



UK EINSTEIN USER GROUP

NEWSLETTER

Volume 2 Number 1 November 1986

CONTENTS

**REGULARS**

Editorial.....	Page 1	Settling In
Adventurers Column.....	No column due	to house move this month
Letters.....	Back Page	Your page for comments
Cat.....	Page 9	Public Domain software

**REVIEWS**

UKEUG visit TATUNG UK.....	Page 5	A walk around Tatung
The Exhibition.....	Page 3	From the door

**FEATURES**

Writer.....	Page 10	A novel word processor in two parts
Screenplus & Picpen.....	Page 18	Linking the two for quicker buildup of pictures
Discs.....	Page 15	What they are, how to get the best out of them and how to recover lost data.
Allsorts.....	Page 19	The right sort for the job

**GAMES**

Merry Christmas.....	Page 8	Another Martin Classic
Missionaries & Cannibals.....	Page 20	Our first in LOGO

**UTILITIES**

Diary.....	Page 2	So you don't miss the next exhibition
Kims Club Helper.....	Page 5	Get your 10% right!
Auto Dial.....	Page 4	Communications for the ultra lazy

**HINTS & TIPS**

Using A Serial Printer.....	Page 19	Linking Serial printers to to existing programs.
Amtat and the Silicon Disc.....	Page 14	Getting AMTAT and the Silicon disc working together

EDITORIAL

This is now my second month at editing the newsletter and its late already!!! My first excuse is that we have been trying to catch up on the backlog of back issues required, caused by the response of people at the exhibition who, having seen what we are doing, joined. Talking of the Exhibition, I must thank all those members who gave up their time and came along to assist in the running and organisation, both on the day and beforehand. Their sterling work ensured that the day was a success. What a success too. Attendance topped the 3000 mark, according to Tatung's head counts. The software houses that attended were dissappointed that they had bought insufficient stocks with them as they ran out of all the popular software, in fact one of them disclosed that their sales exceeded £6000, and they were not the only people selling. None of the exhibitors had a chance to stop for lunch, so this had to be eaten standing up.

On to more mundane things. An upgrade to the Einstein to give it 256 graphic capabilities is being worked on, along with a replacement processor board to expand the system to cope with four or five programs at the same time. This is not simultaneous processing but holding all programs in memory and switching from one to the next without losing any data from any of them. Imagine playing your favourite stock control program and being asked to run off a memo to the M.D., with the new unit fitted you will just call up the wordprocessor and write it, then revert to the stock control exactly where you left off, WATCH THIS SPACE.

DIARY

ust so you don't miss the next exhibition Keith has written this diary program for you to type in. It will create its own data file for the days and months so when it has been run once you can delete the DATA statements.

```

10 PRINT CHR$(12):PRINT" D I A R Y ":D=365
20 DIM D$(D),M$(D):GOTO 200
30 PRINTCHR$(12):PRINT:PRINT"Hello ";N$:PRINTM1$:PRINT:PRINTM$(I)
40 PRINT:PRINT"Would you like to see the next seven days <Y or N>"
50 A$=INCH$:IF A$="Y" OR A$="y" THEN 310:ELSE IF A$="N" OR A$="n"
   THEN 59
55 GOTO 50
59 PRINT CHR$(12)
60 PRINT:PRINT"Do you have any messages for future dates.<Y or N>"
70 A$=INCH$:IF A$="y" OR A$="Y" THEN 110:ELSE IF A$="n" OR A$="N"
   THEN 195
75 GOTO 70
80 PRINT CHR$(12):INPUT"Please enter todays date in this format
   MM/DD.":T$:FOR I=0 TO D:IF D$(I)=T$THEN 90 ELSE NEXT
90 IF LEN(M$(I))>0 THEN M1$="Here are your messages for today :":
   ELSE M1$="I'm sorry but there are no messages for today :":
100 GOTO 30
110 PRINT CHR$(12):PRINT:PRINT"Please enter date required in this
   form MM/DD ":INPUT DD$
120 FOR I=0 TO D:IF D$(I)=DD$THEN 150:ELSE NEXT
130 PRINT"I'm sorry but I can not find this date ";DD$:PRINT:PRINT
   :INPUT"Press ENTER to try again : ";X:GOTO 110
150 PRINT:PRINT M$(I):PRINT:PRINT"Please enter your message now.:
   Press ENTER when finished. ":INPUTMM$:M$(I)=MM$
160 PRINT:PRINT:PRINT"To enter more Messages please press Y : To
   end press N"
170 A$=INCH$:IF A$="Y" OR A$="y" THEN 110:ELSE IF A$="N" OR A$="n"
   THEN 190
180 GOTO 170
190 OPEN"DIARY.DAT",FD$:PRINT#FD$:FOR I=0 TO D:PRINTD$(I);",";
   M$(I):NEXT:CLOSE
195 END
200 OPEN"DIARY.DAT",FD$:INPUT#FD$:FOR I=0 TO D:INPUTD$(I),M$(I):
   NEXT:CLOSE
205 IF N$=""THEN PRINT CHR$(12):PRINT"I'm sorry there are no
   messages today"
210 GOTO 80
300 PRINT"Here are your messages for the next seven days :":
305 F=I:FOR F=I TO I+8
310 FOR F=I+1 TO I+7:PRINT D$(F);" ";M$(F):NEXT F:GOTO 60
500 FOR I=0 TO D:READ D$(I):NEXT
510 DATA 01/01,01/02,01/03,01/04,01/05,01/06,01/07,01/08,01/09
520 DATA 01/10,01/11,01/12,01/13,01/14,01/15,01/16,01/17,01/18
530 DATA 01/19,01/20,01/21,01/22,01/23,01/24,01/25,01/26,01/27
540 DATA 01/28,01/29,01/30,01/31
550 DATA 02/01,02/02,02/03,02/04,02/05,02/06,02/07,02/08,02/09
560 DATA 02/10,02/11,02/12,02/13,02/14,02/15,02/16,02/17,02/18
570 DATA 02/19,02/20,02/21,02/22,02/23,02/24,02/25,02/26,02/27
580 DATA 02/28
590 DATA 03/01,03/02,03/03,03/04,03/05,03/06,03/07,03/08,03/09
600 DATA 03/10,03/11,03/12,03/13,03/14,03/15,03/16,03/17,03/18
610 DATA 03/19,03/20,03/21,03/22,03/23,03/24,03/25,03/26,03/27
620 DATA 03/28,03/29,03/30,03/31
630 DATA 04/01,04/02,04/03,04/04,04/05,04/06,04/07,04/08,04/09
640 DATA 04/10,04/11,04/12,04/13,04/14,04/15,04/16,04/17,04/18
650 DATA 04/19,04/20,04/21,04/22,04/23,04/24,04/25,04/26,04/27
660 DATA 04/28,04/29,04/30
670 DATA 05/01,05/02,05/03,05/04,05/05,05/06,05/07,05/08,05/09

```

680 DATA 05/10,05/11,05/12,05/13,05/14,05/15,05/16,05/17,05/18  
 690 DATA 05/19,05/20,05/21,05/22,05/23,05/24,05/25,05/26,05/27  
 700 DATA 05/28,05/29,05/30,05/31  
 710 DATA 06/01,06/02,06/03,06/04,06/05,06/06,06/07,06/08,06/09  
 720 DATA 06/10,06/11,06/12,06/13,06/14,06/15,06/16,06/17,06/18  
 730 DATA 06/19,06/20,06/21,06/22,06/23,06/24,06/25,06/26,06/27  
 740 DATA 06/28,06/29,06/30  
 750 DATA 07/01,07/02,07/03,07/04,07/05,07/06,07/07,07/08,07/09  
 760 DATA 07/10,07/11,07/12,07/13,07/14,07/15,07/16,07/17,07/18  
 770 DATA 07/19,07/20,07/21,07/22,07/23,07/24,07/25,07/26,07/27  
 780 DATA 07/28,07/29,07/30,07/31  
 790 DATA 08/01,08/02,08/03,08/04,08/05,08/06,08/07,08/08,08/09  
 800 DATA 08/10,08/11,08/12,08/13,08/14,08/15,08/16,08/17,08/18  
 810 DATA 08/19,08/20,08/21,08/22,08/23,08/24,08/25,08/26,08/27  
 820 DATA 08/28,08/29,08/30,08/31  
 830 DATA 09/01,09/02,09/03,09/04,09/05,09/06,09/07,09/08,09/09  
 840 DATA 09/10,09/11,09/12,09/13,09/14,09/15,09/16,09/17,09/18  
 850 DATA 09/19,09/20,09/21,09/22,09/23,09/24,09/25,09/26,09/27  
 860 DATA 09/28,09/29,09/31  
 870 DATA 10/01,10/02,10/03,10/04,10/05,10/06,10/07,10/08,10/09  
 880 DATA 10/10,10/11,10/12,10/13,10/14,10/15,10/16,10/17,10/18  
 890 DATA 10/19,10/20,10/21,10/22,10/23,10/24,10/25,10/26,10/27  
 900 DATA 10/28,10/29,10/30,10/31  
 910 DATA 11/01,11/02,11/03,11/04,11/05,11/06,11/07,11/08,11/09  
 920 DATA 11/10,11/11,11/12,11/13,11/14,11/15,11/16,11/17,11/18  
 930 DATA 11/19,11/20,11/21,11/22,11/23,11/24,11/25,11/26,11/27  
 940 DATA 11/28,11/29,11/30  
 950 DATA 12/01,12/02,12/03,12/04,12/05,12/06,12/07,12/08,12/09  
 960 DATA 12/10,12/11,12/12,12/13,12/14,12/15,12/16,12/17,12/18  
 970 DATA 12/19,12/20,12/21,12/22,12/23,12/24,12/25,12/26,12/27  
 980 DATA 12/28,12/29,12/30,12/31

# REVIEW OF NATIONAL U.K.E.U.G. EXHIBITION By Kandy Bate

Our first Exhibition of the National U.K.E.U.G. was held at the National Motorcycle Museum, Birmingham, on November 8, 1986 from 10 a.m. to 7 p.m. Your response was fabulous, with many of you travelling from the four corners of the British Isles! A conservative estimate of numbers who visited us was one thousand ... and all of us Einey users!

It was an interesting experience sitting at the front desk to realise that many Einey users had confused us with the "Einstein User" Magazine which is published by Tatung..and were unaware of the existence of our National User Group. However, this was soon corrected and you are now members.

We met for an early breakfast at 7 a.m. and we were then on our way to the National Motorcycle Museum. Although it was just after 8 a.m. a number of our exhibitors were already unloading their goodies. It was interesting putting faces to names and voices, and as we introduced ourselves we found that they were as warm, friendly and helpful as they usually were at the end of the telephone!

Among the exhibitors the people I had dealt with and who had been so helpful were Brian Thomas from Blagbrough & Hebblethwaite in Yorkshire, Dr. Mike Baylis and his assistant, Andy from the Business Computer Centre in Telford, Nigel and Ketan from Screens in Middlesex and, of course, Dr. Roy Clarke and David Bell from Tatung. The other software companies supporting us were Syntaxsoft, Surrey Software.

David West one of our members had a table near the entrance, and had programmed Einey to welcome everyone with a verbal message! Our own User Group had about 6 Eineys set up at the rear of the hall, and were kept very busy helping people with their problems, giving them demonstrations of what Einey could do and offering a vast library of software from the Public Domain on their already formatted discs, just for the asking. A few of us, manned the door, just so we could get to



know you as you came in, and explain to you that we were "Users" like yourselves who had got together to help each other sort out problems and exchange ideas. Many of you had confused us with the "Einstein User" magazine which is written and published by Tatung at a cost of £5.00 a year. Your enthusiasm to join us was overwhelming..a special welcome to you new members. Remember to keep your letters and articles rolling in. It is your Newsletter, and do send Chris Giles your gripes and programmes, and letters to include in future issues. Also your ideas for features and anything else you wish to know about. Remember our strength is in our numbers. A special welcome to Dr. Roy Clarke from Tatung. He not only gave his nephew an early Christmas present of a year's membership...he enrolled as a member himself...here's to a fruitful relationship, Roy. We'll all drink to that! Some of you who were so surprised to learn of our existence and took the forms hope with you, I do hope that now you realise we really exist you will have join us. We have formed this non-profit making scheme "for the user by the user". As we explained to you you do not pay a membership subscription to join. What you are paying for is the newsletter, production and distribution...12 issues a year..what more can one ask for for £15.00? We had originally hoped to open the doors at 10.00 a.m. and close at 19.00 hrs. But your enthusiasm was so great that many of you turned up well before 10 a.m. and we did actually start winding up around 5 p.m. as some of the software companies were 'cleaned out', and others were nearly so. Some of us who had friends visit us at the Exhibition whom we hadn't seen for years, found that they were so taken by Einey that they are now deciding to get themselves an Einey, and in one particular case, throw away his outmoded Acorn!! All in all..I can honestly say that taking part in the Exhibition was an extremely enjoyable and exciting experience. We can count on more support from the software companies for the future. All of them had nothing but praise for the success of our first venture. And Einey enjoyed the acclamation it rightly deserves as such an excellent and versatile micro.

To make the next one bigger and better tell us what YOU would like, and do write and let us know what you thought of the Exhibition. The first, of something we would hope to make into an annual event.

#### COMMUNICATIONS FOR THE ULTRA LAZY

Written in XBAS this should be fairly easy to modify for your own use.

```

5 REM AN AUTO-DIAL ROUTINE FOR AN          RTS LINE CONTROLLED MODEM
10 REM WRITTEN BY D.J.WEST AYLESBURY
20 RESTORE:DIM P$(39):IOM 5,0
30 FOR X=0 TO 39 STEP 1:READ P$(X):NEXT X
40 CLS:TCOL4,15:PRINT "                  DIRECTORY          "
   :TCOL15,4:PRINT
50 FOR X=20 TO 39 STEP 2:FMT 2,0:PRINT X-20;TAB(2);P$(X);TAB(20);
   X-19;TAB(2);P$(X+1):PRINT:NEXT X
60 TCOL4,15:PRINT "          PRESS 00 to 19 for OPTION          ":TCOL15,4
70 REM DIAL
80 GOSUB 260:
90 K=EVAL(INCH$(2))
100 IF K<0 OR K>19 THEN 40
110 IF K<>19 THEN 150
120 INPUT " Input Tel.No.(inc.spaces)";P$(K)
130 PRINT P$(K);" Is this correct ? (Y/N)":PRINT
140 AN$=INCH$:IF AN$<>"Y" THEN 120
150 PRINT " DIALING ";P$(K+20):OUT&11,&37:FOR LOOP=1 TO LEN (P$(K))
160 D$(MID$(P$(K),LOOP,1))
170 IF D$=" " THEN GOSUB230:PRINT " ";:NEXT
180 D=EVAL(D$):PRINT D;:IF D=0 THEN D=10
190 FOR IDS=1 TO 500:NEXT:GOSUB 240:REM IDS=Inter-digit space
200 NEXT
210 PRINT:PRINT"R to Redial  any other to continue"
```

```

220 AN$=INCH$:IF AN$<>"R" THEN CLS:END:ELSE OUT&11,&17:GOTO 40
230 FOR SP=1 TO 1000:NEXT:RETURN:REM SP=space between code & number
240 FOR PU=1 TO D:OUT&11,&17:GOSUB 250:OUT&11,&37:GOSUB 250:NEXT
:RETURN:REM PU=number of pulses per digit
250 FOR IPD=1 TO 50:NEXT:RETURN:REM IPD=inter-pulse delay/pulse
length
260 RETURN
270 DATA"0000","1111","2222","3333","4444","5555","6666","7777",
"8888","9999"
280 DATA"0909","0808","0707","0606","0505","0404","0303","0202",
"0101",""
290 DATA "RAQUEL","URSULA","CHARLOTTE","FRANCOISE","JENNY",
"JAMIE LEE","JOSELYN","FARRAH","PAM","SUSAN"
300 DATA "TOM","DICK","HARRY","HARRY","FRED","BERT","JOE","SAM",
"JIM","New No"

```

#### UKEUG VISIT TATUNG UK by Martin Page

The invitation was extended and we eagerly accepted. We couldn't miss the opportunity to see for ourselves the birth of our machines. The first part of the tour was around the television sections showing the circuit boards being assembled and tested. Then into the chassis and the product took on a recognisable shape.

One of the first things we noticed was the now typical Japanese emphasis on quality control. Testing was performed at almost every stage. The speed of the conveyor however was painfully slow. Whether this was Friday afternoon or just after lunch we can only guess.

Televisions take up most of the plant and we were shown the whole line, yes they do give each set a smack with a big hammer.

Then it was over to the computer area. It was explained how Tatung have, and can, make parts for competitors such as AM...AD and IBM. There they all were in various stages of completion, from empty shells to fully finished units on test. They were there but we didn't see any being made. Again was it Friday? We saw 100's boxed and ready to go so one thing we can conclude is that our machine is far from dead.

What was noticably conspicuous by its absence was the 256. We saw one in the display area but as this is the new super all singing all dancing machine just released I would have expected to see hundreds of them. We were shown just about everywhere even the overstocked warehouse but no 256's, on test, in boxes, or in bits. I haven't seen any in the shops. The dealers haven't got any to sell and we haven't recieved a review model yet.

Our appreciation is extended to our hosts for a pleasant and enlightening afternoons tour of their manufacturing plant.

#### KIMS CLUB HELPER

I don't know how many of you wives or other members of the family who run one of the many mail order club books, but the few that I have known seem to get really tied in knots trying to make sure that it all goes well. No More the late night reconciliation of books, just let them use your Einstein to make enough commission to get their own!

Written in BBCBASIC

```

10 REM
20 REM KIMS CLUB HELPER
30 REM
40 @%=&00020206
41 COLOUR143:COLOUR 1
50 CLS:PRINTTAB(18.0)"MENU"
60 PRINT"PRESS S FOR STEP
BY STEP"
70 PRINT"PRESS O FOR SHORT
PREDICTION"
80 PRINT"PRESS C FOR
CALCULATOR"
90 G$=GET$
100 IF G$="S" THEN PROCSTEP
:GOTO50
110 IF G$="O" THEN PROCONCE
:GOTO50
120 IF G$="C" THEN PROCADDUP
:GOTO 50

```

```

130 GOTO90
140 DEF PROCNCE
150 CLS
160 INPUT"WHAT WAS THE LAST
CORRECT BALANCE "
;BALANCE
170 INPUT"WHAT ARE THE WEEKLY
PAYMENTS "
;PAYMENT
180 IF PAYMENT>BALANCE THEN
PRINT"TWIT! THE PAYMENT IS
MORE THAN THE BALANCE
I'LL GO BACK TO THE START"
:FOR L=1TO6000:NEXT:RUN
190 PRINT"THE NUMBER OF "
;PAYMENT;" PAYMENTS IS"
;BALANCE/PAYMENT
200 IF BALANCE/PAYMENT<>
INT(BALANCE/PAYMENT)THEN
PRINT"THE LAST PAYMENT
WILL BE "
;((BALANCE/PAYMENT)-
INT(BALANCE/PAYMENT))*
PAYMENT
210 INPUT"HOW MANY'WHOLE'
PAYMENTS HAVE BEEN
MADE ";WKSPYD
220 PRINT"O.K. SO HERE IS WHAT
YOU SHOULD HAVE SO FAR"
230 TEMP1=BALANCE
240 OVER=0
245 PRINT"INITIAL BALANCE
";BALANCE
250 FOR L=1 TO WKSPYD-1
251 L$=LEFT$(STR$(L),2)
260 TEMP1=TEMP1-PAYMENT
270 IF TEMP1<0 THEN COLOUR 6
280 PRINTTAB(0);"AFTER
PAYMENT ";L$;" THE BALANCE
IS ";TEMP1
290 NEXT L
300 IF TEMP1-PAYMENT<0 THEN
COLOUR 6
310 PRINT"SO! AFTER THE LAST
PAYMENT THE BALANCE IS
";TEMP1-PAYMENT
315 W$=LEFT$(STR$(INT(BALANCE
/PAYMENT)),2)
316 WP$=LEFT$(STR$(WKSPYD),2)
320 COLOUR1:IF TEMP1-PAYMENT<0
THEN PRINT"IT APPEARS THAT
YOU OWE THE CUSTOMER
";ABS(TEMP1-PAYMENT):
PRINT"MAYBE YOU HAVE MADE
A MISTAKE,THE MAX NUMBER
OF WHOLE PAYEMENTS IS "
;W$;" SO ";WP$;" WHOLE
PAYMENTS ARE TO MANY"
330 COLOUR 13 : PRINT"PRESS
ANY KEY TO START AGAIN"
:COLOUR1
340 GET
350 ENDPROC

```

```

360 REM
370 REM_STEP BY STEP SECTION_
380 REM
390 DEF PROCSTEP
400 CLS:DIM A(100,3)
410 P%=0:FLAG%=0
420 PRINT"THIS PROGRAMME WILL
GO THROUGH A COMPLETE
ACCOUNT WITH YOU FROM
BEGINNING TO END"
430 INPUT"WHAT WAS THE BALANCE
RIGHT AT THE BEGINNING
OF THIS ACCOUNT"
;A(0,1):A(0,2)=0:A(0,3)=0
440 INPUT"WHAT WERE THE
PAYMENTS ON THIS ";
PAYMENT:TEMP2=PAYMENT
: PRINT
450 PRINT"GOOD! NOW ALL YOU DO
IS PRESS P TO MAKE A
PAYMENT, PRESS E TO ERASE
THE LAST OPERATION,PRESS B
TO ADD TO THE BALANCE,
PRESS C TO CHANGE THE
WEEKLY PAYMENT, PRESS %
TO CLEAR THE
PERCENTAGETOTAL, PRESS A
TO GO TO THE CALCULATOR,"
451 PRINT"PRESS L FOR PRINTER
OUTPUT, PRESS H FOR HELP"
460 P$= LEFT$(STR$(P%),2)
470 PRINTTAB(0);P$ TAB(3);
"";A(P%,1) TAB(10);"";
A(P%,2) TAB(17);"";A(P%,3)
480 G$=GET$
481 IF G$="L" THEN PROCPRNTER
490 IF G$="P" THEN P%=P%+1
PROCPAY
500 IF G$="B" THEN P%=P%+1:
PROCADBAL
510 IF G$="C" THEN PROCADJPAY
520 IF G$="%" THEN FLAG%=1
530 IF G$="A" THEN PROCADDUP:
PROCOUTPUT
540 IF G$="S" THEN CLEAR:
ENDPROC
550 IF G$="E" AND P%=0 THEN
CLEAR:ENDPROC
560 IF G$="E" THEN P%=P%-1:
PROCOUTPUT:PRINT" CURRENT
PAYMENT IS STILL ";PAYMENT
:PRINTTAB(0,VPOS-1);"";
561 IF G$="H" THEN PROCHELP
570 GOTO 480
580 DEF PROCPAY
590 A(P%,2)=PAYMENT
600 A(P%,1)=A(P%-1,1)-A(P%,2)
610 IF FLAG%=0 THENA(P%,3)=
A(P%-1,3)+A(P%,2)
ELSE A(P%,3)=A(P%,2):
FLAG%=0
620 X=A(P%,3)/8:X2=A(P%,3)/10
630 P$=LEFT$(STR$(P%),2)

```

```

635 IF A(P%,1)<0 THEN COLOUR6
640 PRINTTAB(0);P$ TAB(3);
    " ";A(P%,1) TAB(10);" ";
    A(P%,2) TAB(17);" ";
    A(P%,3) TAB(25);" ";X
    TAB(32);" ";X2
645 COLOUR 1
650 ENDPROC
660 DEF PROCADBAL
670 INPUT"WHAT DO YOU WANT TO
    ADD TO THE BALANCE " EXTRA
680 A(P%,1)=A(P%-1,1)+EXTRA:
    A(P%,2)=0:A(P%,3)=A(P%-1,3)
690 PROCOUTPUT
700 ENDPROC
710 DEF PROCADJPAY
720 INPUT"WHAT IS THE NEW
    WEEKLY PAYMENT "PAYMENT
730 PROCOUTPUT
740 ENDPROC
750 DEF PROCOUTPUT
760 CLS
770 FOR L%= 0 TO P%
780 P$=LEFT$(STR$(L%),2)
781 IF A(L%,1)<0 THEN COLOUR6
790 IF A(L%,2)=0 THEN
    P$=P$+"*"
800 PRINTTAB(0);P$ TAB(3);
    " ";A(L%,1) TAB(10);" ";
    A(L%,2) TAB(17);" ";
    A(L%,3) TAB(25);" ";
    A(L%,3)/8 TAB(32);" ";
    A(L%,3)/10
810 NEXT L%
815 COLOUR 1
820 ENDPROC
830 DEF PROCADDUP
840 CLS
850 PRINTTAB(10,0)"CALCULATOR"
860 PRINT"PRESS A TO ADD"
870 PRINT"PRESS S TO SUBTRACT"
880 PRINT"PRESS M TO MULTIPLY"
890 PRINT"PRESS D TO DIVIDE"
900 PRINT"PRESS T TO GO BACK"
910 G$=GET$
920 IF G$="A" THEN 980
930 IF G$="S" THEN 1090
940 IF G$="M" THEN 1170
950 IF G$="D" THEN 1250
960 IF G$="T" THEN ENDPROC
970 GOTO 910
980 REM*** ADD *****
990 CLS
1000 PRINT"INPUT 0 TO TOTAL"
1010 PRINT
1020 C=0
1030 INPUT"INPUT THE FIRST
    FIGURE ";A
1040 C=C+A
1050 INPUT"INPUT NEXT FIGURE ";
    B
1060 IF B=0 THEN PRINT
    "THE TOTAL IS ";C:
    PRINT"12\% OF IT IS ";C/8:
    PRINT"10% OF IT IS ";C/10:
    GOTO 1330
1070 C=C+B
1080 GOTO 1050
1090 REM***SUBTRACT*****
1100 CLS
1110 INPUT"FIRST NUMBER ";A
1120 INPUT"SECOND NUMBER ";B
1130 PRINTA;"-";B;"=";A-B
1140 PRINT"12\% OF THIS IS ";
    (A-B)/8
1150 PRINT"10% OF THIS IS ";
    (A-B)/10
1160 GOTO 1330
1170 REM*** MULTIPLY *****
1180 CLS
1190 INPUT"FIRST NUMBER ";A
1200 INPUT"SECOND NUMBER ";B
1210 PRINTA;"*";B;"=";A*B
1220 PRINT"12\% OF THIS IS ";
    (A*B)/8
1230 PRINT"10% OF THIS IS ";
    (A*B)/10
1240 GOTO 1330
1250 REM*** DIVIDE *****
1260 CLS
1270 INPUT"FIRST NUMBER ";A
1280 INPUT"SECOND NUMBER ";B
1290 PRINTA;"/";B;"=";A/B
1300 PRINT"12\% OF THIS IS ";
    (A/B)/8
1310 PRINT"10% OF THIS IS ";
    (A/B)/10
1320 GOTO1330
1330 PRINT"PRESS ANY KEY TO GO
    BACK"
1340 Z$=GET$
1350 ENDPROC
1360 REM _____
1370 REM _____HELP_____
1380 REM _____
1390 DEF PROCHELP
1400 CLS
1410 PRINTTAB(17),"HELP":
    PRINT"";
1420 PRINT"THE CONTROL KEYS ARE
    AS FOLLOWS:-":PRINT"";
1430 PRINT"P_____THIS KEY
    AUTOMATICALLY MAKES A
    PAYMENT AND LISTS THE NEW
    STATE OF THE ACCOUNT"
1431 PRINT"E_____THIS ERASES THE
    LAST OPERATION"

```



```

440 PRINT"B__THIS KEY ALLOWS
YOU TO ADD TO THE
BALANCE.THE FIGURE YOU
INPUT WILL BE ADDED TO
THE OLD BALANCE AND THE
ACCOUNTRELISTED WITH THE
NEW BALANCE, THIS
OPERATION WILL BE MARKED
WITH A * IN THEOPERATION
NUMBER COLUMN"
1450 PRINT"C__THIS KEY ALLOWS
YOU TO CHANGE THE
WEEKLY PAYMENT.THE FIGURE
YOU INPUT WILLBE USED AS
THE NEW PAYMENT,THIS CAN
BE DONE IMMEDIATLY
FOLLOWING ANY CHANGE IN
THE BALANCE,BEFORE ANY
FURTHER PAYMENTS ARE MADE"
1460 PRINT"%__THIS KEY, WHEN
PRESSED ZEROS THE
PERCENTAGE TOTAL AND THE
12\% AND 10% COLUMNS,
EFFECTIVELY RESTARTING THE
COMMISSION CALCULATIONS"
1470 PRINT:PRINT"PRESS ANY KEY
TO CONTINUE";
1480 Z$=GET$
1490 CLS
1500 PRINT"A__THIS PUTS YOU
INTO THE CALCULATOR
PROGRAMME.UPON RETURNING
FROM THE CALCULATOR
YOUR ACCOUNT WILL BE
RELISTEDTO THE POINT YOU
LEFT OFF"
1510 PRINT"S__THIS WILL
EFFECTIVELY RERUN THE
PROGRAMME FROM THE START
AND YOUR ACCOUNT
WILL BE LOST"
1520 PRINT"H__WILL BRING YOU
TO THIS FUNCTION"
1521 PRINT"L__SENDS ACCOUNT TO
THE PRINTER"
1530 PRINT"THE COLUMNS ARE AS
FOLLOWS:-":PRINT
1540 PRINT"*THE FIRST COLUMN
IS SIMPLY THE
OPERATION NUMBER"
1550 PRINT"*THE SECOND COLUMN
IS THE CURRENT
BALANCE"
1560 PRINT"*THE THIRD COLUMN
IS THE PAYMENT WHICH HAS
JUST BEEN SUBTRACTED FROM
COLUMN 2"
1570 PRINT"*THE FORTH COLUMN
IS THE TOTAL PAID
SINCE, EITHER THE
BEGGINNING OR FROM THE
LAST TIME THE % KEY WAS
USED"
1580 PRINT"*THE FIFTH COLUMN
IS 12\% OF COLUMN 3"
1590 PRINT"*THE LAST COLUMN
IS 10% OF COLUMN 3"
1600 PRINT"If you can't
remember write it down"
1610 PRINT"PRESS ANY KEY TO
RETURN ";
1620 Z$=GET$
1630 PROCOUTPUT
1640 ENDPROC
1700 REM*PRINTER OUTPUT**
1710 REM=====
1720 DEF PROCPRNTER
1730 CLS
1735 *OPT 2
1740 PROCOUTPUT
1750 *OPT 0
1755 PROCOUTPUT
1760 ENDPROC

```

# MERRY CHRISTMAS

A Christmas offering from all at UKEUG by Martin Page

```

210 SHAPE 150,"011F38362C5C7880 80785C2C36381F01"
220 CLS32:TCOL12:GCOL6:BCOL1:ORIGIN0,0
230 FORJ=0TO15:PRINTCHR$(151);CHR$(150);:PLOT8+16*J,184 :NEXTJ
240 FORJ=1TO22:K=-2*(INT(J/2)-J/2):PRINT@0,J;CHR$(151-K);
245 PRINT@31,J;CHR$(150+K);:NEXTJ
250 FORJ=0TO10:PLOT7,183-16*J:PLOT248,183-16*J:NEXTJ
260 FORJ=0TO14:PRINTCHR$(150);CHR$(151);:PLOT7+16*J,7:NEXTJ
270 PRINT@30,23;CHR$(150);:SPRITE30,247,8,12,151:PLOT246,7
350 READA,B :FORJ=1TO48: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
360 READA,B :FORJ=1TO31: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
370 READA,B :FORJ=1TO34: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
380 READA,B :FORJ=1TO35: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
390 GCOL15
400 READA,B :FORJ=1TO12: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
410 READA,B :FORJ=1TO11: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
420 READA,B :FORJ=1TO12: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
430 READA,B :FORJ=1TO12: READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ

```

```

440 READA,B :FORJ=1TO21:  READC,D:DRAWA,BTOC,D :A=C:B=D :NEXTJ
999 GOTO999
1000 DATA20,133,30,137,40,142,50,148,60,155,72,165,62,160,54,153,
    44,141,50,143,60,146,70,151,80,158,84,161
1010 DATA80,160,70,155,65,153,60,146,56,142,59,140,70,141,78,146,
    70,145,69,143,70,140,80,141,85,145,82,140
1020 DATA86,144,90,144,94,145,90,140,94,144,100,144,105,145,103,
    143,101,140
1030 DATA110,141,118,145,117,137,113,132,105,129,95,126,88,127,87,
    129,91,131,100,134,120,139,140,142
1040 DATA50,118,60,120,66,124,65,128,60,129,53,128,46,125,40,120,
    36,110,38,103,42,100,48,100,60,100
1050 DATA72,124,60,100,67,107,73,109,74,108,70,102,80,109,76,103,
    80,107,84,106,91,108,88,103,97,106,105,111,99,107
1060 DATA104,106,104,104,100,102,96,104
1070 DATA155,116,108,113,113,113,119,123,106,101,109,100,114,103,
    120,109,116,101,120,107,124,108,125,106
1075 DATA123,101,125,106,129,107
1080 DATA132,105,130,101,135,102,143,106,147,107,150,106,147,107,
    143,106,140,102,140,101,147,103,152,107,150,102
1090 DATA160,105,170,110,163,107,170,105,171,104,168,101,159,102
1100 DATA54,56,58,61,61,73,59,84,84,90,59,84,61,73,55,72,51,69,55,
    72,61,73,64,71,67,68,72,70,70,63
1110 DATA75,69,79,68,85,70,82,68,80,65,82,62,85,61,89,62,91,65,88,
    69,95,70,94,61,98,69,101,70,103,66,101,61,103,66
1120 DATA107,69,110,68,109,61,112,62
1200 DATA165,90,165,70,170,65,185,65,190,70,190,90,185,90,185,72,
    183,70,172,70,170,72,170,90,165,90
1210 DATA197,90,197,65,202,65,202,73,210,65,217,65,205,77,218,90,
    212,90,202,80,202,90,197,90
1220 DATA149,50,149,25,174,25,174,30,154,30,154,35,169,35,169,40,
    154,40,154,45,174,45,174,50,149,50
1230 DATA180,50,180,30,185,25,200,25,205,30,205,50,200,50,200,32,
    198,30,187,30,185,32,185,50,180,50
1240 DATA212,45,212,29,216,25,233,25,237,29,237,36,228,36,228,32,
    233,32,232,30,219,30,217,32,217,43,219,45,231,45
1250 DATA233,43,233,38,237,38,237,45,232,50,217,50,212,45

```

#### CAT

The Cataloguing suite available from the P.D.Library has been developed over a number of years and the full history to date comes with the programs on disc. It comprises 5 files, four of which are programs and the last is a dummy file to start the catalogue off.

The four programs are 1)CAT.COM 2)CAT2.COM 3)CATP.COM 4)UDCAT.COM

CAT.COM Allows a DIR type search of the catalogue. i.e. typing CAT \*.XBS will list all the files on catalogue with the extension XBS

CAT GD\*.\* will list all the files on catalogue starting with GD

CAT2.COM lists all files with associated discs

CATP.COM lists all files to the printer.

UDCAT.COM updates the master catalogue.

To use you must first NAME each disc. To do this save a filename from DOS. i.e. SAVE 0 -MYDISC.001 This will save a file of zero length

on the disc with the Directory entry -MYDISC.001 The first character should be a '-' so as to force the disc names to be sorted to the top of the list. (I forgot to mention that all files are sorted alphabetically). Then by using UDCAT in one drive and the disc to be catalogued in another drive the program will take all the Directory entries and sort them adding and deleting files as appropriate. Easy.

A regular run through each of your discs will reveal where you have the same program on more than one disc, enabling you to release space and you will always be able to locate a given program by typing CAT 'programe.ext' and hey presto it will appear on the screen.

Available from UKEUG, send a formatted disc with the programs required and we will send you by return (as soon as we can!) those programs.

WRITER

This is a novel Word Processor written in XBAS with all the facilities you would require, but with the added facility of restructuring the printer output so that it is proportionally spaced as in handwriting. This program is being published in two sections so look out for next months issue to complete.

```

5 DB$="OFF":IOM4,0:IOM5,0
10 PL=60:BL=6:WTH=960:NL=0:F$="<untitled>":DIMLTR$(221),LTR(221),
  S(255),TEXT$(255),TEXT2$(255)
11 SHAPE177,"20 20 20 20 20 20 00 FF 78 CC 84 84 CC 78 00 FF"
12 SHAPE179,"BC BC BC BC BC BC BC BC F4 F4 F4 F4 F4 F4 F4"
15 FORT=0TO3:LTR(146+T)=0:LTR$(146+T)=CHR$(177+T):NEXTT
20 F$="<untitled>"
45 RESTORE50000
50 FORK=1TO95:READA:LTR(K)=A:NEXTK
55 RESTORE50050
60 FORK=1TO95:READA$:LTR$(K)=A$:NEXTK
70 GOTO10000
1000 REM***PRINT TEXT***
1005 FORN=1TONL:PRINT#1;CHR$(13);CHR$(10);
1006 IF KBD$="Q"THENPRINT:PRINT#0;"Printing aborted.":
  FORT=1TO500:NEXTT:GOTO10000
1007 IFLEN(TEXT$(N))<1THENPRINT#1;"":GOTO1096
1010 DB$="OFF":NB=0:FORK=1TOLEN(TEXT$(N)):
  Z=LTR(ASC(MID$(TEXT$(N),K,1))-31):
  IFMID$(TEXT$(N),K,1)="3"THENDB$="ON"
1012 IFMID$(TEXT$(N),K,1)="4"THENDB$="OFF"
1015 IFDB$="ON"THENZ=Z+Z
1020 NB=NB+Z:NEXTK
1025 IF KBD$="Q"THENPRINT:PRINT#0;"Printing aborted.":
  FORT=1TO500:NEXTT:GOTO10000
1030 IFRIGHT$(TEXT$(N),1)="p"THEN1040
1035 IFNB<WTHTHENGOSUB3000:ELSE1040
1036 NB=WTH
1040 NB2=INT(NB/256):NB1=NB-(NB2*256)
1041 PRINT#1;CHR$(27);"L";CHR$(NB1);CHR$(NB2);
1045 FORK=1TOLEN(TEXT$(N))
1050 Z2=ASC(MID$(TEXT$(N),K,1))-31:LTR$=LTR$(Z2)
1055 IFASC(LTR$)=177 THEN UL$="ON":GOTO1090
1056 IFASC(LTR$)=178 THEN UL$="OFF":GOTO1090
1057 IFASC(LTR$)=179 THEN DB$="ON":GOTO1090
1058 IFASC(LTR$)=180 THEN DB$="OFF":GOTO1090
1060 FORT=1TOLTR(Z2)
1070 A=VAL(MID$(LTR$,((T-1)*3)+1,3))
1072 IFUL$="ON"AND(A/2)=INT(A/2)THENA=A+1
1075 IFA=9THENA=17
1076 Z=1:IFDB$="ON"THENZ=2
1078 FORZ1=1TOZ
1080 PRINT#1;CHR$(A);
1082 NEXTZ1
1083 NEXTT
1085 IFS(K)=0THEN1090
1086 FORQ=1TOS(K):PRINT#1;CHR$(0);:NEXTQ
1090 NEXTK
1095 FORQ=1TO255:S(Q)=0:NEXTQ:IF(N/PL)=INT(N/PL)THEN1150
1096 NEXTN
1100 PRINT#0
1110 PRINT:PRINT"Repeat?":A$=INCH$:IFA$="Y"ORAS$="y"THEN1000
1120 IF A$<>"N"ANDA$<>"n"THEN 1110:ELSE10000
1150 IFBL>0THENPRINT#1;MUL$(CHR$(10),BL);:GOTO1096
1160 PRINT#0:PRINT:PRINTCHR$(7);"CHANGE PAPER!":
  PRINTCHR$(7);"Press any key when changed."
1170 FORQ=1TO1000:IFKBD>0THEN1096

```

```

1180 NEXTQ:PRINTMUL$(CHR$(7),5);:GOTO1170
2000 REM***ENTER LINE***
2010 NC=0:NB=0:EDIT=1
2040 PRINT:PRINT"<";L;">";
2050 A$=INCH$:IFASC(A$)=13THENRETURN
2055 IFNC=0THENTEXT$(L)=TEXT$(L)
2060 IFASC(A$)<32ORASC(A$)>126THEN2400
2070 Z=LTR(ASC(A$)-31):IFDB$="ON"THENZ=Z+Z
2080 NC=NC+1:NB=NB+Z:TEXT$(L)=TEXT$(L)+A$
2100 PRINTA$;:IFNC<245ANDNB<(WTH-50)THEN2050
2110 PRINTCHR$(7);:IFNB>WTHORNC=255THEN2130:ELSE2050
2130 IFNB>WTH THENTEXT$(L)=LEFT$(TEXT$(L),NC-1)
2140 PRINT:PRINT"Line full!":FORT=1TO500:NEXTT:RETURN
2150 RETURN
2200 IFNC=0THEN2050
2205 B$=RIGHT$(TEXT$(L),1)
2210 IFB$="3"THENDB$="OFF"
2220 IFB$="4"THENDB$="ON"
2230 Z=LTR(ASC(B$)-31):IFDB$="ON"THENZ=Z+Z
2240 NB=NB-Z:NC=NC-1:TEXT$(L)=LEFT$(TEXT$(L),NC):GOTO2100
2300 IFEDIT=LEN(EDIT2$)+1THENEDIT$="OFF"
2310 GOTO2060
2400 A=ASC(A$)
2410 IFA=179THENDB$="ON":GOTO2070
2420 IFA=180THENDB$="OFF":GOTO2070
2425 IFA=177ORA=178ORA$="p" THEN2070
2430 IFA=25THEN2200
2435 IFA=4ANDNC=0THENTEXT$(L)=""
2440 IFA=4ANDEDIT$="ON"THENA$=MID$(EDIT2$,EDIT,1):
EDIT=EDIT+1:GOTO2300
2450 GOTO2050
3000 REM***FILL OUT LINE***
3005 NS=0
3010 FORQ=1TOLEN(TEXT$(N)):IFMID$(TEXT$(N),Q,1)<>" "THEN3030
3020 IFRND(2)=0THENS(Q)=S(Q)+1:NS=NS+1
3030 IFNS+NB=WTH THENRETURN
3040 NEXTQ:GOTO3010
4000 REM***LOAD FILE***
4010 PRINT:PRINT"Enter file name:";X=POS(1):Y=POS(2):
PRINT:PRINT"<c/r for menu>":PRINT"<'*.'for.TXT files>"
4020 PRINT@X,Y;"";:INPUTA$:IFLEN(A$)<1 THEN10000
4025 IFLEN(A$)>16THENA$=RIGHT$(A$,LEN(A$)-16)
4030 IFA$<>"*."THEN4050
4040 CLS:DIR"*.*.TXT":GOTO4010
4050 IFRIGHT$(A$,4)<>".TXT"THENA$=A$+".TXT"
4060 ONERRGOTO4110
4070 OPENA$,FD$:N=1:TEXT$(1)="" :INPUT&FD$:ONEOF:GOTO4100
4075 PRINT@0,Y+4;"Loading line:"
4080 B$=INCH$:IFASC(B$)=13THENN=N+1:TEXT$(N)="" :B$=INCH$:GOTO4080
4085 PRINT@13,Y+4;N
4090 TEXT$(N)=TEXT$(N)+B$:GOTO4080
4100 NL=N-1:CLOSE:INPUT&0:PRINTMUL$(CHR$(10),4);"Text loaded.":
FORT=1TO500:NEXTT:F$=A$:GOTO10000
4110 PRINTMUL$(CHR$(10),4);"No file.":FORT=1TO500:NEXTT:GOTO4010
4500 REM***SAVE TEXT***
4505 IFNL<1THEN10000
4507 IFF$<>"<untitled>"THEN4530
4510 PRINT:PRINT"Enter filename:";X=POS(1):Y=POS(2):
PRINT"<c/r for menu>":PRINT@X,Y;""
4520 INPUTA$:IFLEN(A$)<1THEN10000
4525 IFLEN(A$)>3ANDRIGHT$(A$,4)<>".TXT"THEN4510
4526 IFLEN(A$)>12THEN4510
4527 F$=A$

```



```

530 IFRIGHT$(F$,4)<>".TXT"THENF$=F$+".TXT"
540 ONERRGOTO4580
550 OPENF$,FD$:CLOSE
560 PRINT:PRINT"Filename in use.":PRINT"Write over(Y/N)?"
565 B$=INCH$:IFB$="Y"ORB$="y"THEN4580
570 IFB$="N"ORB$="n"THEN4510
575 GOTO4565
580 CREATEF$,FD$
590 FORT=1TONL:PRINTFDS;TEXT$(T):NEXTT
600 CLOSE
610 PRINT:PRINT"Text saved.":FORT=1TO500:NEXTT:GOTO10000
000 REM***SET LINE WIDTH***
010 CLS:PRINT:OPEN"LINEWTHS.DAT",FD$
020 ON EOF GOTO5060
030 PRINT"Details";TAB(21);": ":"Line width":
    PRINTMUL$("-",20);": ":"-----"
035 PRINT"Present setting      ":";WTH
040 INPUTFDS;A$:INPUTFDS;B$
050 PRINTA$;TAB(21);": ":"B$:GOTO5040
060 CLOSE:PRINT:INPUTE0
070 PRINT"Enter line width(max.20 chr.s)-":INPUTA$:
    IFLEN(A$)<1THEN10000
075 A=VAL(A$):IFA<1ORA>960THENPRINT"Illegal entry!":
    FORT=1TO500:NEXTT:GOTO5060
080 WTH=A:PRINT:PRINT"Save for future reference?":
    A$=INCH$:IFA$="N"ORA$="n"THEN10000
090 IFA$<>"Y"ANDA$<>"y"THEN5080
100 PRINT:PRINT"Enter details(max.20 chr.s)-":
    INPUTA$:IFLEN(A$)>20ORLEN(A$)<1THEN5100
110 APPEND"LINEWTHS.DAT",FD$:PRINTFDS;A$:PRINTFDS;STR$(WTH):
    CLOSE:PRINT"Details saved.":FORT=1TO500:NEXTT:GOTO10000
5500 REM***SET PAGE LENGTH***
5510 CLS:PRINTMUL$(" ",20);": Page : No.of":
    PRINT"Details";TAB(21,32);": Length :Blanks"
5520 PRINTTAB(21,45);": ":"TAB(30);": ":"TAB(38):PRINT
5530 PRINT"Present setting      ":";PL;TAB(30,32);": ":";BL
5540 OPEN"SETPAGE.DAT",FD$:ONEOFGOTO5560
5550 INPUTFDS;A$:INPUTFDS;B$:INPUTFDS;C$:
    PRINTA$;TAB(21);": ":"B$;TAB(30);": ":";C$:GOTO5550
5560 CLOSE:PRINT:INPUTE0
5570 PRINT"Enter page length,blanks(c/r to leave)-":
    INPUTA$,B$:IFLEN(A$)<1ORVAL(A$)=0THEN10000
5580 PL=VAL(A$):BL=VAL(B$):PRINT"Save for future reference?":
    C$=INCH$:IFC$="N"ORC$="n"THEN10000
5590 IFC$<>"Y"ANDC$<>"y"THEN5580
5600 PRINT:PRINT"Enter details(max.20chr.s)-":
    INPUTC$:IFLEN(C$)<1THEN10000
5610 IFLEN(C$)>20THEN5600
5620 APPEND"SETPAGE.DAT",FD$:PRINTFDS;C$:PRINTFDS;STR$(PL):
    PRINTFDS;B$:CLOSE:PRINT"Details saved."
5630 FORT=1TO500:NEXTT:GOTO10000
5999 END
6000 REM***TITLE TEXT***
6010 PRINT:PRINT"Enter title(max.8 chr.s,c/r to leave)-":
    INPUTA$:IFLEN(A$)<1THEN10000
6020 IFLEN(A$)>8THEN6010
6030 F$=A$:GOTO10000
0000 REM***MAIN MENU***
0001 OFFERR:GOSUB10005
0002 GOTO10029
0003 GOTO10029
0005 CLS:PRINTMUL$(" ",12);"Text:";F$
0010 PRINT"No.of lines ":";NL:PRINT@20,1;"Free lines ":";255-NL

```

```

10020 PRINT"Page width  :";WTH;TAB(21,32);"Page length:";PL
10025 PRINT"No.of blanks:";BL;TAB(21);"Bits left  :";SIZE*8:RETURN
10029 PRINT:PRINT"Main Menu":PRINT
10030 PRINT"1.Edit present text":PRINT"2.Start new text":
PRINT"3.Load text":PRINT"4.Save text"
10040 PRINT"5.Set line width":PRINT"6.Set page length":
PRINT"7.Title text"
10045 PRINT"8.Print text":PRINT"9.End program":
PRINT:PRINT"Enter choice:";
10050 A$=INCH$:IFASC(A$)<49ORASC(A$)>58THEN10050
10060 PRINTA$:A=VAL(A$):ONAGOTO10200,10100,4000,4500,5000,5500,
6000,1000,10070
10070 PRINT:END
10100 IFNL=0THEN10200
10110 PRINT:PRINT"Are you sure(Y/N)?";
10120 A$=INCH$:IFA$="Y"ORA$="y"THEN10150
10130 IFA$="N"ORA$="n"THEN10000
10140 GOTO10120
10150 FORQ=1TO255:TEXT$(Q)="":TEXT2$(Q)="":NEXTQ:NL=0
10200 REM***EDIT TEXT***
10210 GOSUB10005
10220 PRINT:PRINT"Edit menu":PRINT:PRINT"0.Return to Main Menu":
PRINT"1.Edit line":PRINT"2.Delete line(s)"
10230 PRINT"3.Write next line":PRINT"4.Insert line":
PRINT"5.Move lines":PRINT"6.Display lines":
PRINT"7.Search/replace string"
10240 PRINT"8.Free write":PRINT"9.Sort lines":
PRINT:PRINT"Enter choice";
10250 A$=INCH$:IFASC(A$)<48ORASC(A$)>57THEN10250
10260 A=VAL(A$):ONAGOTO10500,11000,12000,12500,13000,14000,
14500,15500,17000
10270 GOTO10000
10500 REM***EDIT LINE***
10510 CLS:GOSUB10005
10520 PRINT:PRINT:PRINT"Enter line no.to edit:":
INPUTA$:IFLEN(A$)<1THEN10200
10530 L=VAL(A$):IFL=0ORL>NLTHEN10200
10540 PRINT:PRINT"Present line:":PRINT"<;L;";>";TEXT$(L)
10550 PRINT"Enter new line(c/r to leave)-":EDIT$="ON":
EDIT2$=TEXT$(L):GOSUB2000
10560 EDIT$="OFF":GOTO10200
11000 REM***DELETE LINES***
11010 CLS:GOSUB10005
11020 PRINT:PRINT"Enter line to delete from-":
PRINT"<c/r for Edit menu>":INPUTA$:IFLEN(A$)<1THEN10200
11030 A=VAL(A$):IFA<1ORA>NLTHEN11020
11040 PRINT:PRINT"Enter line to delete to(inc.)-":
PRINT"<c/r for rest of text>":
INPUTB$:IFLEN(B$)<1THENB$=STR$(NL)
11050 B=VAL(B$):IFB<1ORB>NLORB<ATHEN11040
11060 PRINT:PRINT"Delete lines ";A;" to ";B;".":PRINT"O.K.(Y/N)?"
11070 C$=INCH$:IFC$="N"ORC$="n"THEN10200
11080 IFC$<>"Y"ANDC$<>"y"THEN11070
11090 FORX=ATOB:TEXT$(X)="":NEXTX:IFB=NLTHEN11150
11100 K=0:FORX=ATO(NL-(B-A)):K=K+1:IFB+K>255THEN11050
11110 TEXT$(X)=TEXT$(B+K):NEXTX
11120 FORX=NL-(B-A)TONL:TEXT$(X)="":NEXTX
11150 NL=NL-((B+1)-A):PRINT:PRINT"Lines deleted.":
FORT=1TO500:NEXTT:GOTO10200
12000 REM***WRITE NEXT LINE***
12005 CLS:GOSUB10005
12006 PRINT:PRINT
12010 L=NL+1:IFNL=0THEN12040

```

```

2020 X=NL-2:IFNL<3THENX=1
2030 FORQ=XTONL:PRINT"<";Q;">";TEXT$(Q):NEXTQ
2040 PRINT:PRINT"Enter line(c/r to finish)-":GOSUB2000
2050 IFLEN(TEXT$(L))<1THEN10200
2060 NL=NL+1:GOTO12005
2500 REM***INSERT LINE***
2510 IF NL=255THENPRINT:PRINT"No free lines-255 used.":
FORT=1TO500:NEXTT:GOTO10200
2520 CLS:GOSUB10005
2530 PRINT:PRINT"Enter line no. to insert at-":
PRINT"<c/r for Edit menu>":INPUTA$:IFLEN(A$)<1THEN10200
2540 A=VAL(A$):IFA<10RA>NLTHEN12530
2550 FORX=NL+1TOA+1STEP-1:TEXT$(X)=TEXT$(X-1):NEXTX
2560 L=A:TEXT$(L)="" :PRINT:PRINT"Enter line to insert-":GOSUB2000
2570 NL=NL+1:PRINT:PRINT:PRINT"Line inserted.":GOTO40000
3000 REM***MOVE LINES***
3010 CLS:GOSUB10005
3020 PRINT:PRINT"Enter no. of first line to move-":
PRINT"<c/r for Edit menu>":INPUTA$:IFLEN(A$)<1THEN10200
3030 A=VAL(A$):IFA<10RA>NLTHEN13020
3050 PRINT:PRINT"Enter last line no. to be moved.-":
PRINT"<c/r for remainder>":INPUTB$:IFLEN(B$)<1THENB$=STR$(NL)
3060 B=VAL(B$):IFB<10RB>NLORB<ATHEN13050
3070 CLS:GOSUB10005
3080 PRINT:PRINT"Enter no. of line to move first line to-":
PRINT"<c/r for Edit menu>":INPUTC$:IFLEN(C$)<1THEN10200
3090 C=VAL(C$):IFC<1THEN13080
3095 IFC+B-A+1>255THENPRINT:PRINT"Not enough room!":PRINTB-A+1;
" lines(";A;"-";B;" ) from ";C;" needs ";C+B-A+1;" lines."
3096 IFC+B-A+1>255 THEN PRINT:FORT=1TO1000:NEXTT:GOTO13020
3100 D=B-A+1:FORT=1TOD:TEXT2$(T)=TEXT$(T+A-1):
TEXT$((T+A)-1)="" :NEXTT
3110 IFA<CTHEN13150
3120 X=0:FORT=(A-1)TOCSTEP-1:X=X+1:TEXT$((B-X)+1)=TEXT$(A-X):NEXTT
3130 X=0:FORT=CTOC+D-1:X=X+1:TEXT$(T)=TEXT2$(X):NEXTT:GOTO13200
3150 FORT=ATOC-1:TEXT$(T)=TEXT$(T+D):NEXTT
3160 X=0:FORT=CTOC+D-1:X=X+1:TEXT$(T)=TEXT2$(X):NEXTT
3200 PRINT:PRINT"Text moved.":GOSUB40000
3210 GOTO10200
4000 REM***DISPLAY LINES***
4010 CLS:GOSUB10005
4020 PRINT:PRINT"Enter line no. to display from-":
PRINT"<c/r for Edit Menu>":INPUTA$:IFLEN(A$)<1THEN10200
4030 A=VAL(A$):IFA<10RA>NLTHEN14010
4040 PRINT:PRINT"Enter line no. to display to-":
PRINT"<c/r for rest of text>":
INPUTB$:IFLEN(B$)<1THENB$=NL:GOTO14060
4050 B=VAL(B$):IFB<10RB>NLORB<ATHEN14040
4060 CLS:PRINT:PRINT"At pause press 'Q' to quit":PRINT
4080 FORQ=ATOB:PRINT"<";Q;">";TEXT$(Q):IF(Q/8)=INT(Q/8)THEN14100
4090 NEXTQ:PRINT:PRINT"Press any key to return to Edit Menu":
C$=INCH$:GOTO10200
4100 PRINT:PRINT"Press any key to continue":C$=INCH$:
IFC$="Q"THEN10200
4110 GOTO14090

```

# ACC AMTAT AND TATUNG SILICON DISC

Having already installed a silicon disc, I found that when I acquired an Amtat emulator and ran it according to the instructions, the disc refused to stop booting. ACC were unable to help and opening up my Einstein to remove the second ROM every time I ran Amtat was not an attractive option. After checking the Amtat BIOS, it turned out that making it compatible with the silicon disc is very simple - a switch of the allocation vectors is all that is needed.

The changes are as follows:

Address in memory	Original code	Modified code
EE44	E4 FD	B2 FD
EE56	B2 FD	F8 EE
EE76	F8 EE	C2 FD

Is it too much to hope that ACC will now buy themselves a silicon disc? They might even find it useful.  
C.P. Wallis Oct. 1986

### DISCS

Anyone who has suffered with a cassette recorder as a storage medium will understand the benefits of a disc drive. Being able to load and save programs or data quickly with reliability is of prime importance. Discs are known as random access devices, although data is read serially ie one bit after another, the position of the start of the data can be found at random very quickly.

EINSTEIN has a capability of using four drives, one or two internal 3" and external drives. Any external drives require their own power supply, the data cabling being connected to M004 on the rear of the machine. Any Shugartt interface drive, 3", 3.5", or 5.5" will work. (Shugartt interface is a standard that defines the signals used by the drive and the FDC). The drives are run in a daisy chain configuration, which means data is sent to all drives on a common line, the daisy chain, and a select signal gates the data to the selected drive. The same principal applies to a read from disc. This of course means that only one drive can be in use at any moment in time. The installed 3" drive is single sided, 40 track with the media being reversible and used on both sides. The use of double sided or 80 track drives requires an upgraded DOS such as System 5 from Xtal Research. (or the new DOS available from TATUNG shortly. Sub-Editor)

The recording media used is a magnetic oxide coating on a circular flexible disc, hence floppy disc. When in use the disc is in contact with the read/write head due to slight pressure from a head pad that rests on the opposite side of the disc, (With double sided discs two heads are used). Data is written to the disc in a binary pattern when a current is passed through the head magnetising the disc media. A disc read uses the magnetic changes on the media to create small current pulses in the disc head which are converted back to binary data.

The actual media is enclosed in a plastic case which gives good protection to the disc itself, and offers a rigid mounting to the drive. On the edge of the case that is inserted into the drive is a small notch, this is used by the drive to indicate whether side A or B is in use, via the red or green LED, (Light Emitting Diode), on the front of the drive.

Under the control of the FDC the disc head moves in or out along the disc surface as the disc rotates. The position of the head is controlled by a stepper motor. This is a precision motor that moves in steps, each step being a track on the disc surface. Each track on the disc is divided into 10 portions known as sectors. If you examine a 3" disc, near to the centre of the protective case is a small hole. The disc media also has a small hole, this is the index hole, a sensor within the drive uses this hole to output a pulse for each revolution of the disc creating the index pulses. These pulses are used by the FDC to check the drive is ready and as a reference point for positional information. This type of sectoring is called soft sectoring as it is the software that is formatting the tracks into 10 sectors. (Hard sectoring discs have as many index holes as sectors)

Before we actually go on to use our discs it is good practice to use the following rules :

- 1..DO NOT SWITCH ON OR OFF WITH A DISC IN THE DRIVE.
- 2..DO NOT EJECT, OR INSERT A DISC WHEN THE BUSY LIGHT IS ON.
- 3..DO NOT OPEN THE PROTECTIVE CASING.



...DO NOT STORE DISCS NEAR TO STRONG MAGNETIC FIELDS, SUCH AS HI-FI SPEAKERS.

...DO MAKE A COPY OF ANY SYSTEM DISC, TAKE REGULAR SECURITY COPIES OF IMPORTANT DATA DISCS.

All the above recommendations will help to keep your discs in good working order and will avoid corruption as far as is possible. Provided security copies are available not too much work will have to be re-done to recover if a corruption happens. Remember Murphey's Law; If it is possible, it will happen". So be warned!

As we are using soft sectoring a new disc requires formatting before use, a simple procedure using the BACKUP utility. Formatting the disc writes the sector pattern, writes the character E5 across the storage area of the disc, and copies the system DOS to tracks 0&1. The capacity of the 3" disc before formatting is 250K bytes. (Remember K=1024 bytes). After formatting the usable storage is 188K bytes, so where does the missing 62K go? As stated before there are 40 tracks each divided into 10 sectors. Each sector holds 512 bytes giving  $40 \times 10 \times 512 = 200k$  bytes storage, 50k bytes being used for the soft sectoring. The first two tracks on the disc are occupied by the DOS taking 10k bytes, so now we are down to 190k. 2k of the third track is used for directory entries, leaving 188K for storage of data. Each directory entry uses 32 bytes giving a maximum of 64 DIR entries,  $64 \times 32 = 2k$ ). The 32 bytes of each DIR entry form what is known as a File Control Block (FCB). We can examine a FCB by inserting our backup copy of the master disc and doing the following. (Never use the original system disc as the following examples could corrupt the disc if used)

```
...Load the disc
...MOS<E>
...R 8000 8800 0002<E>
...T 8000 809F<E>
```

The command R will read into memory from disc, the data is read into location 8000H to 8800H from track 2 sector 0. ie the 2K directory is now in memory. Using the tabulate command T we can examine the first five FCB's for XBAS.COM, BACKUP.COM, COPY.COM, LOGO.COM, LOGO.COM

The FCB bytes are as follows: (Using XBAS as the example)

```
0.....Drive Number (00)
-8....File Name (XBAS)
-11...Extension (COM)
2.....Extent
3-14...
5.....No. of 128 byte records (7A)
6-32...Location of 2K byte sector blocks
(010002000300040005000600070008)
```

The disc FCB is copied into memory when the file is loaded. The memory copy is updated as the file is altered and this FCB is written back to disc on saving the file. The first byte of the FCB is the drive number the file was saved under, the next eleven bytes should be quite familiar being the file name and extension, the file name is filled with spaces if less than eight bytes long. Byte 12 is the extent and is used to distinguish between FCB for large files that use more than one FCB e.g. LOGO.COM. Byte 15 is the number of 128 byte records, for XBAS  $7A \times 128 = 15616$  bytes. The remaining bytes indicate the actual location of the file in 2k sector blocks. Each block occupies four sectors, XBAS is on blocks 01-08 giving a total block size of 32k. From this it can be seen that the minimum disc space allocated to a file is 2k bytes. The first block is track 02, sector 04 to track 02, sector 07. The first block of XBAS can be loaded into memory from DOS by:

```
1...R 8000 8800 0402<E>
2...T 8000 8800<E> (800H being 2k)
```

Holding down the break key will stop the display scrolling and escape will re-enter MOS.

After formatting a new disc the directory sectors contain the character E5. It is this character in byte 00 of the FCB that is used to check whether a file exists. To examine this, find the first DIR entry, XBAS, if the copy of the master disc is being used, and:

- 1..Y<E>
- 2..ERA XBAS.COM<E>
- 3..DIR<E>

The file XBAS.COM should not be seen on the display, now do the following:

- 1..MOS<E>
- 2..R 8000 8800 0002<E>
- 3..T 8000 809F<E>

The first byte of the first FCB is now E5, replacing the drive number 00 that was originally there. The file is still present on disc but because the FCB is not valid the system cannot access the file. If we now do:

- 1..M 8000<E>
- 2..00.<E>
- 3..W 8000 8800 0002<E>
- 4..Y<E>
- 5..DIR<E>

XBAS should now be showing in the directory. It is only possible to regain files in total if no disc writes have occurred on the sector blocks originally allocated to the file. Also if the FCB is overwritten the only way to recover files is by reading all the sector blocks and examining the contents for the file, presuming you can recognise what to look for!

It is most likely that at some time the DOS on a disc will become corrupt. A common cause is a mains supply spike. One solution is to reformat the disc but all data will be lost, a better method is to rewrite the DOS tracks and see if the disc will function normally. The reason the disc may not work even after rewriting the DOS tracks is that some of the sector marks are corrupt and without these the FDC cannot position the read/write head to the required sectors. The usual error message for a loss of DOS tracks is "DISC NO SECTOR". To rewrite the system tracks:

- 1..Load a good disc
- 2..MOS<E>
- 3..R 2000 A800<E>
- 4..Remove good disc, insert corrupt disc
- 5..W 8000 A800<E>

The above procedure reads the system tracks from a good disc into memory 8000H to A800H, and then copies from memory to the bad disc.

Files that are not altered, or rarely changed can be Locked so accidental overwriting will not happen. System files such as XBAS should be locked, this is done using the DOS command LOCK. A locked file will show in the directory prefixed with an asterisk. This is also useful on important data files giving the operator an error message if a write to a locked file is attempted. The way DOS recognises that a file is locked is contained within that file's FCB. We can examine this by doing the following:

- 1..Load system disc
- 2..MOS<E>
- 3..R 8000 8800 0002<E>
- 4..T 8000 809F<E>

Memory 8000H onwards is now displaying the first five FCB's as in the previous example. Bytes 9-11 of the FCB are the extension, COM for a command file such as XBAS. The Hex representation for COM is 434F4D, the actual Hex showing in bytes 9-11 of the FCB for XBAS is C34F4D. XBAS on the original disc is supplied as a Locked file, to make a locked file 80H is added to the first character of the extension. So for a COM file C(43H) becomes C(C3H). To prove this we can:

- 1..Load system disc copy
- 2..UNLOCK XBAS.COM<E>
- 3..MOS<E>
- 4..R 8000 8800 0002<E>
- 5..T 8000 809F<E>

The extension bytes for XBAS are now 434F4D, showing the file to no longer locked

The same principle applies to the making of a System File, a System File will not show in a directory list but can be read as normal. It cannot be written to as a System File is also a Locked file. To create a System file the letter S is added to the end of the CK command eg. LOCK XBAS.COM S. The DOS will then add 80H to the second byte of the extension, thus COM-434F4D in hex becomes C3CF4D. System files could be used in an environment where it is desirable that the operator should not be able to list all the files via the DIR command, offering some security to the system.

### SCREENPLUS PLUS

A question has been asked on how to link SCREENPLUS to PICPEN. Well, here goes.

You will have read in the September issue the locations of the Video RAM. This is the key to the solution.

```
5100 FOR J = 0 TO 6143
5110 POKE &8000+J, VPEEK(J)
5120 NEXT J
```

This, then, is reading the pattern of the screen and POKEing it into a free area in Ram to be referenced later. 0=0000H 6143=17FF

```
5130 FOR J = 8192 TO 14335
5140 POKE &7800+J, VPEEK(J)
5150 NEXT J
```

This bit is now storing the colour part of the screen to Ram. 192=2000H 14355=37FF

```
5160 SAVE "SCO.OBJ",&7FF,&B000
```

We now save the piece of Ram with all the data in. Unfortunately we can only save it as a .OBJ file because of the restrictions of the basic. This can be overcome by RENaming it .MEM from within DOS.

```
5170 RETURN
```

OK, so now we have the subroutine. To patch it into PICPEN we have to modify the original.

```
EDIT..... 5010 if cr=&47 then beep:GOSUB 5100:pop:return:rem..
CHANGE... 2300-2330
```

```
4100 .....To suitable messages
```

With this all done we are now in business. Everything will run much the same as before, until <E> is input to end. This will cause control to pass to the new subroutine and your masterpiece will be saved to disc, before handing back control to the original program to end. There is one fault with this routine, it is VERY slow which considering what it is doing is a small price. You may think that it is 'hung up', it hasn't, it just seems like it. Be patient (said the nurse to the old man).

### USEFUL TIP

Type lines 5100-5160 and SAVE them as "SCR-FILE". Then you can 'merge' them as follows:-

- 1) Know the line numbers you want them to become
- 2) SAVE your 'WORK' file if you have one.
- 3) LOAD "SCR-FILE" and list it.
- 4) LOAD your 'WORK' file DO NOT LIST IT
- 5) EDIT the line numbers on screen of SCR-FILE

### NOTES

When this routine is called it will save whatever is on the screen at the time ie READY, the cursor etc. Make sure you put it in the right place.

Another thing I have found is, no matter what colour the background it always goes over as black and empty (You can't use the black colour changer on it). To solve this FILL as much of the picture as possible and touch it up in SCREENPLUS.

Martin Page

oooOoOooo

# USING A SERIAL PRINTER

In issue 2 it was stated that FA3D should be modified to A0. This is not correct. Locations FA00 onwards hold the BIOS (Basic Input Output System) vectors of the CP/M operating system.

- FA00 Cold Start Vector
- FA03 Warm Start Vector
- FA06 Console Status Vector
- FA09 Console Input Vector
- FA0C Console Output Vector
- FA0F Centronics Port Output Vector
- FA12 Serial Port Output Vector

Programs that use the BIOS vectors (Kuma DATABASE is one) can be made to drive the centronics or serial port by altering the vectors or altering the MCAL (Machine Calls) that the vectors point to. The MCAL for serial output is A0, therefore if location FA3F is modified to A0 from 9F (centronics MCAL) output is directed to the serial port.

1. Load DOS
2. Enter MOS
3. M FA3F
4. Type A0 <enter>
5. Type . <enter>

Another interesting modification is as follows:-

1. M FA0C
2. CD <enter> . <enter>
3. M FA10
4. 41 <enter> . <enter>

The printer will now follow the console in DOS only (I haven't worked it out in BASIC yet!)

Graham Betteridge

## ALLSORTS

When writing programs it is not long before you need to sort a set of data into some sort of order, be it alphabetical or numeric. This could be your Telephone Directory or your list of LP's. When the number of items is small it presents no problem but as the list grows so does the length of time taken to sort it. The bubble sort is good for small numbers of items as the code is simple to write and does not take up a great deal of space. Unfortunately the time taken to sort the list obeys a square law, i.e. double the data and the time increases fourfold! (Kuma Database is a good example of sorting large amounts of data, even the manual suggests brewing a cup of tea whilst the sort is done)

In this series of articles we will be investigating various types of SORTS and how they work along with the length of time taken to arrange a given set of data. This months sort is the bubble sort. It gets its name from the way the lighter elements 'bubble' to the top of the list as the sort progresses.

```

10 DIM ENTRY$(100)
20 FOR ENT=1 TO 100
30 READ ENTRY$(ENT)
40 NEXT
50 CLS
60 FOR ENT= 1 TO 100
70 PRINTENTRY$(ENT);" ";
80 NEXT
90 TIS="000000"
100 PRINT:PRINT"BUBBLE SORT
    ";TIS
110 BOTT=100
120 TP=1
130 PNTR=BOTT
140 IF ENTRY$(PNTR)<ENTRY$(
    (PNTR-1)THEN SWAP
    ENTRY$(PNTR),ENTRY$(PNTR-1)
150 PNTR=PNTR-1
160 IF PNTR>TP THEN 140
170 TP=TP+1
180 IF BOTT>TP THEN 130
190 T$=TIS
200 PRINT"SORTED 100 ITEMS
    IN ";TIS
210 FOR ENT=1 TO 100
220 PRINTENTRY$(ENT);" ";
230 NEXT

```

```

1000 DATA"ZXCV","ASDF","QWER","MNBV","LKJH","POIU","WQAS","SAXD"
1010 DATA"DGYB","CYTY","VOIT","FIJO","RTJE","TAJM","GEIE","BOUK"
1020 DATA"YSEF","UOED","ILAR","KORE","OJRF","LSEG","KDRG","JRTW"
1030 DATA"FXUI","DOUK","SSUM","APHN","QDOJ","WFGI","ESDF","RSFJ"
1040 DATA"BSR","VORD","AGGH","CERI","XAIJ","ZEIG","VEFU","WRJF"

```

```

1050 DATA"QFRD","WDKF","EIUD","RDIH","AJGN","CGJH","VJDF","UJGJ"
1060 DATA"PIRJ","IEHH","EHGI","HKDJ","KSUS","SJRG","JJFF","JSEC"
1070 DATA"NAGF","ADRG","DWRQ","WQRF","QPTC","PJHV","JDNU","DCIE"
1090 DATA"VFHR","FKSI","KIAR","IARE","XSFF","CFYJ","VKER","EKJG"
1100 DATA"RGKJ","CJGK","DLRU","JDLR","RUDH","JSBJ","KSDT","SKVI"
1110 DATA"XDGF","EIUU","SKJR","JDCR","CJDF","GFJF","SDVV","DNHV"
1120 DATA"IJFS","MORF","VOUD","LSFD","TSDH","YSCM","HIRT","GEXK"
1130 DATA"XCSZ","EDXX","NOMR","HOUI"

```

brief description of the program,  
lines 10 to 90 read the test data and print it to the screen  
lines 10 to 190 actually do the sort (and these are the lines that will  
be changed each month)  
lines 20 to 230 print the sorted list  
lines 1000 to 1130 contain the list of data to be sorted to ensure  
that the same data is sorted each time.  
Now it works, BOTT points to the last item in the list, TP points to  
the first item in the list, and PNTR points to the item currently being  
compared.  
At line 130 PNTR is set to the last item in the list, which is then  
compared to the next one up. If it is smaller (or lighter) the two will  
be swapped over. PNTR is then reduced by one pointing to the next  
element up the list, the comparison repeated and a swap occurring again  
if needed. In this way the routine will pick up on the smallest (or  
lightest) element and Bubble it to the top of the list (hence the  
name).  
Try halving the number of items to be sorted and check the difference  
in time taken to sort.

#### MISSIONARIES and CANNIBALS

This is an adapted version of a program in Artificial Intelligence; an  
introductory course, which was originally written in Edinburgh Logo.  
The idea of the program is that it searches for a solution to the  
'Missionaries and Cannibals Problem' which is as follows in case you  
haven't come across it before.

Three missionaries and three cannibals seek to cross a river from  
the left bank to the right bank. A boat is available, which will  
hold two people and can be navigated by any combination of  
missionaries and cannibals involving one or two people. If the  
missionaries on either bank are outnumbered at any time by  
cannibals, the cannibals will indulge in their anthropophagic  
tendencies (Alan Bundy does like big words!!!) and do away with  
the missionaries. When the boat is moored at the bank it is  
counted as part of the bank for these purposes.

Find the simplest schedule of crossings that will permit all the  
missionaries and cannibals to cross the river safely.

The program is totally devoid of comments, I'm afraid, partly as I  
never thought anyone would want a copy and partly because it is so  
well documented at every stage of its design in the above mentioned  
book by Alan Bundy, Published by EDINBURGH UNIVERSITY PRESS, which  
retails at a very reasonable price for the quality information it  
contains. (About £7.00)

Although undocumented, the program is fairly easy to understand as it  
uses a lot of very short procedures whose names are self explanatory.  
To start the program load LOGO. Load "mandc, and type 'mandc' when  
prompted. If you want to change the order of the allowed moves that is  
very simple and will allow you to confirm that the suggested schedule  
is the same no matter the order the order in which the instructions  
are held (although it may take a little longer in some cases). It's  
also pretty easy to alter the number of missionaries and/or cannibals  
if you so desire.

to mandc



```

start mandc
tryall :possiblemoves
end
to startmandc
ct
pr [MISSIONARIES AND CANNIBALS PROGRAM]
pr [_____]
pr [.....automatic version.....]
pr []
make "leftbank [m m m c c c boat]
make "rightbank []
make "possiblemoves [[c c boat] [c boat] [m c boat] [m m boat] [m
boat]]
make "stateseen []
recordnewstate
pr [INITIAL STATE]
printstate
end
to recordnewstate
make "stateseen fput getstate :stateseen
end
to getstate
op append (numberof "boat :leftbank) (list (numberof "m :leftbank)
(numberof "c :leftbank))
end
to tryall :setofmoves
if empty :setofmoves [stop]
explorestate :leftbank :rightbank first :setofmoves
tryall bf :setofmoves
end
to explorestate :leftbank :rightbank :movelist
if not applicable :movelist[stop]
applymove :movelist
pr[]
pr se [APPLIED MOVE...] :movelist
if missionarieseatenp [pr [MISSIONARIES EATEN!!! move rejected] stop]
if stateseenbeforep [pr [STATE SEEN BEFORE...move rejected] stop]
pr "NEWSTATE
printstate
if succeededp [pr "SUCCESS!!! pause]
recordnewstate
tryall :possiblemoves
pr "BACKUP
end
to applicalep :move
op not (or (>(numberof "m :move) (numberof "m fromside))(>(numberof "c
:move) (numberof "c fromside)))
end
to fromside
if amongp "boat :leftbank [op :leftbank]
if amongp "boat :rightbank [op :rightbank]
pr [error in procedure 'fromside']pause
end
to applymove :movelist
if amongp "boat :leftbank [moveltor :movelist stop]
if amongp "boat :rightbank [movertol :movelist stop]
end
to stateseenbeforep
op amongp getstate :stateseen
end
to without :deletions :original
local "tempop
if empty :deletions [op :original]

```

```

make "tempop without1 first :deletions :original
op without bf :deletions :tempop
end
to without1 :deletion :original
if empty? :original [pr se [Cannot remove] fput :deletion [from given
list] throw "toplevel]
if = :deletion first :original [op bf :original]
op fput first :original without1 :deletion bf :original
end
to append :list1 :list2
local "templist
if empty? :list1 [op :list2]
make "templist lput first :list1 :list2
op append bk :list1 :templist
end
to amongp :item :list
if empty? :list [op 0 = 1]
if = :item first :list [op 0 = 0]
op amongp :item bf :list
end
to numberof :atom :list
op auxnumberof :atom :list 0
end
to auxnumberof :atom :list :tally
if empty? :list [op :tally]
if = first :list :atom [make "tally (+ :tally 1) op auxnumberof :atom
bf :list :tally ]
op auxnumberof :atom bf :list :tally
end
to Printstate
pr [ _____ ]
printleftbank
printrightbank
pr [ _____ ]
end
to moveltor :movelist
make "leftbank without :movelist :leftbank
make "rightbank append :movelist :rightbank
end
to movertol :movelist
make "rightbank without :movelist :rightbank
make "leftbank append :movelist :leftbank
end
to printleftbank
pr se [leftbank is ....] :leftbank
end
to printrightbank
pr se [rightbank is ....] :rightbank
end
to missionariesatemp
op or meatemp :leftbank meatemp :rightbank
end
to meatemp :bank
op (and (> (numberof "c :bank) (numberof "m :bank))(>(numberof "m
:bank) 0))
end
to succeededp
op empty? :leftbank
end
to applicablep :move
op not (or(>(numberof "m :move)(numberof "m fromside))(>(numberof "c
:move)(numberof "c fromside)))
end

```

## LETTERS LOGO/PROLOG

Thank you for the information on LOGO. I think I will now be able to make good use of the 'property list' facilities, the area where I had made least headway.

I am interested to find that you recommend contacting Dr. Mike Bayliss for information on PROLOG as I had already tried to contact him some time before I wrote to you. The result of my first letter to him was that he sent me a copy of his price list with the words 'Prolog available for Einstein' written along the top. Obviously I was most interested and I wrote back almost immediately to ask for further details, however, I have never heard from him again although I have written again, at least once, if not twice.

Thanks for the information about Public Domain Software. I should be glad to hear about any which you have available as you mention in your letter.

Finally, I am sending you a copy of my only program in Dr. Logo. The reason I haven't any more (useful) programs is that I've only been using Dr. Logo on the Einstein to try out small ideas prior to using them 'full scale' as it were in my project at university for my M.Sc. dissertation.

Thanks once again for all your help and keep up the good work for the UKEUG, Yours sincerely, Steven Salvini.

Dear Keith,

It is very encouraging to see that you have been able to persuade Tony Brewer to provide his Silicon Disc note, but I was disappointed to see the incoherent letter which appeared to disapprove of the August Newsletter. As far as I am concerned this was one of the very few newsletters which contained useful material.

I have been on the point of writing several times, but I have to say that at present I am not getting value in the newsletter for my £15.00. Surely all the readers can't be beginners? The contributions which I have sent in have stimulated no reaction whatsoever and one must have some sort of feedback to avoid discouragement. I had hoped to send in to the newsletter my SUBMIT when debugged, but I need some sort of sign that it will not be wasted on the newsletter readership. Can you not do a little stimulating to get some more advanced discussion? What happened to technical queries? These should generate further contributions. What about an Einstein BIX?

Incidentally, all HISOFTS CP/M software is NOT available on the Einstein. They told me CATALOG won't work, so I bought the AM\*\*AD version and ran it with AMTAT-it still won't work, just swallows string input and asks for more. Any ideas??

Yours Sincerely C P Wallis

Dear Keith, When am I going to get my October issue of the UKEUG? At the exhibition it was promised as "shortly". Perhaps it has got lost in the post? If so could you please let me have an other one. Thank you in anticipation.

Whilst I am writing to you, in the August review on the Silicon Disc what on earth is in paragraph (b) "the SP and SD issue" (c) "SUBMIT" (d) "CCP and BDOS (DDM and OSM) BIOS (HDM)" ????? There seems to be an assumption that one is born with the knowledge of computers.

The last time I went to school was 40 years ago and I have to start from the bottom of the ladder. Programs like Basic

Basic are the things to get started on, after all the language designed for the Albert is XBAS. Yours 'Idiot 3rd class' H.R.Kauter Ilford Essex

PS. I like to say that I am looking forward to every issue (even though I don't understand most of it (yet)), and the reason I am so peeved is that I am still waiting. waiting waiting waiting ETC.

PPS. You're doing a good job (see I am calming down already).  
REPLY.

Yes it was promised 'shortly', in fact at the time of the exhibition half of them had been printed and the rest were not ready until the end of the following week. Remember that this is all done in our 'SPARE' time so please do not get too mad with us if things take a little longer than expected.

Good Question .

(b) The SP and SD were programs supplied with the pre-production version of the Silicon Disc. SP was the Silicon Formatting program and SD was the patch file which interfaced between the computer and the Silicon Disc. The ISSUE was that if you done a RESET or CTRL/BREAK then you had to run SD again or the Silicon Disc did not exist. This has been resolved by putting them onto the second ROM which takes care of this housekeeping automatically.

(c) Xtal Dos and the Einstein have been sold as CP/M 'COMPATIBLE' they are to some extent and probably upto 80 percent if you add a second drive and the 80 column card . But some things just are not there, such as the SUBMIT facility. In CP/M this allows you to create a file of commands which can then be acted upon as if you were typing them in from the keyboard just by typing SUBMIT .

(d) DOS has three sections these are.

CP/M XTAL (p57 Dos/Mos Manual)  
Console Command Processor (CCP) Dos Monitor Module (DMH)  
Basic Input/Output System (BIOS) H/ware Dependent " " (HDM)  
Basic Disk Operating System (BDOS) Operating System " " (OSM)  
The only assumptions that we make are that 'OUR' members (meaning they are your members also) all want to learn more about the Einstein. But, and it's a big but, some know more than others and therefore it is very difficult to balance the Newsletter for everybody, in fact we receive just as many letters wanting more advanced info as those who are after the basic 'lets get started information'.

Yes XBAS was designed for Albert, but Albert was designed to learn any Language.

You may not understand most of it yet but if you keep asking questions it will soon become clearer. Another tip is to re-read back issues as things which did not make sense then could well do now. Keith.

WELCOME TO THE FOLD

Dear UKEUG, thank you for your letter of 21.1.86 You might like to know I ordered the Einstein in early July and received a machine a few weeks later with a defective keyboard. After 2 expensive phone calls and several letters (ending with one to the M.D.) a replacement was received on 22nd October. An altogether unpleasant experience, L.J.Rodericks

REPLY

We at UKEUG are sorry to hear of your initial bad experience but hope that now you have the best 8 bit micro on the market you have many happy hours with it. If you have any further difficulties please do not hesitate to contact us.