Ausgaben wird einzig und allein von der Resonanz auf diese Ausgabe abhaengen. Um weiter hin einen Anreiz fuer die Arbeit mit dem TI zu geben, richten wir ein Forum fuer TURBO PASCAL und C PROGRAMMIERER ein. Wir hoffen so ein weiteres Interessengebiet bei Euch zu wecken

Zum Abschluss moechte ich Ihnen und Ihrer Familie ein frohes Weih nachtsfest und fuer 1988 ein gesundes Neues Jahr wuenschen.

gez. Alfred Scheller

I N H A L T

Titel	Seite
Impressum	2
Vorwort	3
Inhalt	3
Apfelmännchen	4
Lust und Frust	14
Lösen von linearen Gleichungs-systemen	16
Freundschaftstreffen in Belgien	24
Börse	25
Joystickabfrage in Pascal	26
Aufnahmeantrag	29
Letzte Seite	32

Apfelmännchen

Dieses Fractal-Programm enthaelt einige sinnvolle Erweiterungen zu dem in TI-REVUE 9/86 veroeffentlichten Programm. Ausserdem habe ich dieses Programm in 'C' geschrieben, um die Faehigkeiten dieser Sprache auf dem TI auszutesten. Benutzt habe ich dazu den c'99 Compiler V2.1 von Clint Pulley. Geschwindigkeitsmaessig besteht kein grosser Unterschied zum reinen Assemblerprogramm, da die Floating-Point (Fliesskomma) - Routinen den Hauptanteil der Rechenzeit benoetigen, und die sind ja in beiden Faellen gleich.

Zum Betrieb des Programmes werden einige Bibliotheksroutinen benoetigt, die aber alle auf den c'99 Disketten enthalten sind. Folgende Routinen sind notwendig:

1.) BITRTN
Dieses File beinhaltet die Bitmap-Grafik Routinen und muss als C-Quellcode vorliegen, da es mit Hilfe des 'include'-Befehls in den Quellcode des Fractal-Programms einge bunden wird.

2.) FLOAT
Dieses File stellt die Floating Point Routinen zur Verfuegung.

3.) TCIO
Mit diesen Routinen koennen auch Dateien im 'INTERNAL' Format erstellt werden, die zum Speichern der Floating Point Zahlen notwendig sind.

4.) GRF1
Dieses File ist notwendig, um aus dem Bitmap-Modus wieder zurueck in den Grafik- oder Textmodus zu schalten.

5.) CSUP
C Support File. Dieses File muss immer geladen werden.

Die vier letztgenannten Files muessen im Objektcode vorliegen, da sie zum eigentlichen Fractal-Programm dazugeladen werden. Beim Compillieren des Programmes kann die Geschwindigkeit noch etwas gesteigert werden, indem die Frage "Inline Push Code ?" mit Y (Ja) beantwortet wird.

Laden der Programme:

Um Ueberschneidungen des vom Disk-Controllers benoetigten Speichers mit dem vom Bitmap-Mode benoetigten Speicher im VDP-Ram zu vermeiden, muss vor dem Programmstart unbedingt CALL FILES(1) eingegeben werden. Daraus ergibt sich, dass ein Programmstart nur von TI-BASIC aus moeglich ist. Wird dies vergessen, haengt sich der Computer beim Laden oder Speichern eines Fractals auf.

Laden und Starten:

```
CALL FILES(1)
CALL
LOAD("DSK1.FRACT/O","DSK1.TCIO/O","DSK1.GRF1","DSK1.CSUP",
"DSK1.FLOAT")
CALL LINK("START")
```

Bedienung des Programms:

Das Ausgangsmenue enthaelt vier Punkte, die ich im Folgenden beschreiben moechte:

1.) Neues Fractal erstellen

Der Rechner verlangt dazu folgende Eingaben:

- die unteren und oberen Grenzen der x- und y-Koordinaten
- die Anzahl der Iterationen
- den Dateinamen, unter dem das fertige Fractal gespeichert werden soll. Daraufhin erstellt der Rechner ein neues Fractal.

Durch Druecken von 'Q' kann die Rechenroutine abrochen werden, durch Druecken von 'S' wird der momentene Stand der

Berechnung auf Diskette unter dem angegebenen Dateinamen abgespeichert. Das Fractal kann dann zu einem spaeteren Zeitpunkt fortgesetzt werden (s. Punkt 2).
Da die Tastatur nur beim Sprung zum naechsten Pixel abgefragt wird, kann es etwas laenger dauern, bis der Rechner reagiert. Dadurch wird die Rechenzeit aber nicht unnoetig vergroessert.
Zum Schutz des Bildschirms wird dieser waehrend der Berechnung nach ca. 5-10 Minuten dunkelgeschaltet. Durch Druecken einer beliebigen Taste (ausser 'Q' und 'S') kann der Bildschirm wieder heligeschaltet werden.
Nachdem das Fractal fertiggestellt worden ist, wird es auf Diskette gespeichert.

2.) Altes Fractal fertigstellen
Eine unter Punkt 1 mit 'S' unterbrochene Berechnung kann hiermit fortgesetzt werden. Dazu ist der Dateiname einzugeben, unter dem das halbfertige Fractal gespeichert wurde. Nach Tastendruck wird die Berechnung an dieser Stelle fortgesetzt (s. Punkt 1).

3.) Altes Fractal laden
Hiermit kann ein fertiges Fractal geladen und betrachtet werden.

4.) Programm beenden
Die uebliche c'99 Exit/Rerun Meldung erscheint etwas verstuemmelt, da der Zeichensatz fuer Kleinbuchstaben ueberschrieben wurde. Durch Druecken von 'Y' startet das Programm von Neuem, druecken von 'N' bewirkt einen Ruecksprung ins TI BASIC.

Im Moment arbeite ich noch an einer Hardcopy Routine, die in einer der naechsten Ausgaben veroeffentlicht wird.

```
#asm
 REF  VMBW,DSRLNK
PAB  DATA  >0500, >0000
     DATA  >0000, >1800, >0000
PAB1 BSS  16
STAT DATA 0
LO   DATA  >0500
SA   DATA  >0600
WORK BSS  32
#endasm

extern itof(),ftoi(),stof(),ftos(),fexp();
extern fcom(),fpget(),fpput(),fint(),fcpy();
extern grf1();
extern topen(),tclose(),tread(),twrite();
#define float char
#define SEQUENTIAL 0
#define INTERNAL   8
#define OUTPUT     2
#define INPUT      4
#define FIXED      0
#include dsk1.bitrtn

char fn[14];
int kx,ms,n,fp;
float xo[8],xu[8],yo[8],yu[8];

main()
{
 while (1) {
  grf1();
  puts("        APFELMAENNCHEN\n");
  puts("        ..............\n\n");
  puts(" 1   NEUES FRACTAL ERSTELLEN\n\n");
  puts(" 2   ALTES FRACTAL FERTIGSTELLEN\n");
```

```
  puts(" 3 - ALTES FRACTAL LADEN\n\n");
  puts(" 4 - PROGRAMM BEENDEN\n\n\n");
  puts("      > ");
  switch (getchar()) {
  case '1': inp(); fract();
            break;
  case '2': getfn(); dload();
            fract(); break;
  case '3': getfn(); load();
            key(); break;
  case '4': exit(0);
  }
 }
}

inp()
{
 char akx[6],si[14];
 puts("\n\n");
 puts("XU: "); fpget(si,xu);
 puts("XO: "); fpget(si,xo);
 puts("YU: "); fpget(si,yu);
 puts("YO: "); fpget(si,yo);
 puts("ANZ. ITERATIONEN: "); gets(akx);
 kx = atoi(akx);
 getfn();
 n = ms = 1;
 bitmap(16,2);
}

fract()
{
 float dy[8],xc[8],yc[8],xz[8],yz[8],
       xx[8],yy[8],hl[8],hm[8],n1[8],
       m1[8],tw[8],dx[8];
```

```
int   c,m,k,s,h,ze,sp,tu;

h = k = 0; s = 100; ze = 192; sp = 256; tu = 2
itof(tu,tw);
fexp(xo," ",xu,hl), itof(sp,hm); fexp(hl,"/",hm,dx)
fexp(yo," ",yu,hl); itof(ze,hm); fexp(hl,"/",hm,dy)

for (m = ms; m < 193 + m) {
 while (n) {
  c = poll(0);
  if (c = = 'Q') return;
  if (c = = 'S') {
   save(); dsave(m1,n1); return;
  }
  itof(n,n1); fexp(n1,"*",dx,hl); fexp(xu," + ",hl,xc),
  itof(m,m1); fexp(m1,"*",dy,hl); fexp(yu," + ",hl,yc)
  k = 0; xz[0] = xz[1] = 0; yz[0] = yz[1] = 0;
  while ((h < s)&(k ! = kx + 1)) {
    + + k;
   fexp(xz,"*",xz,xx); fexp(yz,"*",yz,yy);
   fexp(xz,"*",yz,hl); fexp(hl,"*",tw,hm); fexp(hm," ",yc,yz)
   fexp(xx," ",yy,hl); fexp(hl," ",xc,xz);
   fexp(xx," + ",yy,hl); h = ftoi(hl);
  }
  if (k! = kx + 1) {
   if (k%2 = = 0)
   plot(n,m,16),
   h = 0;
  }
   + + n;
 }
 n = 1;
}
save();
}
```

```
atoi(s) char *s;
{ int sign,n;
  while (*s = = ' ') + + s;
  sign = 1;
  if (*s = = '-') { sign = -1; + + s; }
  if (*s = = '+') + + s;
  n = 0;
  while ((*s > = '0')&(*s < = '9')) n = 10*n + *(s + + )-'0';
  return (sign*n);
}

getfn()
{
 int  i,pa,p;
 char *adr;
 do {
  puts("\n\nDATEINAME: ");
  gets(fn);
  adr = PAB1;
  for (i = 0; fn[i]! = '\0'; + + i)
   *(adr + i) = fn[i];
 } while (i > 13);
 p = PAB1-1;
 *p = i;
 fn[i] = '/'; fn[i + 1] = 'D'; fn[i + 2] = '\0';
}

file()
{
 int p;
 #asm
  LWPI WORK
  LI   0, >1C00
  LI   1,PAB
```

```
  LI   2,26
  BLWP @VMBW
  LI   0, >1C09
  MOV  0,@ >8356
  BLWP @DSRLNK
  DATA 8
  JEQ  BAD
  CLR  @STAT
  JMP  RETURN
BAD LI 1, >FFFF
  MOV  1,@STAT
RETURN LWPI  >8300
 #endasm
 p = *STAT;
 if (p)  err();
}

load()
{
 #asm
  MOV  @LO,@PAB
 #endasm
 bitmap(16,2);
 file();
}

save()
{
 #asm
  MOV  @SA,@PAB
  #endasm
 file();
}

dload()
```

```
{
 float kxf[8],m1[8],n1[8];
 fp = topen(fn,INPUT + INTERNAL + FIXED,9);
 if (fp) { err(); return; }
 readf(xu); readf(xo);
 readf(yu); readf(yo);
 readf(m1); readf(n1);
 readf(kxf);
 tclose(fp);
 ms = ftoi(m1); n = ftoi(n1); kx = ftoi(kxf);
 load();
}

readf(a)
char *a;
{
 char buff[10];
 int i,s;
 tread(buff,-1,fp,&s);
 for (i = 1;i < 9; + +i)
  *(a + i-1) = buff[i];
}

dsave(m1,n1)
float m1,n1;
{
 float kxf[8];
 itof(kx,kxf);
 fp = topen(fn,OUTPUT + INTERNAL + FIXED,9);
 writef(xu); writef(xo);
 writef(yu); writef(yo);
 writef(m1); writef(n1);
 writef(kxf);
 tclose(fp);
}
```

```
writef(a)
float a[];
{
 char buff[10];
 int i;
 buff[0] = 8;
 for (i = 1;i < 9; + + i)
  buff[i] = a[i-1];
 twrite(buff,0,fp,9).
}

key()
{
 while (poll(0) = = 0) ;
}

err()
{
 grf1();
 puts("\n  DATEIFEHLER\n\n  TASTE DRUECKEN");
 while (poll(0) = = 0) ,
}
```

Tagtaeglich kann man es in allen Medien hoeren und lesen - wer heute noch ohne Computer ist -ist nicht - in - !

Lust und Frust

Hatte nicht frueher mal mein Sohn mit so einem Ding gespielt? Ja natuerlich - ich hatte es sogar zur Sortierung eingesetzt - bzw. ein anderer tat es fuer mich. Aber es hatte doch funktioniert. Also sollte ich das auch koennen. Fein verpackt, stand der Karton auf dem Boden und fristete hier sein Dasein. Dies sollte jetzt geaendert werden. Waere doch gelacht, was andere konnten, konnte ich auch (ach haette ich es doch nur gelassen) ! Nun gut. Frisch ans Werk. Zunaechst erstmal eine Bestandsaufnahme gemacht. Was besass ich denn nun? Eine Konsole TI 99/4A, einen Trafo, einen Joystick, einen TV-Signalumsetzer, verschiedene Module, einen Kassettenrekorder, Kassetten und einen Haufen Zeitschriften und viel Mut. Zudem hatte ich eine Schreibmaschine mit einem Interface, die ich als Drucker einsetzen konnte. Alle Kabel richtig angeschlossen? Potz-Blitz, der kleine Fernseher fehlte noch. So, jetzt alles auf ON - und .....? Ein schoenes Bild hatte ich ja. Aber wie gings nun weiter? Wozu hatte ich denn genuegend Lektuere? Nach zwei Stunden eifrigen lesens, musste ich die erste Tasse Kaffee haben. Ich muss wohl etwas komisch dreingeschaut haben, denn meine Frau fragte besorgt, ob mir nicht gut sei. In der Tat, ich fuehlte mich gar nicht wohl in meiner Haut. Zwei Wochen spaeter hatte ich meine Lektion gelernt. Die Bedienungsanleitung und der Basickurs, den ich unter den Kassetten fand, hatten mir die noetige Zuversicht wieder gegeben. Der TI tat nun sogar, wass ich meistens wollte. Ein paar mal konnten wir uns nicht einigen. In diesen Faellen der Uneinigkeit, habe ich kosequent Haerte gezeigt, mich durchgesetzt und meine Zeitung gelesen. Von wegen Friede, Freude, Eierkuchen, mit mir nicht! Tage spaeter wollte ich nun auch den Drucker benutzen. Doch dazu fehlte mir die Schnittstelle und deshalb durchforstete ich alle Verkaufsanzeigen und siehe da, da war ja schon einer -, der TI-Teile abgeben wollte. Nichts wie hin, damit mir auch kein Anderer

dazwischen kam. Also eine Konsole hatte ich ja und sonst? Ja, da war noch ein Laufwerk, nebst einem Disk-Kontroller. Ob ich vielleicht nun ich habe gekauft - aber mein Drucker lief immer noch nicht. Bei den Unterlagen, die ich vorgefunden hatte, war mir auch eine Adresse vom Aachener TI-Club aufgefallen. Es war schon ziehmlich spaet, ob ich dort wohl noch anrufen konnte? Egal, wie war noch die Nummer? Die sonore Stimme am anderen Ende klang sehr freundlich und ich schuettete, diesem mir Unbekannten, mein ganzes Herz aus. Und dieser Mensch hatte sogar Verstaendnis und lud mich ein und ich durfte ihn am gleichen Abend noch besuchen. Na - wie fand ich denn das? Etwas spaeter waren wir schon fleissig beim Fachsimpeln. Und was dort alles stand. Es dauerte nicht lange und ich wusste, solche Anorndnungen nannte man Konfiguration in so einer Box konnte man einfach Karten aufstecken und ....! So etwas musste ich auch unbedingt haben. Also wirklich, das sah nicht nur profihaft aus, es war auch so, denn mit einigen Griffen spulte die Anlage ihr vorher eingegebenes Programm ab, sodass ich nur mit offenstehendem Mund staunen konnte. Ehe ich mich wieder fassen konnte, hatte der Mensch einen Ordner in der Hand, fischte eine Adresse heraus und ein paar Tage spaeter war auch ich gluecklicher Besitzer einer Box. 99/4 PERIPHERAL EXPANSION SYSTEM nannte man das. Als meine Frau meinen schoenen Fang besichtigte, traf mich ein mitleidiger Blick. Und richtig, ich hatte jetzt alles doppelt und dreifach. Zwei Konsolen, zwei Disk-Kontroller, Buecher, Module, Kassettenrekorder u.s.w., also mich musste wirklich der Teufel geritten haben, denn mein Drucker lief immer noch nicht. Macht nichts, ich war ja zum morgigen TI-Clubabend eingeladen. Vier Mann waren schon da, als ich kam und der Rest ...? Zum weiterem ueberlegen kam ich erst gar nicht. Nach meiner Vorstellung wurden meine Probleme durchleuchtet. Und ehe ich mich versah wurde ich nach Haus geschickt, um den Drucker zu holen. Nach dem Test beim Kollegen war klar, dass ein Steuerungskabel nicht richtig tickte. Spontan wurde Hilfe in Reinkultur angeboten. Der Drucker wanderte mit in die Eifel und zwei Tage spaeter hatte ich das Geraet nebst angepasster Verbindung zurueck und der TI-Club Aachen ein Mitglied mehr. So wurde aus Frust schliesslich Lust! Fortsetzung folgt.

Euer ARMAN

Lösen von linearen Gleichungssystemen

Dieses Programm dient einerseits zum Loesen von linearen Gleichungssystemen mit maximal neun Unbekannten und andererseits der Approximation von Polynomen. Es ist selbstdokumentierend geschrieben, daher sind weitere Erlaeuterungen ueberfluessig.

```
 100 ! ********************
110 ! *                  *
120 ! *     Lineare      *
130 ! *   Gleichungs-    *
140 ! *     systeme      *
150 ! *       und        *
160 ! *     Polynom-     *
170 ! *  approximation   *
180 ! *                  *
190 ! *                  *
200 ! * (C) 1983-87 by   *
210 ! *                  *
220 ! * Günther Faust    *
230 ! * Schwanenstr. 56  *
240 ! * 5173 Aldenhoven  *
250 ! * Tel. 02464/6516  *
260 ! *                  *
270 ! *  DSKx.LINEARGL5  *
280 ! *                  *
290 ! *   Version 8/86   *
300 ! *                  *
310 ! ********************
320 !
330 DATA Lineare Gleich.-Sys-
teme..1,Polynomapproxima-
tion.....2,Zurück zum Menü.....3
340 !
350 ! *  Deklarationen   *
360 !
370 ! 380 GOTO 460
385 DIM
A(10,10),B(30),X(30),Y(30),V(10),U(
19),XZ(30),YZ(30),TEXT$(4)
390
C,D,E,F,G,H,I,J,K,L,M,N,O,P,XU,XO
,HX,KEY,SI2,DE2,STATUS = Z
400 F$,G$ = KO$
410 CALL CHAR :: CALL CLEAR
:: CALL HCHAR :: CALL KEY
420 IMAGE ##
430 IMAGE ########,##
440 IMAGE ##.#####
450 !@P-
460 !
470 !Sonderzeichen
480 != = = = = = = = = = = = =
490 !
```

```
500 CALL
CHAR(91,"00280038444474440028007C4444447C0028004444444438")
510 CALL
CHAR(123,"0000280038447C44000028007C44447C00002800444444380000EE884422EE")
520 CALL
CHAR(129,"0000003C48484830")
530 !
540 !***********************
550 !
560 !
570 ! Programmwahl
580 ! = = = = = = = = = = = = =
590 !
600 FOR I=1 TO 3 :: READ
TEXT$(I):: NEXT I
610 DISPLAY AT(3,11)BEEP
ERASE ALL:"Men}:":
:TEXT$(1):TEXT$(2):TEXT$(3)
620 GOSUB 2130 :: IF KEY=49
OR KEY=50 OR KEY=51 THEN
630 ELSE 620
630 I=KEY-48 :: DISPLAY
AT(3,1)ERASE ALL:TEXT$(I)
640 GOSUB 2120 :: IF KEY=78
THEN 610 ELSE IF KEY<>74
THEN 640
650 ON I GOTO 720,1140,660
660 RUN "DSK1.LOAD"
670 !
680 ! ******************
690 ! Lineare Gleichungen
700 ! ******************
710 !
720 GOSUB 2720 ! Text
730 DISPLAY AT(24,1):: DISPLAY
AT(8,5)BEEP:"Matrix-Ordnung?
M="
740 CALL EIN2(8,23,9,M)
750 GOSUB 2120 ! Text
760 IF KEY=74 THEN 780 :: IF
KEY<>78 THEN 750 ELSE 730
770 !
780 ! Matrix Eingabe
790 !
= = = = = = = = = = = = = =
800 !
810 DISPLAY AT(8,3):: DISPLAY
AT(24,1)
820 Z=10-M/2 :: FOR J=1 TO M
830 FOR I=1 TO M :: DISPLAY
AT(Z+I,6):"A(";STR$(I);",";STR$(J);"
)=" :: NEXT I
840 FOR I=1 TO M :: GOSUB
880 :: NEXT I
850 GOSUB 2120 ! Text
860 IF KEY=74 THEN 890 ELSE
IF KEY<>78 THEN 850
870 DISPLAY AT(24,6)BEEP:"Ein
gabe I= " :: CALL
EIN2(24,16,M,I):: GOSUB 880 ::
GOTO 850
880 CALL EIN1(Z+I,14,A(I,J)):
CALL ZP(Z+I,14,A(I,J)):: RETURN
890 DISPLAY AT(24,1):: NEXT J
900 GOSUB 2240 ! Eingabe der
rechten Seite
910 GOSUB 2720 ! Text
920 GOSUB 2160 ! Text
930 GOSUB 2360 ! Matrix A
```

```
berechnen
940 CALL PS(1,M,A(,),B(),V())
950 KO$ = "X" :: GOSUB 1030 !
Anzeige der x(i)
960 DISPLAY
AT(24,1)BEEP:"Andere rechte
Seite? - J/N"
970 GOSUB 2130 :: IF KEY = 74
THEN 980 :: IF KEY = 78 THEN
2170 ELSE 970
980 GOSUB 2240 :: P = B(1)::
B(1) = B(H):: B(H) = P :: GOTO
940
990 !
1000 !Anzeige x(i) oder a(i)
1010
!= = = = = = = = = = = = = = = =
= = = = = =
1020 !
1030 Z = 10-M/2 :: FOR I = 1 TO
M :: DISPLAY AT(Z + I,1)
1035 DISPLAY
AT(Z + I,6):KO$;"(";STR$(I);") = "
1040 IF ABS(V(I)) < 1E8) THEN
DISPLAY AT(Z + I,12):USING
430:V(I)ELSE DISPLAY
AT(Z + I,11):USING 440:V(I)
1050 NEXT I :: RETURN
1060 F$ = "" :: FOR I = 1 TO M-1 ::
F$ = F$&" + "&KO$&STR$(I + 1)&"*X^
"&STR$(I)::NEXT I
1070 K = INT(LEN(F$)/21):: G$ = ""
:: FOR I = 0 TO K-1
1073
G$ = G$&SEG$(F$,1 + I*21,21)&"
" :: NEXT I
1075
G$ = "A1"&G$&SEG$(F$,1 + I*21,LEN
(F$)-I*21)
1080 DISPLAY
AT(10 + M/2 + 4,1)BEEP:"Y = "&G$
:: RETURN
1090 !
1100 ! ********************
1110 ! Polynomapproximation
1120 ! ********************
1130 !
1140 DISPLAY AT(1,1)ERASE
ALL:"Polynomapproximation nach
der Methode der
kleinsten":TAB(9);"Quadrate"
1150 DISPLAY AT(5,1):"Ein-
zugeben sind:": :"Der Grad des
Polynoms (M), die Stützstellen
X(i) und die Stützwerte Y(i)"
1160 DISPLAY AT(12,1):"Bedin-
gungen und Grenzen !":"N-
1 > = M < = 9;I = 1,2,3...N = 30"
1170 DISPLAY AT(16,1):"F}r
M = N-1 liegt das":"Inter-
polationspolynom vor.":"Das
heißt:":"In den Stützstellen X(i)"
1180 DISPLAY AT(21,1):"werden
die Stützwerte Y(i)":"erfüllt."1190
GOSUB 2190 ! Text
1200 GOSUB 2130 :: IF
KEY < > 87 THEN 1200
1240 GOSUB 2730
1250 DISPLAY AT(10,1):"Suchen
Sie die beste Appro = ":"ximation
durch sukzessive":"Änderung des
Grades.": :"Eingabe Polynomgrad
```

```
M = "
1260 CALL EIN2(14,23,9,M)
1270 GOSUB 2120 ! Text
1280 IF KEY = 74 THEN 1290 :: IF
KEY < > 78 THEN 1270 ELSE 1240
1290 CALL CLR(10,24):: DISPLAY
AT(10,1)BEEP:"Liegen
gleichabständige":"Stützstellen
vor? = J/N"
1300 CALL TIME(200):: GOSUB
2130 :: IF KEY = 74 THEN 1360 ::
IF KEY = 78 THEN 1600 ELSE 1300
1310 !
1320 ! ********************
1330 ! * Eingabe Modus "1" *
1340 ! ********************
1350 !
1360 CALL CLR(8,24):: DISPLAY
AT(8,1)BEEP:"Unteres X, (XU) =
":"Oberes X, (XO) = ": "Delta X,
(HX) = "
1370 CALL EIN1(8,18,XU):: CALL
ZP(8,18,XU)
1380 CALL EIN1(9,18,XO):: CALL
ZP(9,18,XO)
1390 CALL EIN1(10,18,HX)::
CALL ZP(10,18,HX)
1400 GOSUB 2120 ! Text
1410 IF KEY = 78 THEN 1370 :: IF
KEY < > 74 THEN 1400
1420 N = (XO XU)/HX + 1 :: IF
N > 30 THEN N = 30
1430 IF M = N THEN GOSUB 2320
1440 GOSUB 2300 ! Tabelle
1450 FOR I = 1 TO N :: K = I
1460 X(I) = XU + HX*(I-1)::
GOSUB 2150
1470 CALL ZP(Z,5,X(I))
1480 GOSUB 2710 ! Eingabe der
y(i)
1490 IF I-INT(I/15)*15 = 0 OR I = N
OR P = -1 THEN 1510
1500 NEXT I
1510 GOSUB 2120 ! Text
1520 IF KEY = 78 THEN 1540 :: IF
KEY < > 74 THEN 1510
1530 IF K = N THEN 1750 ELSE
I = K :: P = 0 :: GOTO 1500
1540 P = 0::DISPLAY
AT(24,1)BEEP:"I = ":CALL
EIN2(24,5,N,I)::Z = 4 + I
15*INT(I/16):: P = 1 :: GOTO 1480
1550 !
1560 ! ********************
1570 ! * Eingabe-Modus "2" *
1580 ! ********************
1590 !
1600 CALL CLEAR :: GOSUB
2300 :: I = 0 !Eingabekopf
1610 DISPLAY AT(24,1):"Ende mit
X(I) = 9E99"
1620 I = I + 1 :: N = I :: P = 0
1630 GOSUB 2690 :: IF
X(N) = 9E99 AND P = 0 THEN 1660
1640 GOSUB 2710 :: IF P = 1 OR
N = 15 OR N = 30 THEN 1660
1650 N = I :: GOTO 1610
1660 GOSUB 2120 :: IF KEY = 74
AND X(N) = 9E99 THEN N = N 1
GOTO 1750
1670 IF KEY = 74 AND N = 15
THEN I = N :: GOTO 1610
```

```
1680 IF KEY=74 AND N=30
THEN 1740
1690 IF KEY<>78 THEN 1660
1700 DISPLAY AT(24,1)BEEP:"I=
" :: CALL EIN2(24,5,N,I):: P=-1 ::
GOTO 1630
1710 !
1720 !********************
1730 !
1740 GOSUB 2320
1750 GOSUB 1930 :: DISPLAY
AT(21,1)BEEP:"Sollen zu":"den
Stützstellen X(I)":"die Stützwerte
Y(I)":"berechnet werden? = J/N"
1760 CALL TIME(200):: GOSUB
2130 :: IF KEY=78 THEN CALL
CLR(20,24):: GOTO 1870 ELSE IF
KEY<>74 THEN 1760
1770 CALL CLEAR :: GOSUB
2300 :: FOR I=1 TO N :: GOSUB
2150
1780 CALL ZP(Z,5,X(I))
1790 CALL FX(I,M,V(),X(),YZ())
1800 CALL ZP(Z,18,YZ(I))
1810 IF I=15 THEN DISPLAY
AT(24,1)BEEP:"Weiter?  W" ELSE
1830
1820 GOSUB 2130 :: IF
KEY<>87 THEN 1820
1830 NEXT I
1840 IF M=N THEN DISPLAY
AT(21,1):"Das vorliegende
Polynom ist":"ein Inter-
polationspolynom!" :: GOTO 1870
1850 DE2=0 :: DISPLAY
AT(24,1):: FOR I=1 TO N ::
DE2=DE2+(Y(I)-YZ(I))^2 ::
NEXT I
1860 SI2=DE2/(N-M):: DISPLAY
AT(22,3):CHR$(129);" =";SQR(SI2)
1870 DISPLAY
AT(24,1)BEEP:"Neuer
Polynomgrad? = J/N"
1880 CALL TIME(200):: GOSUB
2130 :: IF KEY=78 THEN 1920 ::
IF KEY<>74 THEN
1890 DISPLAY
AT(24,1)BEEP:"Polynomgrad M="
:: CALL EIN2(24,15,9,M)
1900 IF M<N THEN 1750
1910 CALL CLR(20,24):: GOSUB
2140 :: GOTO 1890
1920 GOSUB 2040 :: GOTO 2170
1930 GOSUB 2730 :: GOSUB
2160 ! Text
1940 !
1950 ! Koeffizienten-Matrix
1960 ! ********************
1970 !
1980 M=M+1 :: CALL
MATRIX1(0,2*M-2,1,N,X(),U())
1990 CALL
MATRIX2(1,M,1,N,Y(),X(),B()):
FOR J=1 TO M :: FOR I=1 TO
M :: A(I,J)=U(I+J-2):: NEXT I ::
NEXT J
2000 GOSUB 2340 :: CALL
PS(1,M,A(,),B(),V())
2010 CALL CLR(10,24)::
KO$="A" :: GOSUB 1000 ! An-
zeige der a(i)
2020 GOSUB 1060 ! Anzeige des
```

```
Polynoms
2030 RETURN
2040 DISPLAY AT(21,1)BEEP:"Sol-
len f}r beliebige X(I)":"die Y(I)
berechnet werden ?": :"- J/N"
2050 CALL TIME(200):: GOSUB
2130 :: IF KEY=78 THEN
RETURN ELSE IF KEY< >74
THEN 2050
2060 DISPLAY AT(1,1)ERASE
ALL:"* *  Berechnung: y=f(x)  *
*" :: I=0 :: GOSUB 2310 :: DIS-
PLAY AT(24,3):"f}r Ende  =
X(I)=9E99"
2070 I=I+1
2080 IF I>30 THEN 2170
2090 GOSUB 2150 :: CALL
EIN1(Z,5,XZ(I)):: IF XZ(I)=9E99
THEN RETURN ELSE CALL
FX(I,M,V(),XZ(),YZ())
2100 CALL ZP(Z,5,XZ(I)):: CALL
ZP(Z,18,YZ(I))
2110 GOTO 2070
2120 CALL CLR(20,24):: DISPLAY
AT(24,6)BEEP:"Eingaben ok?
=J/N"
2130 CALL KEY(0,KEY,STATUS)::
IF STATUS=0 THEN 2130 ELSE
RETURN
2140 CALL CLR(22,24):: DISPLAY
AT(22,3)BEEP:"Achtung: M> =N!
M=....;STR$(M);",";"N=";STR$(N)::
RETURN
2150 Z=4+I-15*INT(I/16):: DIS-
PLAY AT(Z,1):USING 420:I ::
RETURN
2160 CALL CLR(5,24):: DISPLAY
AT(10,5)BEEP:"< < < Bitte war-
ten! > > >" :: RETURN
2170 DISPLAY AT(24,1)BEEP:"  S
= Start ; E = Ende"
2180 GOSUB 2130 :: IF KEY=83
THEN 610 ELSE IF KEY=69
THEN END ELSE 2180
2190 DISPLAY
AT(24,5)BEEP:"Wenn weiter = W"
:: RETURN
2200 !
2210 ! Eingabe rechte Seite
2220 !
= = = = = = = = = = = = = = = =
= = = =
2230 !
2240 CALL CLR(Z,24):: DISPLAY
AT(24,2):"Eingabe der rechten
Seite"
2250 FOR I=1 TO M :: DISPLAY
AT(Z+I,6):"B(";STR$(I);") = " ::
NEXT I
2260 FOR I=1 TO M :: GOSUB
2290 :: NEXT I
2270 GOSUB 2120 :: IF KEY=74
THEN RETURN ELSE IF
KEY< >78 THEN 2270
2280 DISPLAY AT(24,6)BEEP:"Ein-
gabe I= " ::CALL
EIN2(24,16,M,I)::GOSUB 2290
::GOTO 2270
2290 CALL EIN1(Z+I,13,B(I))::
CALL ZP(Z+I,13,B(I)):: RETURN
2300 DISPLAY AT(1,1)ERASE
ALL:"* * *  Variablenliste  * * *"
```

```
2310 DISPLAY
AT(3,2):"I";TAB(5);"X(I).......";TAB(18
);"Y(I)....." :: RETURN
2320 CALL CLR(8,10):: IF
M > = N THEN GOSUB 2140 ::
M = N-1 ::DISPLAY AT(23,1):"Es
wird mit M = ";STR$(M);" gerech-
net." :: RETURN
2330 !
2340 ! Zeilenvertauschung
2350 !
2360 FOR J = 1 TO M :: FOR I = 1
TO M :: IF
ABS(P) > ABS(A(I,J))THEN 2370
ELSE P = A(I,J):: H = I :: L = J
2370 NEXT I :: NEXT J :: IF H = 1
AND L = 1 THEN 2410
2380 FOR J = 1 TO M ::
P = A(1,J):: A(1,J) = A(H,J)::
A(H,J) = P :: NEXT J :: P = B(1)::
B(1) = B(H):: B(H) = P
2390 FOR I = 1 TO M ::
P = A(I,1):: A(I,1) = A(I,L)::
A(I,L) = P :: NEXT I
2400 !
2410 ! Matrixinversion
2420 ! ------
2430 !
2440 ! (U)nter  & (O)bere
2450 ! Dreiecksmatrix
2460 ! (A) = (U)*(O)
2470 ! -----
2480 !
2490 D = 1 :: FOR I = 1 TO M ::
D = D*A(I,I):: IF ABS(D) > 1E-99
THEN 2510
2500 DISPLAY AT(10,1)BEEP
ERASE ALL:"Das zuletzt ein
gegebene":"Gleichungssystem
ist":"linear abhängig  = >  D = 0" ::
GOTO 2170
2510 FOR J = I + 1 TO M ::
A(J,I) = A(J,I)/A(I,I):: FOR K = I + 1
TO M :: A(J,K) =
A(I,K)*A(J,I) + A(J,K):: NEXT K ::
NEXT J :: A(I,I) = 1/A(I,I):: NEXT I
2520 !
2530 !Inversion von (U)&(O)
2540 !
2550 !
2560 FOR I = 2 TO M :: FOR J = 2
TO I :: G =  A(I,J-1) :: E = -A(M-
I + 1,M-J + 2)*A(M-J + 2,M-J + 2)::
FOR K = J TO I-1 :: G = G-
A(I,K)*A(K,J-1)
2570 E = E-A(M-I + 1,M-K + 1)*A(M-
K + 1,M-J + 2):: NEXT K
2580 A(I,J-1) = G :: A(M-I + 1,M-
J + 2) = E*A(M-I + 1,M-I + 1)::
NEXT J :: NEXT I
2590 !
2600 !INV(A) = INV(U)*INV(O)
2610 !
2620 !
2630 FOR I = 1 TO M :: FOR J = 1
TO M-1 :: G = 0
2640 IF I < = J THEN G = A(I,J)
2650 K = I :: IF I < J + 2 THEN
K = J + 1
2660 FOR K = K TO M ::
G = G + A(I,K)*A(K,J):: NEXT K ::
A(I,J) = G
```

```
2670 NEXT J :: NEXT I :: IF H=1
AND L=1 THEN RETURN
2680 FOR J=1 TO M ::
P=A(1,J):: A(1,J)=A(L,J)::
A(L,J)=P :: NEXT J :: RETURN
2690 GOSUB 2150 :: CALL
EIN1(Z,5,X(I))
2700 CALL ZP(Z,5,X(I)):: RETURN
2710 CALL EIN1(Z,18,Y(I)):: CALL
ZP(Z,18,Y(I)):: RETURN
2720 DISPLAY AT(3,3)ERASE
ALL:"Lineares Gleichungssystem"
:: RETURN
2730 DISPLAY AT(3,5)ERASE
ALL:"Polynomapproximation" ::
RETURN
2740 !@P+
2750 SUB EIN1(Z,S,W):: DISPLAY
AT(Z,S-1):: ON WARNING NEXT
:: ACCEPT AT(Z,S)BEEP
VALIDATE(NUMERIC," "):W ::
SUBEND
2760 SUB EIN2(Z,S,M,W)
2770 ON WARNING NEXT
2780 ACCEPT AT(Z,S)BEEP
SIZE(1)VALIDATE(DIGIT):W :: IF
W>M THEN 2780
2790 SUBEND
2800 SUB TIME(TI):: FOR I=1
TO TI :: NEXT I :: SUBEND
2810 SUB
MATRIX1(U1,O1,U2,O2,X1(),Y1()):
FOR J=U1 TO O1 :: Y1(J)=0 ::
FOR I=U2 TO O2 ::
Y1(J)=Y1(J)+X1(I)^J :: NEXT I
:: NEXT J :: SUBEND
2820 SUB
MATRIX2(U1,O1,U2,O2,A1(),X1(),Y1
()):: FOR J=U1 TO O1 ::
Y1(J)=0 :: FOR I=U2 TO O2 ::
Y1(J)=Y1(J)+A1(I)*X1(I)^(J-1)::
NEXT I :: NEXT J :: SUBEND
2830 SUB
PS(UG,OG,F1(,),F2(),PS())!
Produktsumme
2840 FOR I=UG TO OG ::
PS(I)=0 :: FOR J=UG TO OG ::
PS(I)=PS(I)+F1(I,J)*F2(J)::NEXT
J :: NEXT I :: SUBEND
2850 SUB FX(I,M,V(),X(),Y()):
Y(I)=0 :: FOR K=M TO 2 STEP
1 :: Y(I)=(Y(I)+V(K))*X(I):: NEXT
K :: Y(I)=Y(I)+V(K):: SUBEND
2860 SUB ZP(Z,S,W)
2870 IF ABS(W)<1E8 THEN DIS
PLAY AT(Z,S):USING 430:W
ELSE DISPLAY AT(Z,S-1):USING
440:W
2880 SUBEND
2890 SUB CLR(AZ,EZ):: CALL
HCHAR(AZ,1,32,32*(EZ-AZ+1)):
SUBEND
```

Freundschaftstreffen in Belgien

Am 27.09.1987 hatten unsere belgischen Freunde zu einem internationalen Treffen der TI-User eingeladen, das, um es vorwegzunehmen, einen erstaunlichen Anklang fand und viele Besucher nach Grobendonk (Naehe Antwerpen) lockte.

U.a. unsere Freunde vom koelner Workshop und auch aus Bremen kamen sie mit fuenf Mann hoch. Auch wir Aachener wurden herzlichst aufgenommen und anderen Gruppen vorgestellt. Dabei ging es keineswegs formell zu. Es war ein richtiges Familientreffen, wo alte Freunde sich wiedertrafen und neue Freundschaften geknuepft wurden. Doch im Mittelpunkt wie konnte es anders auch sein stand unser TI-99/4A nebst Hard- u.Software. Auf eine Flaeche von ca. 120 Qum. hatten die Einlader ihre Geraete aufgebaut und gaben uns Einblick in ihren Wissens-stand. Erfreulicherweise beteiligten sich viele Jugendliche, die nicht nur Spiele zeigten. Angagiert setzten sie den TI zur Steuerrung ihrer LEGO-Kraehne ect. ein und freuten sich spitzbuebisch, als sie mit wenigen Handgriffen uns eine richtig funktionierende Analog-Uhr auf den Bildschirm zauberten und dafuer viel Beifall erhielten. Aber auch die Grossen hielten mit ihrem Koennen nicht hinterm Berg und versetzten manchen Besucher in Erstaunen. Hinter einigen Operater bildeteten sich wahre Menschentrauben und man musste schon regelrecht Glueck haben, um einen Blick auf den Monitor werfen zu koennen. Das dabei so manches Programm auch auf unsere Disketten landete, duefte weiter nicht verwunderlich sein. Insbesondere muessen wir uns bei unseren Bremer Freunden herzlichst bedanken, die uns reichlich mit Material versorgten. Da auch einige Haendler aus dem benachbarten Niederlande und aus Belgien praesent waren, wurde dieses Treffen zu einer grenzueberschreitende Demonstration der Verbundenheit mit der Erkenntnis, dass unser TI-99/4A nicht nur eine grosse Vergangenheit, sondern auch Zukunft hat.

Armann

BÖRSE BÖRSE BÖRSE

TI 99/4A mit RS 232 Schnittstelle ,Basickurs und Handbuch,Buch Programing Basic with TI Homecomputer gegen Gebot. bei: Hermann Josef Wirtz Artilleriestr.25 5170 Juelich Tel.:02461/54798

Verk. TI99/4A + Ex Basic Modul(incl Handbuch) + Recorder(Minerva KR 150) + Rec.Kabel + Joyst. Adapter + Assembler packet(Lohmann.Erweiterung und Software) + zahlreiche Listings + 15 C20 Cassetten(voll) an Meistbietenden(mind.350 DM) bei. Jens Koblin, Egenhofenstr. 37e. 8033 Planegg.(089) 859753

Verkaufe gegen Gebot, möglichst komplett: 2 Konsolen TI99/4A, 2 Laufwerke, 32K Erweiterung, V24 und Pio Karte, G-RAM-Karte, TI Joystiks und Software: Peter Voigt, Breslauer Str. 20, 4060 Viersen 11, Tel. 02162/56520

Nur ungern verzichtet man auf vertraute Routinen, die manchmal das Leben etwas erleichtern. In einer solchen Lage befindet man sich in Turbo Pascal, wenn man eine Prozedur sucht, die eine Joystickabfrage zur Verfuegung stellt.

Joystickabfrage in Pascal

Zwar sieht der Sprachumfang eine Tastaturabfrage vor, jedoch fuer Joysticks hilft diese bekanntlich wenig weiter. Um eine Joystickabfrage zu ermoeglichen benoetigt man zwei Dinge:
1. Informationen ueber Adressen die etwas mit der Abfrage zu tun haben, 2. Prozeduren die einen Zugriff auf den Speicher moeglich machen. Der zweite Punkt wurde kuerzlich schon von in der TI Revue (Quelle:Reiner Ziegler ,Turbo Pascal und Maschinensprache) behan delt. Auf das vorgestellte Modul memlib* soll hier auch zurueckgegrif fen werden. Somit waere nur noch der 1.Punkt zu klaeren.Nach mehr oder weniger langem Suchen findet man in der Literatur folgendes:
Adresse: >8374 , unter dieser Adresse muss der Tastaturmodus ab gelegt werden.

Wert: 00 gesamte Tastatur
01 linke Tastaturhaelfte und Joystick 1
02 rechte Tastaturhaelfte und Joystick 2

Andere Werte sind noch moeglich, hier jedoch nicht von Interesse.
>8376 , unter dieser Adresse ist die aktuelle vertikale Position des Joysticks (1 oder 2) abgelegt.

Wert: 04 Position: oben
00 : mitte
FC : unten

>8377 , unter dieser Adresse ist die aktuelle horizontale Position des Joysticks(1 oder 2) abgelegt.

Wert: 04 Position: rechts
00 mitte
FC links

Mit disen Informationen ausgestattet kann man nun das Modul z.B. so schreiben:

```
MODUL JOYSTICK;
PROCEDURE PEEK(ADD:INTEGER;VAR WERT:INTEGER);EXTERNAL;
PROCEDURE POKE(ADD:INTEGER;WERT:INTEGER);EXTERNAL;
PROCEDURE JOYSTICK(TM:INTEGER;VAR XW,YW:INTEGER);
```

```
CONST MADD = -31884;XADD = -31881;YADD = -31882;
BEGIN
   POKE(MADD,TM);
   PEEK(XADD,XW);
   PEEK(YADD,YW);
   IF XW = 252 THEN XW: = -4;
   IF YW = 252 THEN YW: = -4;
END;.
```

Der Aufruf aus einem Programm erfolgt nun etwa:

```
PROGRAM TEST;
VAR XWERT,YWERT:INTEGER;
PROCEDURE JOYSTICK(TM:INTEGER;VAR XW,YW:INTEGER);EXTER
NAL;
BEGIN
   JOYSTICK(1,XWERT,YWERT);
   WRITELN(XWERT,YWERT);
END.
```

Man erhaelt hier fuer Joystick 1(Tastaturmodus 1) die aus Ex.Basic bekannten Rueckgabewerte. Ein Nachteil dieser Prozedur wird jedoch recht schnell offenbar: verwendet man dieses Modul in einem Programm,so muss der Linker angewiesen werden, ausser dem Modul Joystick auch das Modul mit den Prozeduren Peek und Poke zuzubin den. Dies ist erstens etwas umstaendlich, zweitens besteht die Moeglich keit es zu vergessen. Dieser Nachteil ist jedoch schnell zu beheben. Man gehe folgendermassen vor: Zunaechst kompiliert man das Modul z.B. unter dem Namen Joyst auf Diskette. Dann laedt man das erhal tene Assemblerfile wieder in den Editor ein.
Am Anfang des Quelltextes findet man folgende Zeilen:

```
REF PEEK
CB$ B@ PEEK
REF POKE
 CC$ B@ POKE
```

Diese werden entfernt. Nun kan man die Assemblerroutinen von Reiner Ziegler(siehe Quellenangabe im Text) folgendermassen einbauen:

1. unter DEF JOYST folgende Zeile einbauen:AA BSS 32
2. am Ende (vor END) einfuegen:

```
 CB$
   Assembliertext von PEEK
 CC$
   Assembliertext von POKE
 END
```

Assembliert man den so entstandenen Quelltext z.B.unter dem Namen JOYST** ,so muss bei Verwendung mittels Linker nur dieses Modul zugebunden werden.

Ein frohes
Weihnachtsfest
und ein glückliches
Neues Jahr

*** TI-COMPUTER E.V. AACHEN ***

==========

Vereinsanschrift: Alfred Scheller
Alte Vaalser Str. 65
5100 Aachen
Tel. 0241/83325

Vereinskonto: Stadtsparkasse Aachen
Ktn.Nr.: 10040186
Blz: 39050000

Aufnahmeantrag

Name: _____
Vorname: _____
Straße: _____
Plz,Ort: _____
Geburtsdatum: _____
Telefon: _____
Beruf: _____

Bemerkungen/Interessensgebiet
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

Ich besitze folgende Konfigurationen:
(Bitte ankreuzen)

() TI-99/4A
() Kassettenrecorder
() Peripherie-Box
() Disketten-System
() RAM-Erweiterung
() serielle-paralelle Schnittstelle
() G-RAM Karte
() Pascal-System
() Sprachsynthesizer
() Drucker Marke _____
() Modul Extended-Basic
Programmiersprachen (bzw. Module) _____

() Ich habe Interesse an der Zeitung und möchte sie in Zukunft nach meinen Möglichkeiten mitgestalten.

Ihre Daten werden elektronisch gespeichert.
Ort, Datum: _____
Unterschrift: _____
(des neuen Mitgliedes bei Minderjährigen des Erziehungsberechtigten)

Thanks to 99'er:

Holger Kusch (from germany)

for scanning of this document and shared with us.

- Reworked by:

TI99 Italian User Club in the year 2015.

(info@ti99iuc.it)

Downloaded from www.ti99iuc.it

Die Redaktion bittet :

Um die Bearbeitungszeit dieser Ausgabe zu verkuerzen,moechten wir,Sie die Autoren,auffordern ein wenig daran mitzuarbeiten.

Da viele von Ihnen Artikel mit dem TI Writer oder einem anderen DIS/VAR 80 Programm schreiben ,koennten Sie den Text bitte in fol gender Form gestalten:

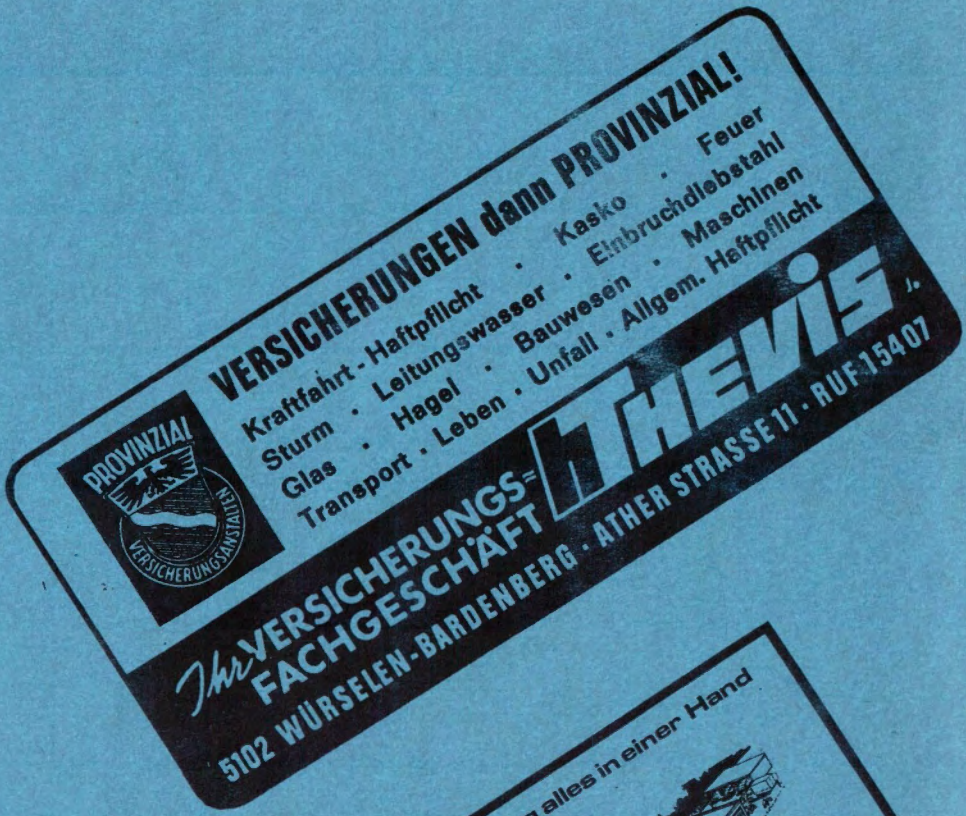
1. Text nicht mit dem Formatter bearbeiten
2. keine Druckersteuerzeichen verwenden
3. keine speziellen Zeichen des deutschen Zeichensatzes verwenden. dafuer: ae,ue,oe und ss

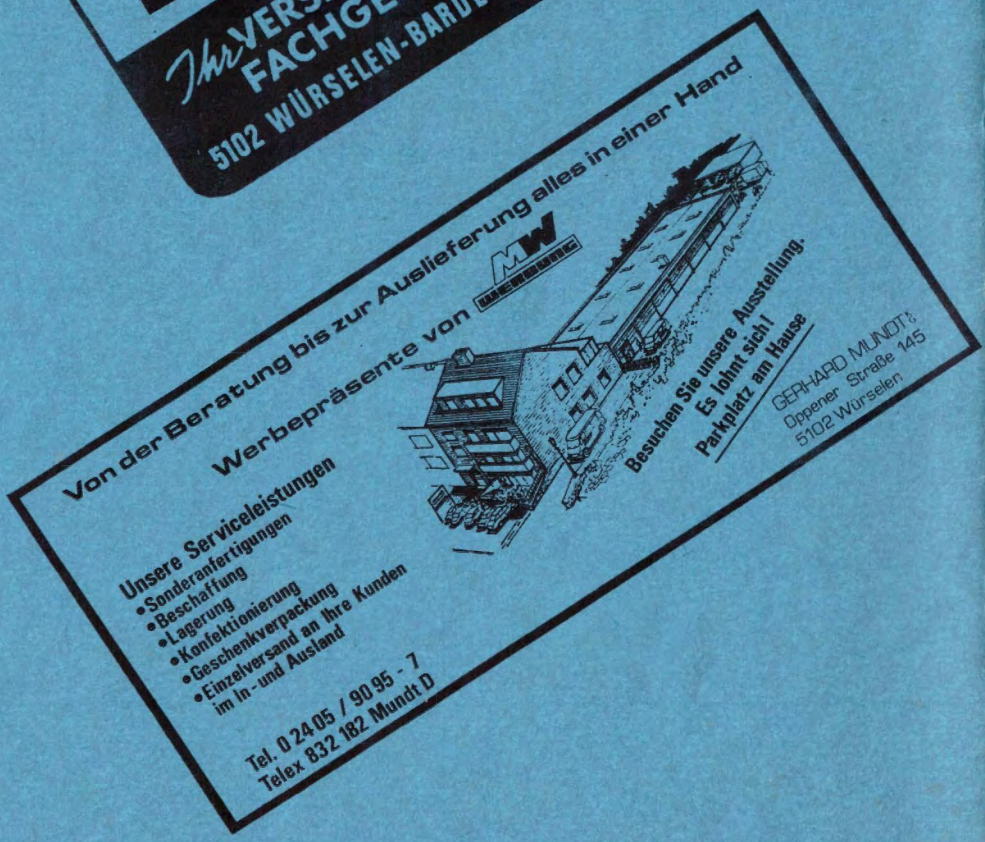
Wenn Sie uns dann eine Diskette mit dem Text und Ihrer Adresse,anstelle des ausgedruckten Textes zusenden,sparen Sie uns viel Arbeit.

Als Gegenleistung fuer Ihr Bemuehen,kopieren wir Ihnen alle Programme der Ausgabe,in der Ihr Artikel erscheint,auf Ihre Diskette.

Mit freundlichem Gruss,die Redaktion

Redaktionsschluss fuer die naechste Ausgabe: 15.02.1988


VERSICHERUNGEN dann PROVINZIAL!
Kraftfahrt · Haftpflicht · Kasko · Feuer
Sturm · Leitungswasser · Einbruchdiebstahl
Glas · Hagel · Bauwesen · Maschinen
Transport · Leben · Unfall · Allgem. Haftpflicht
PROVINZIAL
VERSICHERUNGSANSTALTEN
HEVIS
Ihr VERSICHERUNGS-FACHGESCHÄFT
5102 WÜRSELEN-BARDENBERG · ATHER STRASSE 11 · RUF 15407


Von der Beratung bis zur Auslieferung alles in einer Hand
Werbepräsente von
MW
Unsere Serviceleistungen
•Sonderanfertigungen
•Beschaffung
•Lagerung
•Konfektionierung
•Geschenkverpackung
•Einzelversand an Ihre Kunden im In- und Ausland
Tel. 0 24 05 / 90 95 - 7
Telex 832 182 Mundt D
Besuchen Sie unsere Ausstellung.
Es lohnt sich!
Parkplatz am Hause
GERHARD MUNDT
Oppener Straße 145
5102 Würselen