THE AMSTRAD USER

Issue no. 1 $3.00 February 1985

$2 500 WORTH OF PRIZES TO BE WON!

- *USER GROUP INFORMATION & NEWS*
- *USERS' HOT-LINE*
- *GAMES*

FOR THE NOVICE & EXPERIENCED USER

NEW

STRATEGY SOFTWARE

A Winning Hand!

The Trials of Arnold Blackwood

A test of the ingenuity of Arnold and thus the Adventurer. Strangely, but probably inevitably, the actual mission is unstated! The tasks are thus to be determined by adjudging the 'clues', the logic, and the nature of the introduction. There is a nicely atmospheric use of sound and The Trials would be a very worthwhile addition the the Adventurer's library.

Diamond Miner

To refuel your saucer you must descend into the pit and recover the many diamonds there. Deep below there are many dangers including evil spiders, moving bridges and falling rocks – you only have a limited amount of time – so hurry! Joystick control optional.

Character Editor

Until now, when creating game programs for the Amstrad, you've had to re-define graphic characters each time you loaded an unfinished program prior to beginning work on your game. Character Editor will allow you to re-define your characters and save your definitions to tape as a subroutine, to be loaded back whenever you wish to work on your program. Anyone writing programs incorporating user-defined graphics will find this aid indispensable.

Amstrad Unlocked

What the handbook doesn't tell you! 24k of hints, tips, PEEKS and POKES; hex–what, why and how; extra commands, scroll, bell, MID$=, Timeset, FIND; BASIC bugs and how to overcome them; sort routines; how programs are stored; tokens; control keys and codes; line zero; ROM listout, etc.

Royal Quest

In this text only adventure you will be able to experience some of the thrills and dangers in exploring a strange and magical land. You will instruct a guide to carry out whatever commands you feel are necessary to complete your unknown task.

Enquiries to:
Strategy Software, Suite 4a, 33-45 The Centreway, Blackburn Road,
Mt. Waverley, Victoria 3149 Tel: (03) 233 9227

THE AMSTRAD USER

Issue No. 1 February 1985

Contents

All enquiries and contacts should be made to The Amstrad User, Suite 4a, 33-45 The Centreway, Blackburn Road, Mt. Waverley 3149, Australia. Tel: (03) 233 9227.

The Amstrad User is published each month by Strategy Publications. Reprinting of articles in The Amstrad User is strictly forbidden without written permission. All rights reserved. Copyright 1985 by Strategy Publications. The single copy price of $3.00 is the recommended retail price only. Also available on subscription for 12 months.

Please note that whilst every effort is made to ensure the accuracy of all features and listings herein, we cannot accept any liability whatsoever for any mistakes or misprints.

Contributions are welcomed from readers or other interested parties. If you want them returned, then please send a large SAE with all submissions.

Editoral

G'day,

After Kev and I had finally thrown in the towel and were bathing our sore feet a thought came to us simultaneously. Why trudge around newsagents buying mountains of magazines just to get scraps of information about the new Amstrad CPC464? Why not publish a magazine devoted specifically to the CPC464? Why not use the publication to promote User Groups and exchange information and ideas?

And so, The Amstrad User was born.

It would have been a difficult birth had it not been for encouragement and help from various sources. AWA in their efforts to publicise the magazine on our behalf, Sharon for providing layout and design assistance, Peter for minimising the paper he was wasting whilst emulating the Old Masters, and Philip the Keystroke King. Not to mention Kev, whose feet are much better now, and Colin for dotting the i's and crossing the t's.

Much effort has gone into the production of this first edition, and those sensible enough to have invested in a subscription (a mere $30) can be assured of a constant flow of very relevant information over the next twelve months and beyond. But the ultimate success of The Amstrad User will depend upon contributions from the 'field', in other words the Users. Naturally any published contributions will be rewarded and can take the form of software, reviews, hints, letters of praise (or moans) and User Group information. So put pen to paper (or fingers to keyboard) and give us heaps.

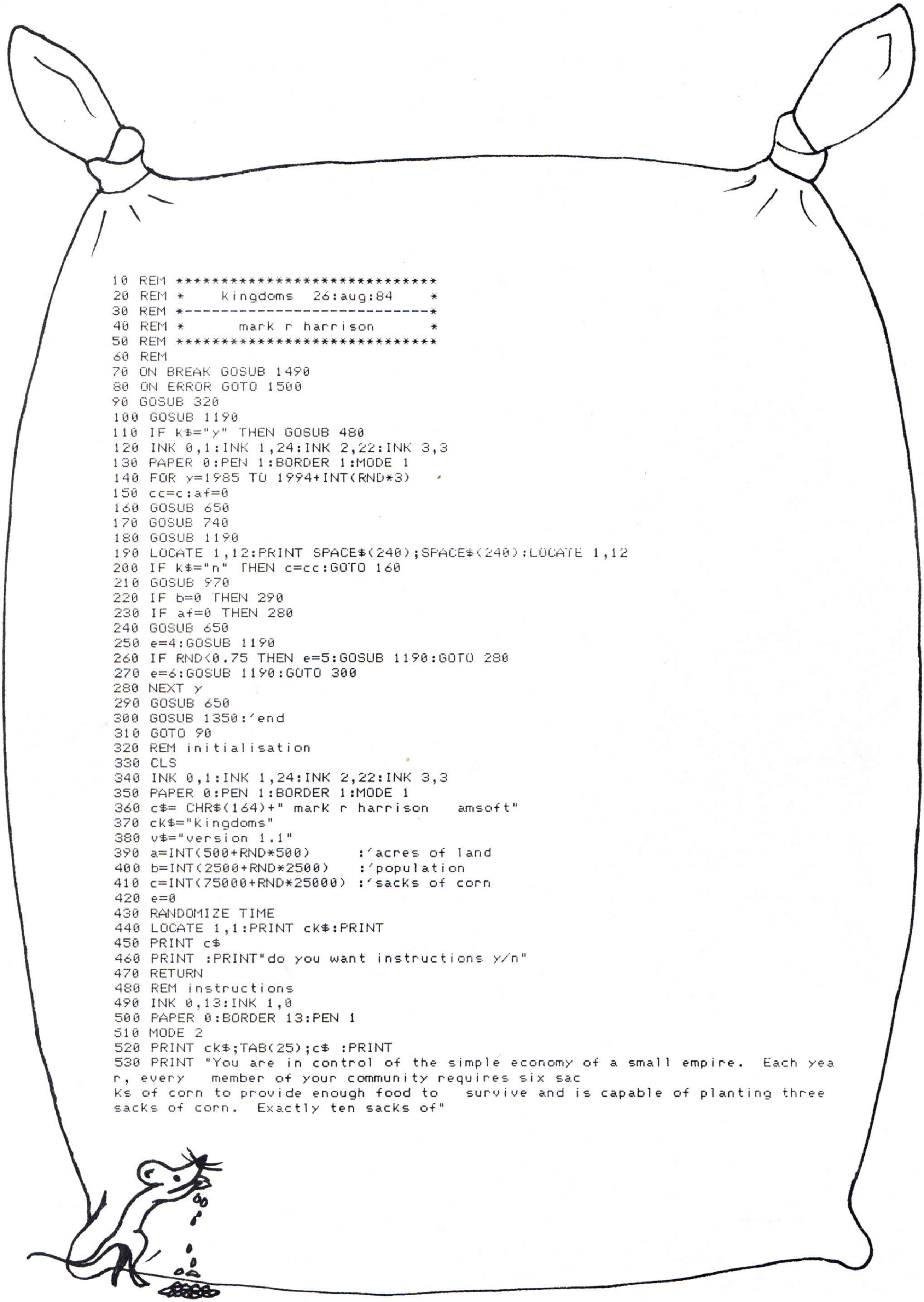
Talking about heaps, take a look at the prizes being offered in our first Competition. If they don't stir you into action nothing will!

See you next month,

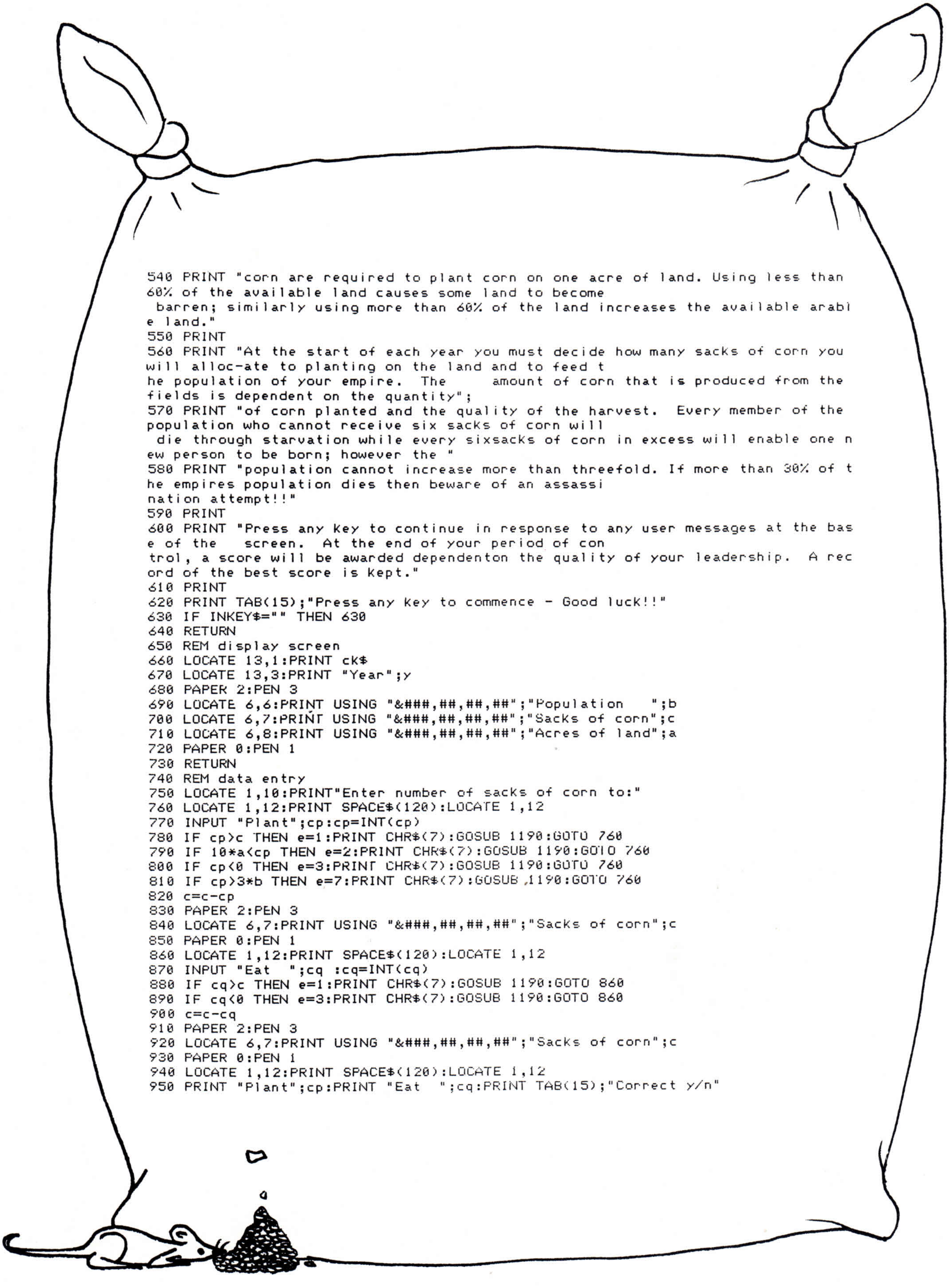
Ed

Kingdoms

This is a brand of program that slots somewhere between an adventure and a strategy game, and has the distinct advantage that it is the type of program example much beloved by Basic Programmers since it can be nurtured to grow and expand in very many directions. You can add elements to the 'circumstances', illustrate with graphics and so on. See if you can simulate a field of growing corn. The extensive editing facilities of Locomotive Basic will help a lot.



```
10 REM ******************************
20 REM *     kingdoms  26:aug:84     *
30 REM *----------------------------*
40 REM *      mark r harrison        *
50 REM ******************************
60 REM
70 ON BREAK GOSUB 1490
80 ON ERROR GOTO 1500
90 GOSUB 320
100 GOSUB 1190
110 IF k$="y" THEN GOSUB 480
120 INK 0,1:INK 1,24:INK 2,22:INK 3,3
130 PAPER 0:PEN 1:BORDER 1:MODE 1
140 FOR y=1985 TO 1994+INT(RND*3)
150 cc=c:af=0
160 GOSUB 650
170 GOSUB 740
180 GOSUB 1190
190 LOCATE 1,12:PRINT SPACE$(240);SPACE$(240):LOCATE 1,12
200 IF k$="n" THEN c=cc:GOTO 160
210 GOSUB 970
220 IF b=0 THEN 290
230 IF af=0 THEN 280
240 GOSUB 650
250 e=4:GOSUB 1190
260 IF RND<0.75 THEN e=5:GOSUB 1190:GOTO 280
270 e=6:GOSUB 1190:GOTO 300
280 NEXT y
290 GOSUB 650
300 GOSUB 1350:'end
310 GOTO 90
320 REM initialisation
330 CLS
340 INK 0,1:INK 1,24:INK 2,22:INK 3,3
350 PAPER 0:PEN 1:BORDER 1:MODE 1
360 c$= CHR$(164)+" mark r harrison   amsoft"
370 ck$="kingdoms"
380 v$="version 1.1"
390 a=INT(500+RND*500)      :'acres of land
400 b=INT(2500+RND*2500)    :'population
410 c=INT(75000+RND*25000)  :'sacks of corn
420 e=0
430 RANDOMIZE TIME
440 LOCATE 1,1:PRINT ck$:PRINT
450 PRINT c$
460 PRINT :PRINT"do you want instructions y/n"
470 RETURN
480 REM instructions
490 INK 0,13:INK 1,0
500 PAPER 0:BORDER 13:PEN 1
510 MODE 2
520 PRINT ck$;TAB(25);c$ :PRINT
530 PRINT "You are in control of the simple economy of a small empire.  Each yea
r, every   member of your community requires six sac
ks of corn to provide enough food to   survive and is capable of planting three
sacks of corn.  Exactly ten sacks of"
```



```
540 PRINT "corn are required to plant corn on one acre of land. Using less than
60% of the available land causes some land to become
 barren; similarly using more than 60% of the land increases the available arabl
e land."
550 PRINT
560 PRINT "At the start of each year you must decide how many sacks of corn you
will alloc-ate to planting on the land and to feed t
he population of your empire.  The      amount of corn that is produced from the
fields is dependent on the quantity";
570 PRINT "of corn planted and the quality of the harvest.  Every member of the
population who cannot receive six sacks of corn will
 die through starvation while every sixsacks of corn in excess will enable one n
ew person to be born; however the "
580 PRINT "population cannot increase more than threefold. If more than 30% of t
he empires population dies then beware of an assassi
nation attempt!!"
590 PRINT
600 PRINT "Press any key to continue in response to any user messages at the bas
e of the   screen.  At the end of your period of con
trol, a score will be awarded dependenton the quality of your leadership.  A rec
ord of the best score is kept."
610 PRINT
620 PRINT TAB(15);"Press any key to commence - Good luck!!"
630 IF INKEY$="" THEN 630
640 RETURN
650 REM display screen
660 LOCATE 13,1:PRINT ck$
670 LOCATE 13,3:PRINT "Year";y
680 PAPER 2:PEN 3
690 LOCATE 6,6:PRINT USING "&###,##,##,##";"Population   ";b
700 LOCATE 6,7:PRINT USING "&###,##,##,##";"Sacks of corn";c
710 LOCATE 6,8:PRINT USING "&###,##,##,##";"Acres of land";a
720 PAPER 0:PEN 1
730 RETURN
740 REM data entry
750 LOCATE 1,10:PRINT"Enter number of sacks of corn to:"
760 LOCATE 1,12:PRINT SPACE$(120):LOCATE 1,12
770 INPUT "Plant";cp:cp=INT(cp)
780 IF cp>c THEN e=1:PRINT CHR$(7):GOSUB 1190:GOTO 760
790 IF 10*a<cp THEN e=2:PRINT CHR$(7):GOSUB 1190:GOTO 760
800 IF cp<0 THEN e=3:PRINT CHR$(7):GOSUB 1190:GOTO 760
810 IF cp>3*b THEN e=7:PRINT CHR$(7):GOSUB 1190:GOTO 760
820 c=c-cp
830 PAPER 2:PEN 3
840 LOCATE 6,7:PRINT USING "&###,##,##,##";"Sacks of corn";c
850 PAPER 0:PEN 1
860 LOCATE 1,12:PRINT SPACE$(120):LOCATE 1,12
870 INPUT "Eat  ";cq :cq=INT(cq)
880 IF cq>c THEN e=1:PRINT CHR$(7):GOSUB 1190:GOTO 860
890 IF cq<0 THEN e=3:PRINT CHR$(7):GOSUB 1190:GOTO 860
900 c=c-cq
910 PAPER 2:PEN 3
920 LOCATE 6,7:PRINT USING "&###,##,##,##";"Sacks of corn";c
930 PAPER 0:PEN 1
940 LOCATE 1,12:PRINT SPACE$(120):LOCATE 1,12
950 PRINT "Plant";cp:PRINT "Eat  ";cq:PRINT TAB(15);"Correct y/n"
```

```
960 RETURN
970 REM calculate values
980 aa=cp :bb=cq
990 IF bb>2*b THEN bb=3*b.
1000 ap=aa*100 :a=INT(a+(ap-60))
1010 IF bb <0.7 THEN af=1
1020 b=INT(bb)
1030 hv=RND
1040 c=c+INT(4*cp*(hv+0.5))
1050 LOCATE 1,12:PRINT SPACE$(120):LOCATE 1,12
1060 PAPER 2:PEN 0
1070 e=INT(hv*5+1)
1080 IF e=1 THEN e$="Very poor harvest"
1090 IF e=2 THEN e$="Poor harvest"
1100 IF e=3 THEN e$="Average harvest"
1110 IF e=4 THEN e$="Good harvest"
1120 IF e=5 THEN e$="Excellent harvest"
1130 e=0
1140 LOCATE 10,15:PRINT e$
1150 FOR tt=0 TO 2500:NEXT tt
1160 PAPER 0:PEN 1
1170 LOCATE 1,15:PRINT SPACE$(40)
1180 RETURN
1190 REM user messages
1200 IF e=0 THEN 1310
1210 IF e=1 THEN e$="Insufficient supply of corn"
1220 IF e=2 THEN e$="Insufficient fertile land"
1230 IF e=3 THEN e$="Positive values only"
1240 IF e=7 THEN e$="Insufficient labour force"
1250 IF e=4 THEN e$="BEWARE !!  assassination attempt"
1260 IF e=5 THEN e$="You were lucky - take more care"
1270 IF e=6 THEN e$="You have been shot dead !!"
1280 PAPER 3
1290 LOCATE 1,24:PRINT e$
1300 e=0
1310 k$=INKEY$:IF k$="" THEN 1310 ELSE k$=LOWER$(k$)
1320 PAPER 0
1330 LOCATE 1,24:PRINT SPACE$(40)
1340 RETURN
1350 LOCATE 1,10:PRINT SPACE$(200)
1360 k$=""
1370 LOCATE 1,10
1380 s=INT(b*c*a/1E+09)
1390 IF af=1 THEN s=0
1400 PRINT "Your reign has come to an end":PRINT
1410 PRINT "You have been awarded";s;"points":PRINT
1420 IF s>h THEN h=s
1430 PRINT "The best score obtained today is";h:PRINT
1440 PRINT "Do you wish to play again y/n ?":PRINT
1450 k$=INKEY$:IF k$="" THEN 1450 ELSE k$=LOWER$(k$)
1460 IF k$="n" THEN CLS:NEW
1470 IF k$<>"y" THEN 1450
1480 RETURN
1490 STOP   :'break
1500 GOTO 70:'error trap
```

How to Review

If you ever get the opportunity to review any software, or are contemplating sending your own creation for an expert evaluation, then here are a few pointers.

Most reviews are carried out by using a 'weighting method'. That means a points value is awarded to each element of the software (originality, use of graphics etc.), and all the element scores are totalled to give an overall assessment which can be compared with other software of a similar nature. The points awarded can get complicated as standards change and new benchmarks are applied, but to keep things simple the following ratings cover most situations and opinions:-

Excellent	– 8 points
Good	– 6 points
OK	– 4 points
Well, maybe	– 2 points
Yuk!	– No points

The two points difference between each rating allows a certain amount of flexibility if, for example, originality is considered to be better than Good, but does not come into the Excellent category, a score of 7 can be awarded. If you really want to be fussy, use of decimals can be made.

Now let's take a look at the elements of the software to be evaluated.

Ease of Use – naturally, the age of the end-user has a lot to do with any evaluation. So bearing that in mind, a piece of software which uses a joystick and a fire button or a minimum number of keys to acheive a result should score well. A cluttered screen will cause confusion and cost points. The choice of colours is important too as clarity has much to do with ease of use.

Speed – not much can be done about the loading speed from cassette (apart from loading from disc), but much can be achieved during the running of the program. Good response times to operator action helps to make software less frustrating. This is particularly important for games and in many cases turns an otherwise good game into a mediocre one.

Entertainment value – again, associated with games and to some extent educational software, the entertainment value is crucial to the presentation of the software as a whole. Despite the fact that a piece of software can score well in most or all of the other elements, it will lose out if it is just too easy or much too hard. Games which allow different levels to be selected generally get over this problem and prove to have a wider range in entertainment value and therefore, score higher.

Documentation – how many times have you purchsed a piece of software with documentation restricted to 'Type RUN when loaded', and you are left to figure out the rest? Documentation consists of three parts. First, the general description which is meant to outline the purpose of the software. Second, the loading instructions including any peculiarities which may be encountered and/or the use of keys, function key, joysticks etc. Finally, the instructions which appear on the screen should be optionally selected. Software for business purposes will normally have (and should have) a much more sophisticated level of documentation, and educational software may contain additional hints, tips or exercises.

Originality –all the best pieces of software have their mimic versions, but more often than not, it is the original that has the best success. With such a wealth of software available, it is becoming increasingly difficult to provide or find original software. So when something different appears it scores top marks. But with the number of programmers increasing proportionately with the increase in microcomputer sales, the resources for original works is far from diminishing.

Use of Graphics –with coloured screens, the use of graphics can be quite amazing or plainly irritating. Too many colour changes during a run, for example flashing borders, tend to tire the eyes quickly. It should be noted that, in general, pastel shades are the most pleasing to the eyes. High marks will be achieved for an uncluttered screen and careful design of animated objects (e.g. space ships, planes, boats, etc.) Graphics can be used to very good effect in educational software, but sadly lacks in many products.

Ability to hold interest – after all is said and done, a piece of software which has superb graphics, is easy to use, has good documentation and is original in concept will undoubtedly fail if it is unable to maintain interest from the user. It may be too difficult, to easy, or just plain boring. A number of successful products have gained good points in this element because of an injection of humour. Introducing time limits, with the reducing time displayed, tends to keep the adrenaline running, as will unexpected events (except program bugs of course!). Sound effects also play an important part. Users will turn off quickly if they have to endure a lengthy tune before starting a game, or are subjected to a monotonous background beat during the run. In essence, sound should be used wisely.

Summary – the above elements cover the areas of most interest in formulating an opinion about a piece of software. However, it is the end user who will ultimately determine the success or failure. So, if you do find yourself in a position to submit software for evaluation, try it out on a number friends first. They may spot something you have missed, or suggest improvements. They may even turn up some bugs for you! You will also find the article helpful as a check list before submitting your entry to The Amstrad User Competition. It could make all the difference.

User Group Information

Each month, as much space as demanded will be allocated to this section, as we believe it to be one of the major reasons for launching The Amstrad User, that is to assist in the formation of User Groups throughout Australia and establish an 'information network'.

This will be acheived by providing a central point of co-ordination and regularly publishing an updated list of newly formed and existing Groups. The list will be divided into States, and for each Group we will print the name, address and telephone number of the local branch convenors. In addition, meeting times and places or any other information or messages will be printed.

At a national level we are well placed to direct people making the many enquiries we receive concerning membership of User Groups to the nearest local branch convenor. We are also in a good position to monitor the progress of Groups, technical problems, opinions and bring to subscribers' attention any major points of interest, including developments by Amstrad themselves.

However, nothing will be achieved without a positive response from you, the CPC464 user. As a member of a User Group you will learn much more and get greater enjoyment from your computer than you would isolated at home. So, if you can bear to tear yourself away from this first edition of The Amstrad User for a moment, drop us a line now and indicate where and when you would like to meet other users in your area. This won't mean you will automatically become the branch convenor, that will be up to the Group as a whole to decide, but it will get the ball rolling.

By the way, don't forget that there is a period of about 2 weeks between The Amstrad User going to press and distribution commencing, so to ensure that any information for publication is not held over to the following month, it is best to submit by the last day in any month.

WELCOME

In the last few days, we are pleased to advise that the ball has started to roll in Western Australia, ACT and Victoria.

Western Australia

Bob Harwood would like to establish an Amstrad user club covering the Rockingham and Kwinana areas. He can be contacted during office hours on (09) 527 1777, or you may wish to write to him at 29 Millgrove Avenue, Cooloongup, 6168.

Dave Andersen has written to us as the local organiser of the 'Central Wheatbelt Amstrad User Group of W.A.'. You may write to Dave at 6 Kitchener Road, Merredin, 6415 but no personal callers please. He advises that the aim of the group is to "assist in the promotion of sales of all material relating to the Amstrad computer, and to assist in the education of all Amstrad users, be they business or private".

A.C.T.

John Payne has offered to establish a group in Canberra. You can contact him on (062) 82 2277 or write to Electricland Pty Ltd., 28 Townshend Street, Phillip, 2606. John is probably already well known to Amstrad users in Canberra, and appears to have a number of ideas in the establishment of the group.

Victoria

Mrs. G. Champan has kindly offered the use of her home to establish a group in South Clayton. Anyone wishing to organise an inaugural meeting may ring Mrs. Chapman on (03) 551 4897.

We look forward to publishing more 'ball rollers' in the next edition of The Amstrad User.

·COMPETITION·

$2500 worth of prizes to be won over four classes

Class 1
Best overall program
Wins an AWA Video Recorder

Class 2
Best amusement/adventure
Wins a new DD1 disk drive

Class 3
Best educational software
Wins a new DD1 disk drive

Class 4
Best business software
Wins a new DD1 disk drive

How to enter

Think about your program and map it out in a series of events or features. Write the program onto cassette based around these events and check that the program runs as intended. Once you are satisified, send a copy of the cassette in a suitable envelope along with the following:

1 A brief summary of the program in 500 words or less.
2 A clear program listing if available.
3 A stamped, self addressed envelope of adequate dimensions if you would like your entry returned.
4 Your name and address.

You may make as many submissions as you want, but no entrant may win more than one prize.

Conditions of Entry

1 All entries must run on a CPC464, and must include a cassette copy of the program (plus loading instructions where necessary), a brief summary of the program and its purpose and, if possible, a full listing.
2 All entries must arrive by 15th May 1985, and winners will be printed in the July edition of The Amstrad User.
3 The decision of the judges is final.
4 It is a condition of entry that all entrants have exclusive ownership of the copyright of the material submitted, and the winners agree to assign all copyright in the winning submissions to The Amstrad User. Where the entrant is more than one individual, then one person must be nominated and empowered to act on behalf of the entire group. All entrants must undertake not to submit the same or a similar program to any other magazine, publisher or organisation until after the announcement of the winning entries.
5 We, The Amstrad User, may offer to publish programs other than the winners in the magazine or as commercial software, in which case we will agree terms on an individual basis with the author(s) concerned. We reserve the right to amend, alter or revise any program we publish.
6 No employees of The Amstrad User or Strategy Publications, or their relatives may enter this competition.
7 The Amstrad User cannot be held responsible for any loss or damage to any submission.
8 No entrant may win more than one prize.

The Trials of Arnold Blackwood

A review by Philip Riley

I've just 'died' again. That makes the score: Trials = 3, Me = 0. Not that The Trials of Arnold Blackwood keep score, it's much too sophisticated for mundane score keeping, rather, it is an adventure game, and a most unusual one at that.

Arnold Blackwood awakes from a knock on his head to find himself suffering from amnesia, although he does remember his name and the fact that he is in the middle of Lord Erebus' estate. The eerie music adds to the atmosphere. Your help is enlisted to get him out of the mess, so you jump on his shoulders and proceed N, S, E, or W.

The game is well written with clear graphics and a good sense of humour. A sensible feature is a display that shows you your previous moves, and a facility to save a partly completed game. It follows the now standard 'two word' commands and uses some 80 verbs and 50 nouns. This allows the operator to be less than precise in providing instructions, and use 3-characters only for each verb and noun. But the game is full of wit and certainly uses the sound generator to good effect. The fact that you don't know what Arnold Blackwood and yourself are trying to achieve makes the game that more interesting. Being faced with problems such as:

Will Arnold eat the other half of the kipper?
Will Mexican Pete find his true vocation in life?
Does the ORACLE tell the truth?
What is in the mysterious chest? (In fact where is the mysterious chest?!)

It is very difficult to tear yourself away from the game and thereby ruin your train of thought.

For those who enjoy adventure games, devious in their nature, yet humorous, thought provoking yet entertaining, I would seriously recommend The Trials of Arnold Blackwood.

Now for another attempt, and this time when I see the Brick Wall I'll ....BANG HEAD?

The Learning Centre

Each month we will provide explanations and techniques to develop the skills of new programmers. In this first article certain ASCII codes and CHR$ are discussed, and a method for saving memory.

ASCII codes

Certain keys on the CPC464 have specific code values. Here are some examples:

KEY	CODE
ENTER	13
TAB	9
CLR	16
DEL	127
COPY	224
SPACE BAR	32

To illustrate how to check if a particular key has been pressed the following simple program can be entered.

```
1Ø CLS
2Ø A$=INKEY$:IF A$="" THEN
   2Ø
3Ø A=ASC(A$)
4Ø IF A=32 THEN 6Ø
5Ø GOTO 2Ø
6Ø PRINT"YOU PRESSED THE
   SPACE BAR"
   RUN
```

The screen will clear and the program "hang" (that means the program will not proceed any further) until it receives an input. Line 2Ø will loop within itself until a key depression is detected. Line 4Ø will check if the ASCII code generated by the key depression has a value of 32, and force the program to Line 6Ø to display the message if the check is positive. If not positive, the program will drop through to Line 5Ø which puts it back to Line 2Ø to await another key depression.

Try using different code values. You may also adapt this routine to check for Y or N in your programs.

CHR$

If you have not discovered the PRINT CHR$ commands yet, you may be in for a surprise. Some of them are quite useful, so it's worth experimenting. Here is a list of some of them.

PRINT CHR$(1) – Displays a small arrow

PRINT CHR$(4) – Clears the screen and returns cursor to the top left corner of the screen.

PRINT CHR$(7) – Sounds a 'beep'

PRINT CHR$(12) – Same as PRINT CHR$(4)

PRINT CHR$(14) – Sets the background and character colours to yellow regardless of the previous colours.

PRINT CHR$(19) – Clears the screen but the cursor remains static.

PRINT CHR$(24) – Reverses the background and character colours. (Enter this command again to return to original colours).

PRINT CHR$(29) – Sets the border flashing pink and grey.

PRINT CHR$(3Ø) – Moves the cursor to the top left corner of the screen but does not clear the screen.

PRINT CHR$(31) – Moves the cursor half way down the screen without clearing the screen.

Saving Memory

You will find that as you write larger programs, the memory space becomes important. This is especially relevant to adventure programs. So here are some points to remember.

EXAMPLE 1

```
1Ø FOR T=1 TO 5
2Ø PRINT"HELLO"
3Ø NEXT
```

EXAMPLE 2

```
1Ø FOR T=1 TO 5:PRINT
   "HELLO":NEXT
```

These two examples perform identically, but Example 1 uses 36 bytes of memory and Example 2 uses only 28 bytes. The reason for this is that each line number consumes 5 bytes of memory. By compressing the coding onto one line, you can save 10 bytes. But before you race to the phone to tell us we can't add up, the other two bytes are lost from printing the COLONS between the commands.

EXAMPLE 3

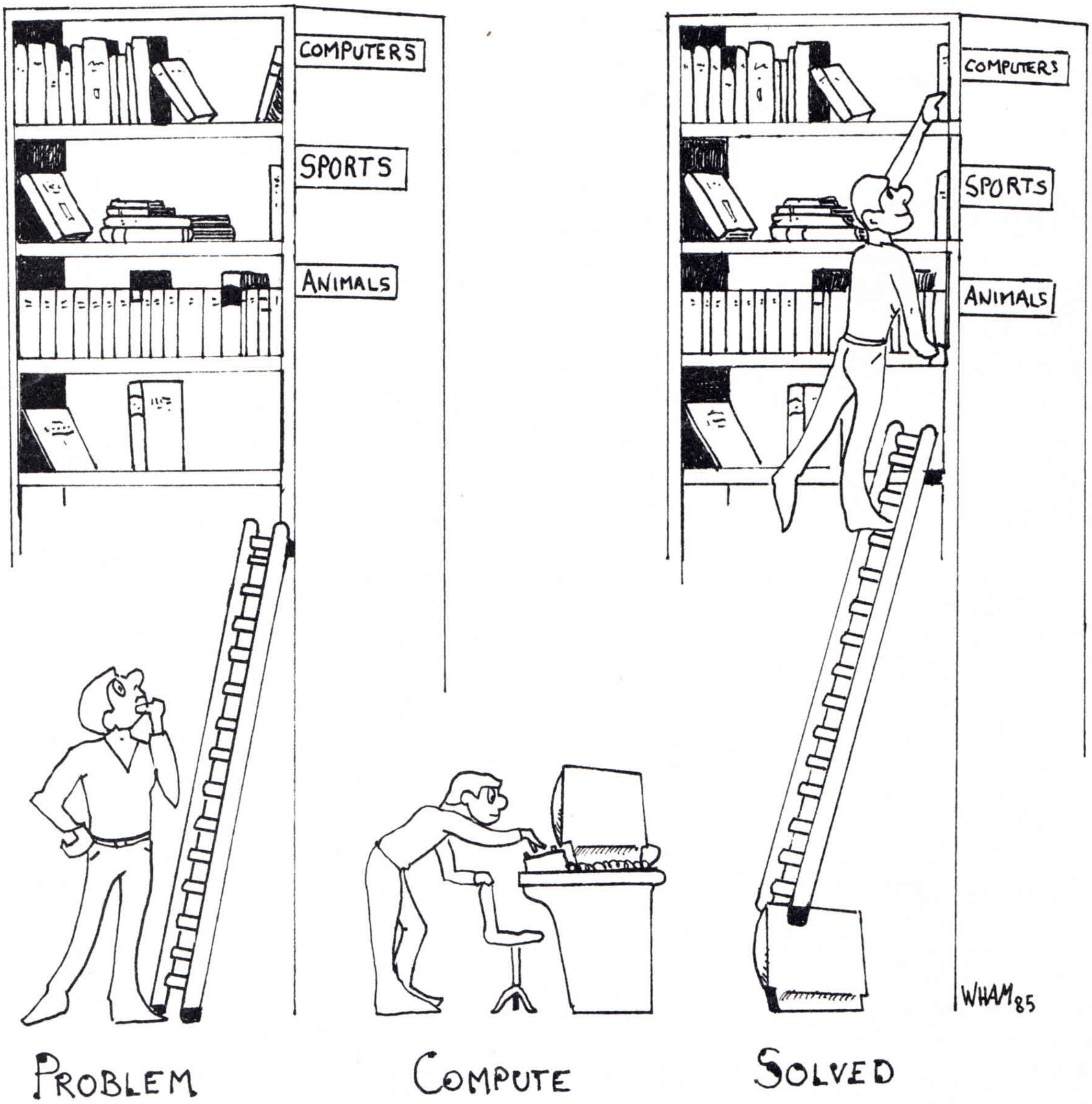
```
  10 IF A$="T" THEN 50
```

EXAMPLE 4

```
  10 IF A$="T"THEN 50
```

The difference in memory taken between Example 3 and 4 is 1 byte. It is not necessary to put a space after quotation marks. Try some examples for yourself and use the PRINT FRE(0) instruction to check the available memory after each test.

Note – more experienced users are very welcome to impart some of their knowledge to newcomers by submitting simple but useful routines, hints or explanations to this article.



Users' Hot-Line

A number of 'burning' questions have been raised in correspondence we have received over the past few weeks. Can you help other users and provide the answers?

Dave Anderson of Merredin, W.A. has a problem in saving and retrieving data to cassette.

"I've tried the OPENOUT "filename" and PRINT#9 but to no avail. When I try to get back the data, there appears to be nothing there. There's something on the tape but it's not my data, help me please ....."

Another problem is how to separate the integer from the whole of the number in order to use what's left.

Example: A=3Ø.6212:Z=A*1ØØ:
X=Z-INT(Z)
therefore X should equal Ø.12, but no,
X=Ø.119999886!

"Can any of your other readers make any sense of that, I can't?"

John Harris of Diannela, W.A. needs "information on the availability of a suitable, proven modem and interface that would be compatible with the Amstrad CPC464". We hope to have an answer from AMSTRAD themselves next month, but in the meantime perhaps someone can give John some advice. Our information is that Protek Computing of Edinburgh, UK are planning to make a modem available for the CPC464 but we are not sure when.

Another technical question comes from D.J. Walsh of Currajong, Qld. He requests any information on the availability of an interface for "a communication receiver for the reception of Fax or RTTV".

Mrs. Beryl Ferguson of Ayr, Qld. would like to know if there is any software available which will record details of memberships "and possibly a money management program to help in deciding how and when to invest". This could be an opportunity for someone to fill a gap in the software market for the CPC464.

Although most of our energy has gone into solving the logistics and teething problems of the first edition of The Amstrad User, we have been able to provide answers to some questions. These can be found in the "QA" section.

If you have any problems you feel could be answered by other users, send them in to "Users' Hot-Line", The Amstrad User, Suite 4a, 33-45 The Centreway, Blackburn Road, Mt. Waverley, Victoria 3149.

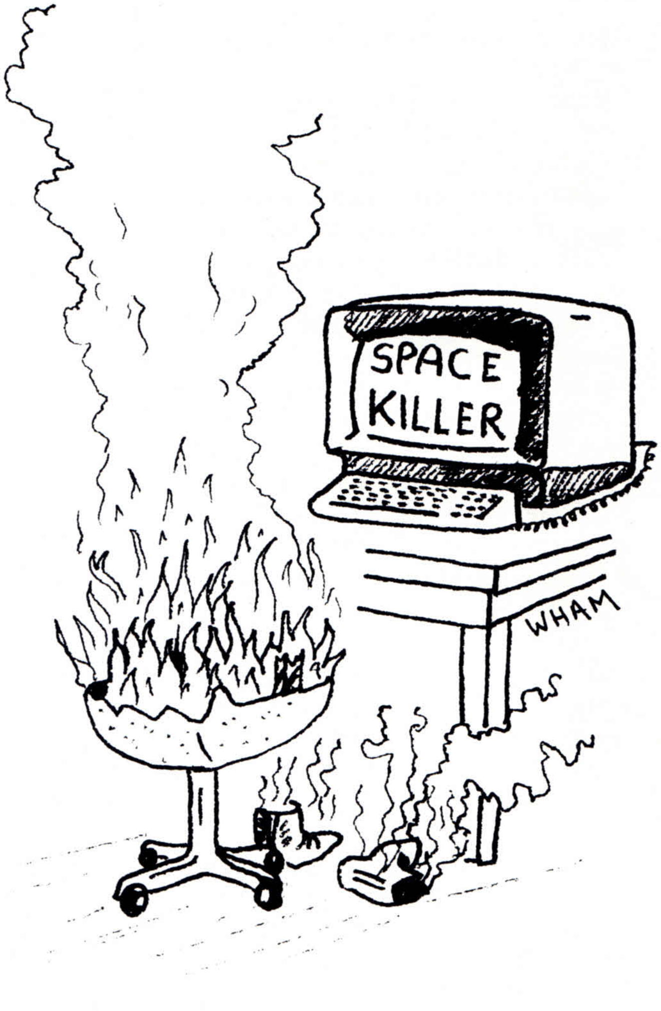
Bytes and Pieces

Eventually to be hints and tips from readers to make programming of the CPC464 a little easier, we start things off with the following snippets.

- For those people with big fingers who press the Escape Key instead of a '1', try entering KEY DEF 66,1,Ø
  The Escape key will now only operate with either CTRL or Shift.
- If you have been experimenting with user-definable characters with the Symbol after command to create a new character set, then try CALL &BAØA. This will divide and rearrange each character.
  A complete hard reset will occur by entering CALL Ø within a program.
- The treatment by the Basic interpreter of hexadecimal numbers as signed integers may cause problems. For example PRINT &DDDD returns -8739. To get over this you can use 1Ø DEF FNhex(h)=-65536*(h<Ø). FNhex(h) will provide the corrected decimal value for h. To prove this, PRINT FNhex(&DDDD) should return 56797. You will also find that this same function will provide the right values for binary numbers using the '&X' format.
- Here are two small, but useful, programs. As the REM statements suggest the first is probably the world's shortest word processor and the second a 'Day for a given Date' finder.

```
1 REM This program will find the DAY
for any DATE after 1753. Very useful
for Calendar printing etc.
The actual subroutine begins at line 3ØØØ.
2 REM Written by Les Aston Aug 1st 1984
1Ø CLS
1ØØ WINDOW 1Ø,3Ø,8,16
11Ø CLS: INPUT"Month (1-12)";m
12Ø INPUT"Day (1-31)";d
13Ø INPUT"Year ";y
135 PRINT
14Ø IF y<1753 THEN PRINT"Invalid Date"
15Ø GOSUB 3ØØØ
16Ø PRINT:GOTO 11Ø
3ØØØ d$=SUNMONTUEWEDTHUFRISAT":M$=
     JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC"
     :M$=MID$9M$,(M-1)*3+1,3):IF M<>5 THEN M$=M$+"."
3Ø1Ø M$=M$+STR$(D)+","+STR$(Y):M=M-2:IF M<1 THEN
     M=M+12:Y=Y-1
3Ø2Ø Y%=Y-INT(Y/1ØØ)*1ØØ:M%=
     INT(2.6*M-Ø.19)+D+Y%+INT(Y/4ØØ)+
     INT(Y%/4)-INT(Y/4ØØØ)-2*INT(Y/1ØØ)
3Ø3Ø M%=M%-INT(M%/7)*7:D$=MID$(D$,M%*3+1,3)+","+m$
3Ø4Ø PRINT d$:FOR t=1 TO 4ØØØ:NEXT:RETURN
```

```
1 REM Probably the World's shortest
  word processor! Having entered the number
  of char. per line, type away and the
  text will be printed out each time ENTER is
  pressed twice.
2 REM the printer MUST be ready.
  This program is handy for speed
  and could be expanded in all directions!
5Ø CLS: INPUT"How many characters per line";c
6Ø CLS:q=Ø
8Ø a$=INKEY$:IF a$="" THEN 8Ø
81 q=q+1: IF q*2>c THEN PRINT CHR$(13):q=1
1ØØ PRINT#8,b$;:PRINT a$;
11Ø b$=INKEY$:IF b$="" THEN 11Ø
13Ø PRINT#8,a$;:PRINT b$;:GOTO 8Ø
```



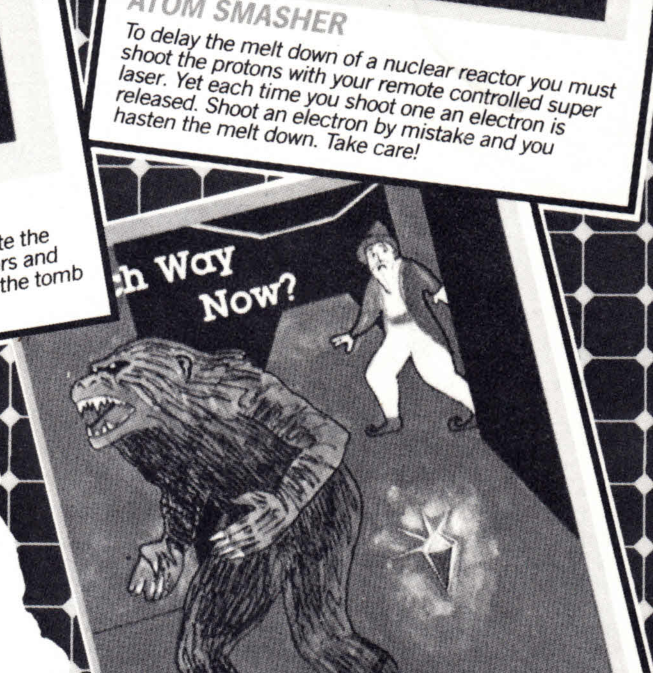
AMSTRAD UNLEASHES A NEW WORLD OF HIGH PERFORMANCE VIDEO GAMES.

OH MUMMY!

Untold riches are to be found if you have the courage to excavate the Pyramids. But dare you desecrate these ancient burial chambers and arouse the wrath of the Guardians? A breathless chase around the tomb could be your sole reward.

ATOM SMASHER

To delay the melt down of a nuclear reactor you must shoot the protons with your remote controlled super laser. Yet each time you shoot one an electron is released. Shoot an electron by mistake and you hasten the melt down. Take care!



SULTAN'S MAZE

Battle your way through a colourful maze in search of the Sultan's jewels. But beware, the mad Guardian is after you and your strength is running low. Can you gather the jewels and escape with your life?

With the brilliance of Amstrad high resolution graphics!

Welcome to the world of Amstrad home entertainment, featuring many of the most exciting and dynamic games ever devised for the home computer.

The substantial RAM in the Amstrad CP464 computer provides software authors with the scope they've been waiting for to develop ideas … ideas which they had literally run out of space for on lesser machines.

Furthermore, games originally written for other machines take on a startling new clarity when programmed for the Amstrad computer. The high resolution graphics (640 x 200 pixels) coupled with the precision of the Amstrad CTM640 RGB monitor offer visibly superior screen images. So you'll get more out of your Amstrad computer in games, educational programs, even business software.

For full details of Amstrad software for home and business, ask at your nearest Amstrad dealer.

STAR COMMANDO. The galaxy is under attack. Pirates and raiders have brought trade to a standstill. A successful mission will earn you your Commando's Wings ... if you're quick and brave enough.

GEMS OF STRADUS. Your quest is the treasure of the Great AM. Your search takes you through a myriad of rooms and obstacles. Avoid unpleasant surprises and you may reach your goal...

LASER WARP Can you survive the intergalactic aliens and invaders despatched by the Master? Can you reach level 10 and destroy him? Even if you do, he reappears. Look out!

XANAGRAMS. A word game for all ages, combining the excitement of hangman with the fun and challenge of anagrams and crosswords. Stimulating, and a real test of skill.

ADMIRAL GRAF SPEE. You command the Admiral Graf Spee on a search and destroy mission through the North and South Atlantic. You have the firepower to succeed, but stay alert — enemy ships lurk, ready to destroy you.

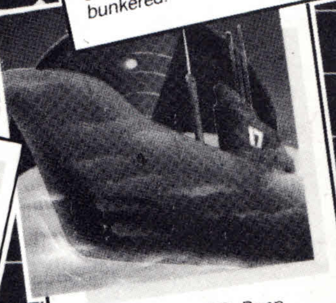
HAUNTED HEDGES. Trapped within a bewildering maze of ghostly 3D hedges, you chase after a fortune in scattered gold coins. At every turn, spooks appear to frustrate your attempts.

CRAZY GOLF. Can you master the crazy electronic golf course? You decide on the strength and direction of your shots, but be careful or you'll end up well and truly bunkered.

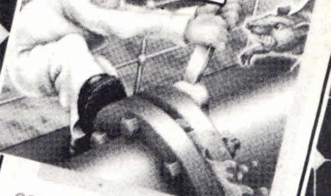
ALIEN BREAK-IN. You've been left to guard Earth's uranium supply against the invading Zargons. Shoot them up! Knock them down! But you'll get no points for destroying their Mother Ships.

ROLAND IN THE CAVES. Roland's in trouble again. Can you guide him past flesh-eating plants and the hungry pterodactyl, away from the strange, superbly detailed cave into which he has fallen?

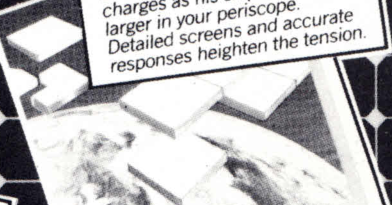
ELECTRO FREDDY. Pack the products, load the conveyors, ship the goods! Your one-man battle against electronic trickery could save your fellow workers from redundancy — if you're clever enough.



HUNTER KILLER. Deep under the sea your S-type submarine seeks out the enemy. Beware the depth charges as his ship looms ever larger in your periscope. Detailed screens and accurate responses heighten the tension.

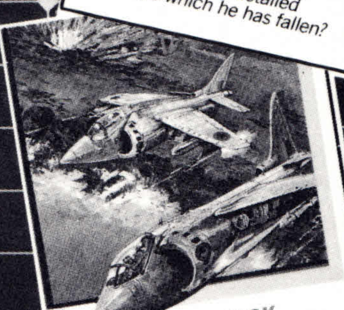


SPANNERMAN. Armed only with a spanner, you race to repair the leaky coolant pipes in a nuclear reactor. Vicious rats, falling debris and rising floods conspire to frustrate your every move.



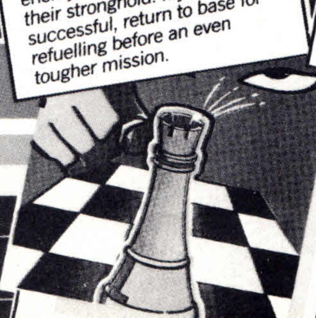
CUBIT. The space age game of noughts and crosses. Pit your skills against friends, or battle the computer in this 3D test of skill and perception.

ROLAND ON THE ROPES. Roland is trapped at the bottom of a mysterious tomb. You must guide him to safety past mummies, ghosts, rats and deadly drops of acid.



HARRIER ATTACK Guide your Harrier through enemy territory to destroy their stronghold. If you're successful, return to base for refuelling before an even tougher mission.

CODENAME MAT Can you save mankind from the deadly Myons? It's our last hope, as you take control of the USS Centurion for an incredible 3D journey through the solar system.



AMSTRAD

Distributed throughout Australia by

AWA-THORN

TPB278

D.I.Y. Typewriter

This program sets up the CPC464 as a simple to use editing typewriter system (a poor person's word processing system). It provides facilities to store and retrieve text to cassette or disk, cut and insert lines, re-type lines of text, and display on screen in the 'what you see is what you get' format.

It's user-friendly with menu prompts at the top of the screen. There's a panic feature that gets you out to the main menu without loss of data - simply press **[ESC]** twice, then the small **[ENTER]** key on the numeric keypad to return to the menu.

The program does not feature a full screen editor, but uses the same facilities as the BASIC line editor. Each line is numbered, and menu functions manipulate text by use of this index number.

Notes:

1. Don't use double quote marks " when the document is to be stored onto tape or disc. The computer interprets these as delimiters, and will terminate the line at the first set of quote marks.

2. When storing the file to disc, make sure that the filename chosen conforms to the AMSDOS & CP/M standard of up to eight permitted characters, plus a three character subscript: eg `ABCDEFGH.DOC`

3. To clear a document that that has been printed or filed, press **[ESC]** twice and re-`RUN` the program. Press the smaller **[ENTER]** key if you want to return without wiping off the document memory.

4. During text input, typing `E` on its own at the beginning of a line, then pressing **[ENTER]**, will return you to the menu..

The program operates with all Centronics interface printers. If you are using an EPSON compatible printer with features such as paper out sensing, auto form feeds, you may wish to override these functions by inserting lines 322-326.

```
322 PRINT#4,"PRINTER IS NOT ON LINE":SOUND
     1,200:PRINT#8,CHR$(13);:SOUND 129,0
324 PRINT#8,CHR$(27);CHR$(56);:REM
     DISABLE PAPER OUT ON EPSON
326 PRINT#8,CHR$(27);CHR$(79);:REM
     DISABLE FORM FEED ON EPSON
```

If you are using a green monitor, then....

```
60 MODE 2:BORDER 0:INK 0,0:INK 1,26:
    PAPER 0:PEN 1:GOSUB 1370
```

....will produce the best display combination.

Commentary

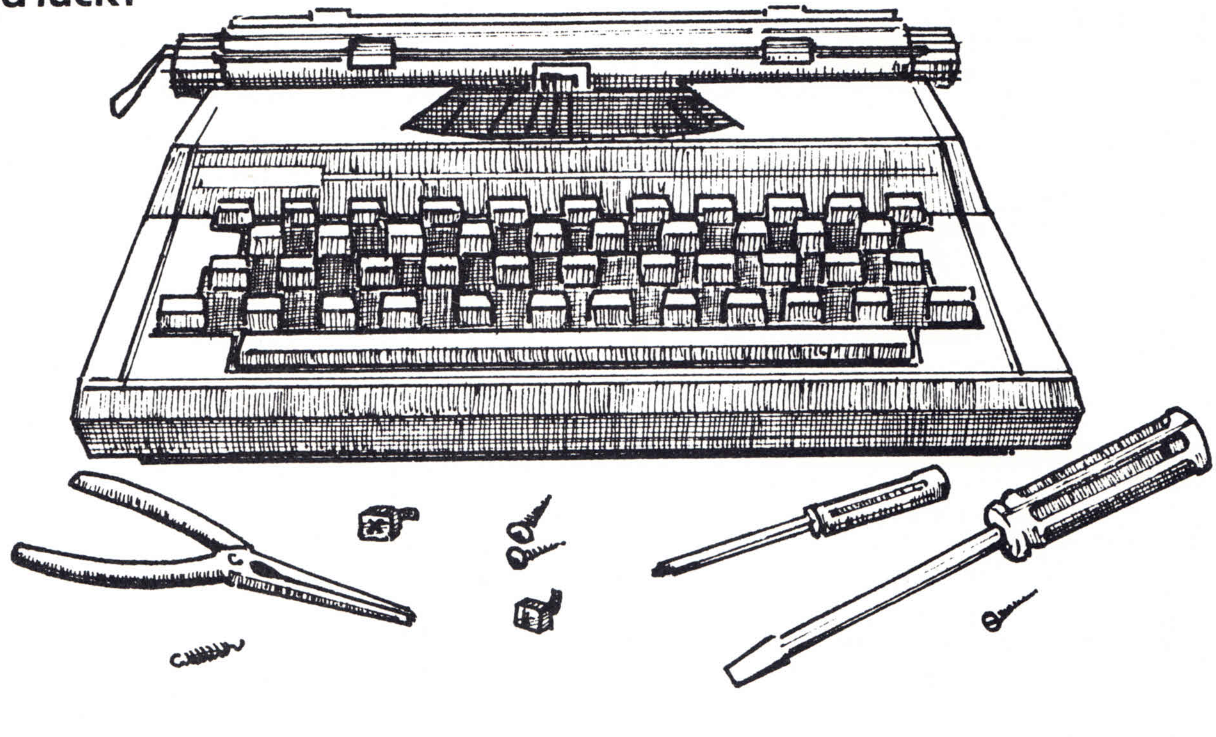
40-50	Force all user accessible variables to integer. Redefine keys. Set error trap to cope with invalid user input.
60	Set colours, mode and windows.
70	Allocate a file buffer for cassette (or disc) in the memory.
80-120	Option to load an external file.
130	Error trap point.
140-180	Read in external file
190-210	Set up conditions for text input.
220-280	Type in text.
300-460	Print out text.
480-600	Display text on screen.
620-690	Retype previous line of text
710-780	Add to existing text.
800-870	Store text onto cassette or disc.
890-1080	Insert text lines.
1100-1150	Cut text lines.
1170-1320	Menu.
1390-1470	Subroutine to display a line of text.
1490-1610	Subroutine to input a line of text and indicate remaining lines.
1630-1690	Accept user input for starting and finishing points.

```
10 ' D.I.Y. TYPEWRITER by IVOR SPITAL
20 ' **** (c) AMSTRAD PLC. 1984 ****
30 '
40 DEFINT F,I,P,S,X:KEY DEF 18,1:KEY 139,"CLG:GOTO 1170"+CHR$(13)
50 ON ERROR GOTO 130:KEY 140,"       ":KEY DEF 68,1,140
60 MODE 2:BORDER 26:INK 0,26:INK 1,13:PAPER 0:PEN 1:GOSUB 1340
70 OPENOUT "FILEBUF":MEMORY HIMEM-1:CLOSEOUT
80 SOUND 1,20:PRINT #4,"DO YOU WANT TO LOAD AN EXTERNAL FILE (Y/N)?"
90 F$=INKEY$:IF F$="" THEN 90
100 IF UPPER$(F$)="Y" THEN 140
110 IF UPPER$(F$)="N" THEN 190
120 GOTO 90
130 SOUND 1,200:PRINT #4,"INVALID ENTRY!":FOR T=1 TO 2000:NEXT:RESUME
140 SOUND 1,20:INPUT #4,"ENTER NAME OF FILE ";G$
150 SOUND 1,20:PRINT #4,"INSERT DISC OR CASSETTE FILE THEN PRESS ANY KEY"
160 IF INKEY$="" THEN 160
170 WINDOW SWAP 0,4:OPENIN G$:INPUT #9,L,PW:MXL=INT(70000/PW):DIM TEXT$(MXL)
180 FOR X=1 TO L:INPUT #9,TEXT$(X):NEXT:CLOSEIN:WINDOW SWAP 0,4:GOTO 1170
190 SOUND 1,20:INPUT #4,"ENTER PRINTER WIDTH (10-70)";PW
200 IF PW<10 OR PW>70 THEN 190
210 MXL=INT(70000/PW):DIM TEXT$(MXL):F=0:B=24:K=0:GOSUB 1340
220 GOSUB 270:FOR X=1 TO MXL:GOSUB 1490:IF F=1 THEN 1170
230 IF K=1 THEN 250
240 NEXT
250 SOUND 1,200:PRINT #4,"CANNOT CONTINUE - NUMBER OF LINES EXHAUSTED"
260 FOR T=1 TO 3000:NEXT:GOTO 1170
270 PRINT #4,"E=END TEXT"
280 PRINT #2,STRING$(23,60);:RETURN
290 '
300 'PRINT-OUT
310 '
320 IF L=0 THEN 1310
330 SOUND 1,20:PRINT #4,"PRINT-OUT FULL DOCUMENT (Y/N)?"
340 H=0:B$=INKEY$:IF B$="" THEN 340
350 IF UPPER$(B$)="Y" THEN START=1:FINISH=L:GOTO 390
360 IF UPPER$(B$)="N" THEN GOSUB 1630
370 IF H=1 THEN 390
380 GOTO 340
390 PRINT #4,"PRINTER IS NOT ON-LINE":SOUND 1,200:PRINT #8,CHR$(13);
400 SOUND 129,0:WIDTH PW:CLG:FOR X=START TO FINISH:PRINT #4,"LINE";X
410 PRINT #8,TEXT$(X):GOSUB 1390:NEXT
420 SOUND 1,20:PRINT #4,"REPEAT PRINT-OUT (Y/N)?"
430 C$=INKEY$:IF C$="" THEN 430
440 IF UPPER$(C$)="Y" THEN 390
450 IF UPPER$(C$)="N" THEN 1170
460 GOTO 430
470 '
480 ' DISPLAY
490 '
500 CLG:IF L=0 THEN 1310
510 SOUND 1,20:CLS#4:LOCATE #4,20,1:PRINT #4,1;SPC(3)"[ENTER]":LOCATE #4,1,1
520 INPUT #4,"DISPLAY FROM LINE ";START:IF START=0 THEN START=1
530 IF START>L OR START<0 THEN 510
540 PRINT #4,"HOLD DOWN SPACE BAR TO DISPLAY"SPC(35)"Q=QUIT DISPLAY";
550 SOUND 1,20:FOR X=START TO L
560 E$=INKEY$:IF E$="" THEN 560
570 IF E$=" " THEN 600
580 IF UPPER$(E$)="Q" THEN 1170
590 GOTO 560
600 GOSUB 1390:NEXT:GOTO 1170
610 '
620 ' RETYPE
630 '
640 IF L=0 THEN 1310
650 SOUND 1,20:INPUT #4,"ENTER LINE NUMBER TO RETYPE ";X
660 IF X<1 OR X>L THEN 650
670 PAPER #3,1:PEN #3,0:GOSUB 1390:LOCATE #3,1,24:LOCATE #0,1,24
```

```
680 PRINT #3,"RETYPE:>";:GOSUB 280:PAPER #3,0:PEN #3,1
690 LINE INPUT #0,"",TEXT$(X):CLG:GOSUB 1390:GOTO 1170
700 '
710 ' ADD
720 '
730 IF L=0 THEN 1310
740 CLG:IF L>5 THEN S=(L-4) ELSE S=1
750 FOR X=S TO L:GOSUB 1390:NEXT:GOSUB 270
760 F=0:B=24:FOR X=(L+1) TO MXL:GOSUB 1490:IF F=1 THEN 1170
770 IF K=1 THEN 250
780 NEXT:GOTO 250
790 '
800 ' STORE
810 '
820 IF L=0 THEN 1310
830 SOUND 1,20:INPUT #4,"ENTER CHOSEN NAME FOR FILE ";G$
840 SOUND 1,20:PRINT #4,"INSERT DISC OR CASSETTE FILE THEN PRESS ANY KEY"
850 IF INKEY$="" THEN 850
860 WINDOW SWAP 0,4:OPENOUT G$:WRITE #9,L,PW:FOR X=1 TO L
870 WRITE #9,TEXT$(X):NEXT:CLOSEOUT:WINDOW SWAP 0,4:GOTO 1170
880 '
890 ' INSERT
900 '
910 IF L=0 THEN 1310
920 SOUND 1,20:INPUT #4,"INSERT AFTER WHICH LINE NUMBER ";START
930 IF START<0 OR START>L THEN 920
940 SOUND 1,20:INPUT #4,"INSERT HOW MANY LINES (50 MAX) ";I:IF I>50 THEN 940
950 FOR X=L TO START+1 STEP -1:TEXT$(X+I)=TEXT$(X):NEXT:L=L+I
960 FOR X=START+1 TO START+I:TEXT$(X)="":NEXT:CLG
970 PRINT #4, "LINES INSERTED ="I:FOR T=1 TO 1000:NEXT:Y=L
980 SOUND 1,20:PRINT #4,"DO YOU WANT TO TYPE INTO THE INSERT? (Y/N)"
990 I$=INKEY$:IF I$="" THEN 990
1000 IF UPPER$(I$)="Y" THEN 1030
1010 IF UPPER$(I$)="N" THEN 1170
1020 GOTO 990
1030 IF START>5 THEN V=(START-4) ELSE V=1
1040 FOR X=V TO START:GOSUB 1390:NEXT:GOSUB 270
1050 F=0:B=24:FOR X=START+1 TO START+I:GOSUB 1490:L=Y
1060 IF F=1 THEN TEXT$(X)="":GOTO 1170
1070 IF K=1 THEN 250
1080 NEXT:SOUND 1,20:PRINT #4,"INSERT FILLED":FOR T=1 TO 2000:NEXT:GOTO 1170
1090 '
1100 ' CUT
1110 '
1120 IF L=0 THEN 1310
1130 GOSUB 1650:PRINT #4,SPC(45)"CUTTING IN PROGRESS - PLEASE WAIT"
1140 FOR N=1 TO (FINISH-START)+1:FOR X=START TO L:TEXT$(X)=TEXT$(X+1):NEXT
1150 L=L-1:LOCATE #4,1,1:PRINT #4,"RE-LOCATING LINE";X SPC(3):NEXT:CLG
1160 '
1170 ' MENU
1180 '
1190 GOSUB 1340:SOUND 1,20:PRINT #4,"SELECT:>";
1200 PRINT #4,TAB(12)"D=DISPLAY  P=PRINT-OUT  R=RETYPE  ";
1210 PRINT #4,"A=ADD  C=CUT  S=STORE  I=INSERT"
1220 A$=INKEY$:IF A$="" THEN 1220
1230 IF UPPER$(A$)="D" THEN 480
1240 IF UPPER$(A$)="P" THEN 300
1250 IF UPPER$(A$)="R" THEN 620
1260 IF UPPER$(A$)="A" THEN 710
1270 IF UPPER$(A$)="C" THEN 1100
1280 IF UPPER$(A$)="S" THEN 800
1290 IF UPPER$(A$)="I" THEN 890
1300 GOTO 1220
1310 SOUND 1,200:PRINT #4,"NOTHING ENTERED IN MEMORY"
1320 FOR T=1 TO 2000:NEXT:GOTO 740
1330 '
```

```
1340 ' DEFINE WINDOWS
1350 '
1360 WINDOW #0,9,(PW+8),3,25:WINDOW #2,(PW+9),(PW+9),3,25:WINDOW #3,1,8,3,25
1370 WINDOW #4,1,79,1,1:PAPER #4,1:PEN #4,0:CLS#4:RETURN
1380 '
1390 ' DISPLAY SUBROUTINE
1400 '
1410 LOCATE #0,1,24:LOCATE #3,1,24
1420 PRINT #3,X;:PRINT #3,TAB(8)">";:LT=LEN(TEXT$(X))
1430 IF INT(LT/PW)=LT/PW AND LT>0 THEN PRINT #0,TEXT$(X);
1440 IF INT(LT/PW)<>LT/PW OR LT=0 THEN PRINT #0,TEXT$(X)
1450 IF INT(LT/PW)<>LT/PW AND LT>PW THEN FOR N=1 TO INT(LT/PW):PRINT #3:NEXT
1460 IF INT(LT/PW)=LT/PW AND LT>PW THEN FOR N=1 TO (INT(LT/PW)-1):PRINT #3:NEXT
1470 RETURN
1480 '
1490 ' LINE INPUT SUBROUTINE
1500 '
1510 COUNT=INT((FRE(0)-1000)/PW)
1520 IF COUNT<1 THEN CLS#2:CLS#3:K=1:RETURN
1530 LOCATE #4,44,1:PRINT #4,"NUMBER OF LINES REMAINING ="COUNT;SPC(3)
1540 LOCATE #3,1,24:LOCATE #0,1,B
1550 PRINT #3,X;:PRINT #3,TAB(8)">";:LINE INPUT #0,"",TEXT$(X)
1560 L=X:IF UPPER$(TEXT$(X))="E" THEN L=X-1:F=1:RETURN
1570 LT=LEN(TEXT$(X)):B=24
1580 IF INT(LT/PW)=LT/PW AND LT>0 THEN B=23
1590 IF INT(LT/PW)=LT/PW AND LT>PW THEN FOR N=1 TO (INT(LT/PW)-1):PRINT #3:NEXT
1600 IF INT(LT/PW)<>LT/PW AND LT>PW THEN FOR N=1 TO INT(LT/PW):PRINT #3:NEXT
1610 RETURN
1620 '
1630 ' START/FINISH SUBROUTINE
1640 '
1650 SOUND 1,20:INPUT #4,"COMMENCING LINE NUMBER ";START
1660 IF START<1 OR START>L THEN 1650
1670 SOUND 1,20:INPUT #4,"FINAL LINE NUMBER ";FINISH
1680 IF FINISH<START OR FINISH>L THEN 1670
1690 H=1:RETURN
```

Good luck!



QA

The Amstrad User postbag is a rich source of prompts for further information on the CPC464. Naturally, in the interests of brevity we reserve the right to edit letters, and if you would like a specific response you must include a stamped addressed envelope.

Q *I have just bought a CPC464 and when reading through the instruction manual I have noticed the word 'stream' popping up everywhere. As the manual does not seem to explain just what a 'stream' is, perhaps you can.*

JK, Footscray

A A 'stream' is a channel that is used by the computer to denote a particular 'area' of the computer. For instance, to print anything on the printer, you would use LPRINT#8,"MESSAGE". In relation to the screen, the stream refers to previously defined Text Windows that have been set using the Window#n command, e.g. to write to Window#4 you would use PRINT#4, "MESSAGE".

If the text window has not been defined then the text output will be in the top left corner of the screen. This is because all undefined windows default to cover the whole text stream. Also, omitting the stream number causes the output to go to stream#Ø or window#Ø, in other words, the default window.

Q *When listing programs, I can find no way of preventing double spaced printing with my SHINWA CP-80. The printer has been checked, the DIP switches have been set correctly, so it seems that the CPC464 is sending two line feeds on carriage return.*

Any suggestions?

A This is a very common problem, and is simply resolved by cutting the connection to pin 14 on the Centronics plug. The CPC464 sends a single line feed/carriage return, but there are many printers out there whose interface requires pin 14 to be open circuit in order to suppress the automatic line feed.

Q *I have a problem: the machine refuses to save and load successfully at the 2000 baud rate, although it works fine at 1000 baud. What should I try?*

A As the speed goes up, the tape quality becomes increasingly important. Try a high spec TDK tape and see if this cures your problem. Failing that, you will have to seek assistance from the store where the computer was purchased.

Bear in mind that an inferior or very old tape may very rapidly 'clog' the head and render it considerably less efficient, so always use only tape of known pedigree, and beware of some of the proprietory tape head cleaners; the fluid can dissolve the plastic casing!

Q *I have noticed an annoying background hiss/whine which is generated continuously whilst the computer is running. The noise increases when a program is being executed and reaches a peak when a part of a program like the following is reached*

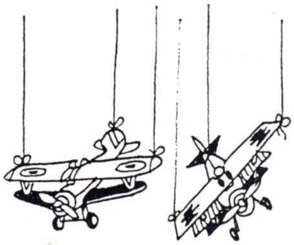
```
25Ø a$=INKEY$
26Ø IF a$="" THEN 25Ø
```

I would be grateful if you would inform me as to the nature of this noise, and whether it be a fault specific to my own machine, or a design fault.

ADW

A The noise is the sound of Arnold thinking to himself. It's generated by the same process that exists to allow the user to hear when the tape is reading or writing. It is at a very low level, which may be completely silenced by volume control if required. It's not a fault: it's a feature to provide unobtrusive reassurance that 'things are happening'.

You will notice that the sound extension socket is completely free from all background noise – so if you want sound in your programs without any of the informative little burbles that tell you when the processor is working hard, use this socket in conjunction with an amplifier. The overall effect from the extension sound socket can be quite breathtaking when played through a stereo system.



Mass Storage Media

A personal account of the history of Mass Storage by Arthur Harris

I am writing this article from my own experiences, and admit that there were, and still are, variations too voluminous to mention. Coupled with the fact that time tends to dull the memory, I expect that there are likely to be a few inaccuracies on which people, more knowledgeable than myself in this particular area, will pick me up.

The areas covered in this article span the earliest days of computers through to the latest published infomation that I have seen on developments in mass storage for mini and micro computers.

Because the earliest computers were programmed by wire links and switch settings, the earliest storage of programs was the handwritten instructions for these links and switch settings, using pen and paper. It is not certain which came next. It is believed to be a drum recorder developed by IBM. This drum had a pitifully small capacity of something like 256 bytes. It is not known at what stage magnetic tape was first used. It is assumed that the earliest tape usage was in serial format where information was stored one bit at a time, in order.

Somewhere in there, able to be used because the ubiquitous Teletype was used as an input/output device, is the punched paper tape. This medium consists of a length of paper approximately 25 mm wide, in which 8 lines of holes could be punched, each hole representing one bit of a byte. A fine-toothed wheel in the centre of the reader/punch provided the drive to move the tape past the pins. I can only speculate on its operation. Obviously, the presence of an ON bit activates a burst of current, forcing the appropriate pin to punch a hole in the paper during the recording of information. During reading, the pin must have some pressure onto the paper. When it encounters a hole, the pin falls into it, producing a short burst of current. Whether the withdrawal mechanism is mechanical or electrical, I don't know. I have used this form of storage, both in Tasmania (at TCAE) and in Victoria (on a Teletype in the PWD).

With independently controlled heads, the access to a particular piece of information, anywhere on the pack, is extremely fast, despite the vast amount of information held on each pack.

IBM developed the 8 inch (200 mm) floppy disk drive as the next stage after its drum storage. For quite a long time, this was the industry "standard" and is still used extensively on minicomputers for security backup for the hard disks that form their primary mass storage. Some microcomputers, for business purposes, still rely exclusively on 200 mm floppy disk drives. When configured for double-sided, double-density operation, these disks achieve a capacity of 1.2 Mb.

Fortunately, for the microcomputer industry, in 1976, Shugart released the 5¼ inch (133 mm) floppy disk drive. The 200 mm drive is too expensive for home computer enthusiasts. The first release of the 133 mm drive was a 35 track, single-sided, single density format with the gloriously large capacity of 89,600 bytes. Subtract from this the space occupied by the Disk Operating System and the result is a total of about 79 Kb. It is a constant source of annoyance to me that, until recently, this was the industry standard for these disks and programs were written for this format without allowing the user to alter the format to one of the many formats that emerged.

The various configurations that appeared included 40, 77 and 80 track in both single- and double-density and single- and double-sided formats. This results in various capacities of up to 720 Kb per single drive.

Now we come to the latest developments. The mass storage devices are proceeding in two directions. Both have the same objective of providing increasingly larger amounts of information in an ever decreasing volume of space.

One technology that I haven't mentioned so far is the bubble memory. It is not known whether this is really a mass storage device or a replacement for system RAM. Sharp is one of two companies who are using this technology which received a lot of publicity a few years ago but died as a real issue because of reported high costs and lack of results in development. There have not been any reports of significant advances in this technology for some time.

The latest development I have seen, appeared in an article in the December, 1984 issue of Electronics Today International (Australian version). It was announced that the 3M Corporation had released details of a construction technique, for disks, called "stretched surface" recording. It is reported to reliably store 5 Mb on each side of a 133 mm floppy. The development is aimed to increase this capacity by five to ten times in the near future.

Briefly, the construction consists of a rigid substrate with slightly raised edges. A flexible, coated membrane, 60 microns thick, is stretched across both faces of the disk, with the raised edges ensuring that the membrane is held clear of the rigid substrate. The read/write heads "fly" just above the magnetic surface (typically about 75 microns in the Winchester and hard disk technology). The disk rotates at about 3400 rpm and air pressure build-up between the head and the membrane causes the membrane to deflect, producing a dimple and giving a clearance of about 5 microns.

As the disk rotates, the dimple moves with the head, restoring itself to a flat surface with an action likened to that of a trampoline. This action throws off dirt particles and any other debris, resulting in an essentially clean surface. Dirt particles are largely responsible for data errors in read/write operations. The resilience of the membrane makes these disks far less prone to impact damage.

Another paper-based storage method is the punched cards. At RMIT, during my studies there (ending in 1979), I used hundreds of these cards. We were lucky enough to have card-punch machines that were used like a typewriter. Our fellow students at Swinburne had to punch their cards by hand. Each card holds 80 characters and each character is designated by up to three holes located in 12 rows on the card. Each column represents a character. This method ensures that the card is not excessively weakened by punched holes. Reading the information is accomplished by passing the card through a machine which uses high pressure air to indicate the location of the holes. The system requires meticulous maintenance as the reader is prone to blocking of its holes with the punchouts from the cards.

Yet another storage system which survived for a considerable period was called CRAM (for Card Random Access Memory). This system used a pack of cards coated with magnetic material. Each card was about 375 mm long by about 75 mm wide and stored something like 256 bytes. A pack consisted of up to 255 cards, giving a storage capacity of nearly 64 Kb. The cards were hung on 8 square pins and rotation of a combination of pins so that their sides were vertical, released a particular card. This card was then carried over the reading/writing heads by a revolving drum and returned to the pack. The noise from the clacking of the pins rotating the holder opening to receive the card after its trip around the drum, was reputedly deafening. This sytem was developed in the early 1960's and believed to be still in use on some sites in Melbourne until about 5 years ago.

The next development was the multiplatter hard disk. This form of mass storage is still in use, probably having undergone many stages of refinement to reach its present stage of development. Present day hard disks have capacities of up to 800 Mb on a single pack. Each platter has its own reading and writing head. The information is recorded in concentric circular tracks, called cylinders.

One path of this development began with the IBM development of the Winchester disk technology. It is possible to obtain a Winchester disk drive with a capacity of 80 Mb. The size of the Winchester drives ranges from 25 mm, with 1 Mb capacity, up to 300 mm, with the most common size being 200 mm. The larger capacity disk drives are heading into laser technology. One article that I read referred to the development of the laser technology but made the point that although a single 300 mm disk could store 2000 Mb, it was not possible to erase or overwrite information. A month later, a second article announced that the technology had been developed which allows the erasing and overwriting of information on laser disk. A substance (plastic?) had been found which could be made to change from an amorphous form to a crystalline form by changing the frequency of the irradiating laser. The disk was still capable of storing 2000 Mb of information on a 300 mm disk. The first commercial release of the laser technology was announced to take place in February of 1984 at the astonishingly low price of $7 per Mb.

The other path involves the increase in capacity of existing disks and the miniaturisation of disk drives. Apart from the 25 mm Winchester drive mentioned above, it is now easy to obtain the 133 mm (5.25 in.) disk drives with a capacity of 1 Mb. There has also been a proliferation of drives using disks between 75 mm and 100 mm diameter. The main sizes are 79 mm, 82.5 mm, 89 mm, 94 mm and 99 mm. There is no definite indication to which of these sizes is likely to become the de facto industry standard. As somewhat of a pointer, a couple of major computer marketing firms are using the 89 mm drives. There is also R & D being carried out on this size, including a development announced by Toshiba about 2 years ago. Toshiba announced a single-sided 89 mm drive storing 3 Mb. The means of achieving this was by vertical magnetisation in a layer 0.5 micron thick. The gap in the recording/writing heads was only 0.4 microns wide. (1 micron = 1E-6 metre).

A track density of 345 tracks per inch (compared to 96 tpi for conventional floppy disks) will be used on the early units. On later issues, this spacing is expected to be halved (to 690 tpi) to give 48 Mb capacity for fixed disk versions and 37 Mb capacity for removeable disk versions.

That takes us from the beginning up to the latest developments. The big question is where are we heading? What can we expect to become the standard in the future? It would be a very game person who would attempt to answer this question. My guess is for some development of the technology about which we have not yet heard. Two things are certain – the size of the device will be very small and its capacity will be very large.

An Overview of the DD1

by Kevin Poynton

We recently took possession of one of the first Disk Drives for the CPC464 to arrive in Australia. The drive is being used for software development. Because we are a Software House we felt it best that a full review should only be written by an end user and this article is, therefore, a simple overview of the product and designed to give you some indication of what you get for your money.

Firstly, AWA-Thorn advise me that the drives will be available in small quantities from mid-January onwards and will retail at $449. What do you get? The unit comes complete with interface, connecting cable, the drive, one 3" disk with AMSDOS, CP/M and DR LOGO and a ring bound manual of approximately 50 pages similar in style to the Basic Manual provided with the machine. The cable provided with the drive allows connection of either DD-2 or a standard 5¼" drive. The only bad news is that you lose 1280 bytes of RAM to allow for the ROM based drivers provided in the interface. This means that some existing software will not be able to be transferred to disk.

The AMSDOS is very simple and easy to use and should not provide too many headaches for even the novice user. The only problem I have encountered so far was in using our GRAPHIC MAGIC program to save screen dumps to disk. After many hours a sudden flash of inspiration solved the problem by the following method:

```
INPUT "FILENAME", A$
A$ = "!" + A$
SAVE A$, Ø,$CØØØ, $4ØØØ
```

Why the "!" is required I don't know but if not used the DOS appears to double the first character of a filename and exclude the last character when loading back. The following is typical of what occurs.

```
LOAD "FROG"
FFRO.  NOT FOUND
```

Please remember this problem does not occur in normal usage – only when saving screen dumps.

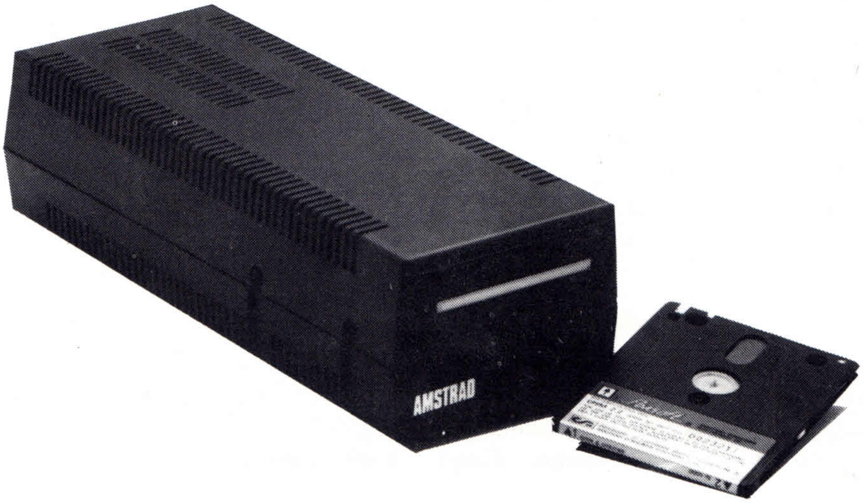
Other than the examples given, the AMSDOS operates perfectly and is very fast, typically loading about 30k in approximately 5 or 6 seconds.

The version of CP/M provided is the 48k 2.2 version and is complete in every respect. AMSDOS operates under the shell of CP/M which may be used to copy AMSDOS files from disc to disc.

The manual provided is adequate for the novice users but higher level users would probably wish to purchase the technical manual available from AMSOFT.

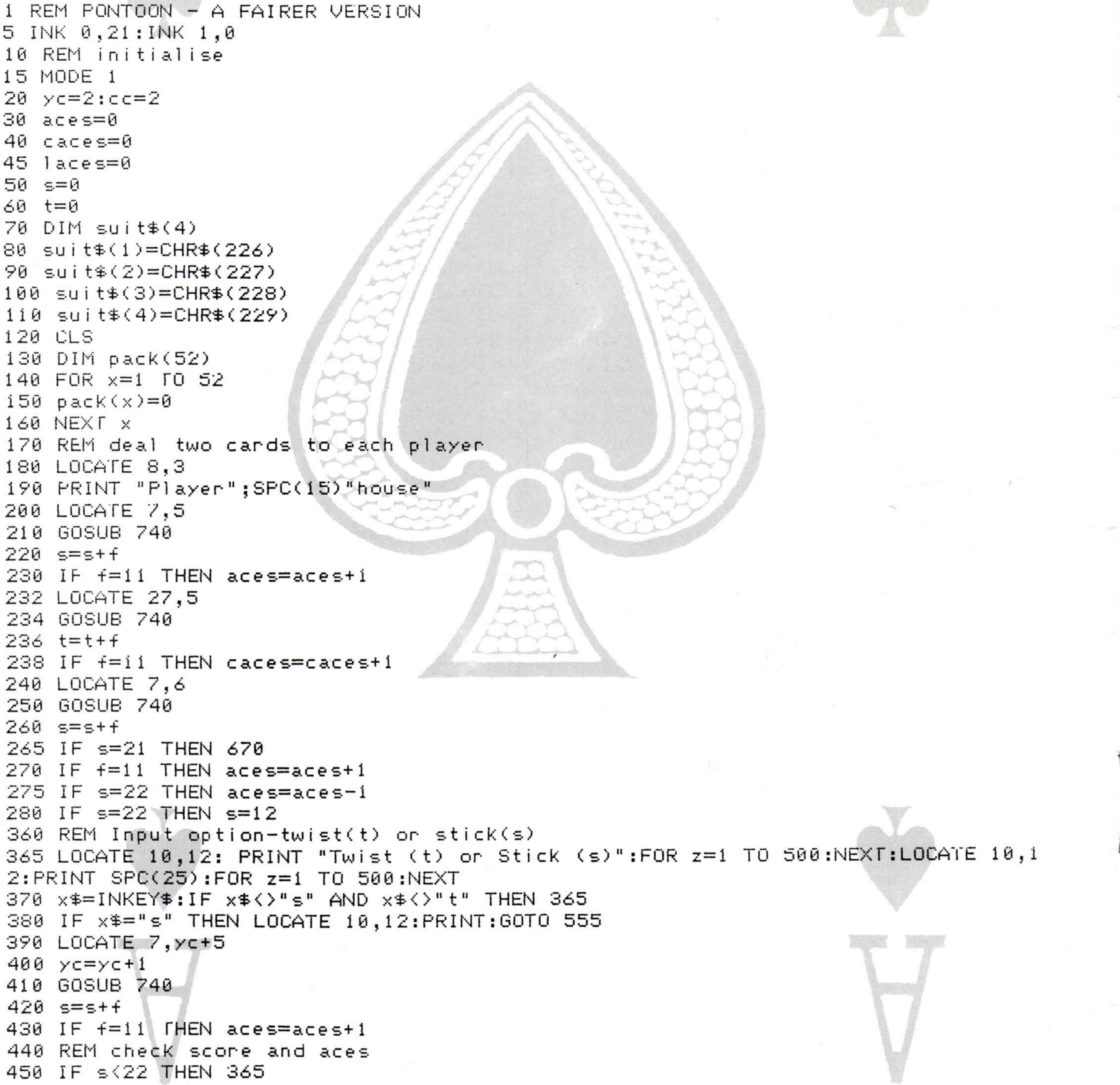
It is divided into six chapters plus appendices. Foundation Course is introductory and covers the set-up procedures and a description of AMSDOS and CP/M. Chapter 1 gets into the 'nitty gritty' of using the system and covers Back-Ups, Utilities, Formatting etc. Chapters 2 and 3 cover AMSDOS and CP/M respectively and include plenty of examples of normal usage. Chapter 4 is the largest chapter and gives a fairly good introduction to LOGO. Chapter 5 provides technical information for the advanced user. All in all, the manual is well thought out and should be of assistance to most users. If you require more information on CP/M or LOGO I would suggest you purchase technical manuals available from AMSOFT or dig around in your local bookstore for books on the subject of interest.

Is it worth the money? – A very definite yes! Anyone who gets bored loading cassettes or wishes to develop software should buy one for a change. The end-user can now purchase a full CP/M system (with the bonus of LOGO) with a truly fast disk drive for well under $1000. AMSTRAD and AWA are to be congratulated on bringing the serious user an excellent system at such a price.

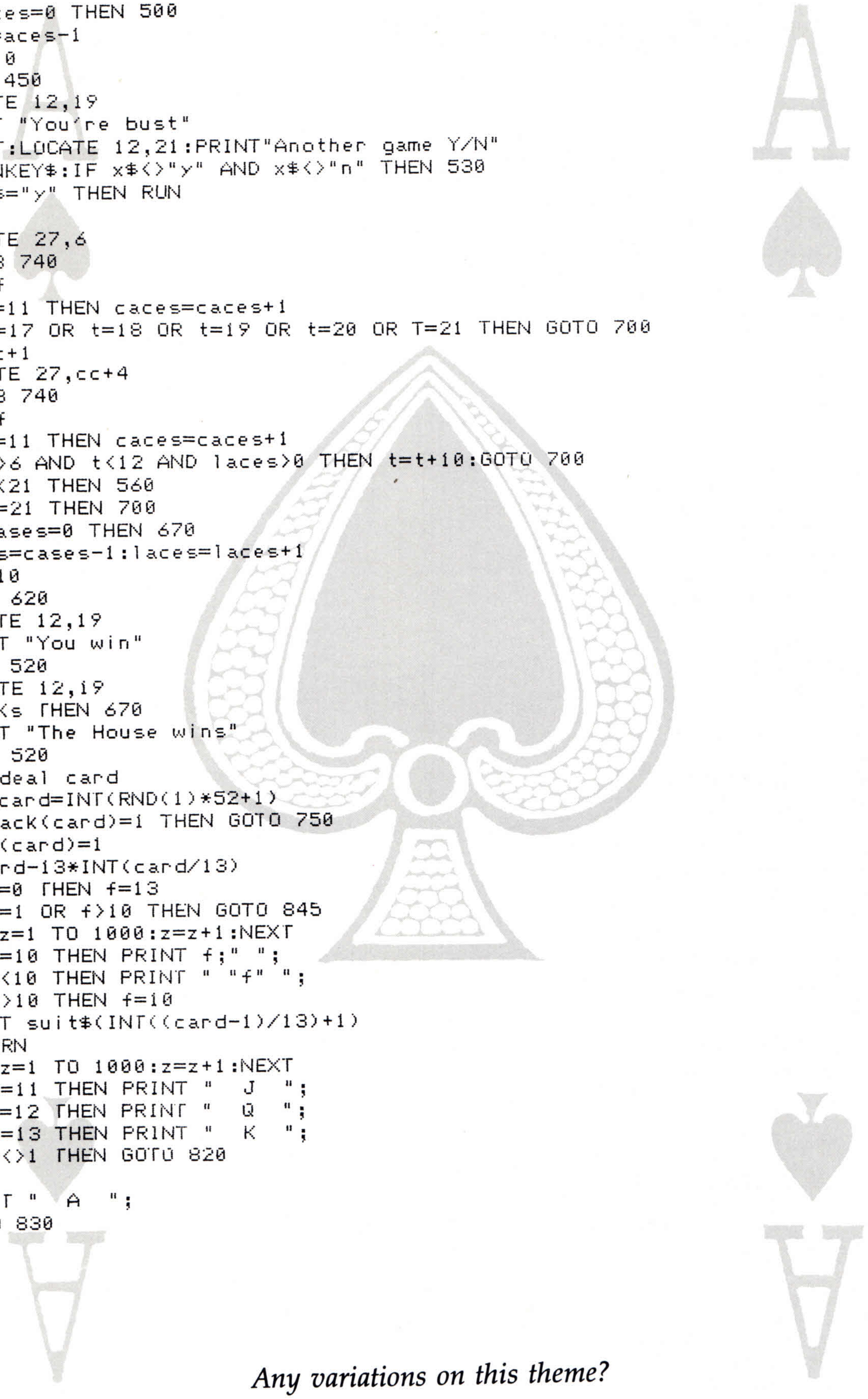


Pontoon – a Fairer Version

Stephen Greenwood, another ardent Amstrad admirer, discovered that the Pontoon program in the User Guide failed to give him a fair deal. In fact, the outcome seemed positively devious at times.



```
1 REM PONTOON - A FAIRER VERSION
5 INK 0,21:INK 1,0
10 REM initialise
15 MODE 1
20 yc=2:cc=2
30 aces=0
40 caces=0
45 laces=0
50 s=0
60 t=0
70 DIM suit$(4)
80 suit$(1)=CHR$(226)
90 suit$(2)=CHR$(227)
100 suit$(3)=CHR$(228)
110 suit$(4)=CHR$(229)
120 CLS
130 DIM pack(52)
140 FOR x=1 TO 52
150 pack(x)=0
160 NEXT x
170 REM deal two cards to each player
180 LOCATE 8,3
190 PRINT "Player";SPC(15)"house"
200 LOCATE 7,5
210 GOSUB 740
220 s=s+f
230 IF f=11 THEN aces=aces+1
232 LOCATE 27,5
234 GOSUB 740
236 t=t+f
238 IF f=11 THEN caces=caces+1
240 LOCATE 7,6
250 GOSUB 740
260 s=s+f
265 IF s=21 THEN 670
270 IF f=11 THEN aces=aces+1
275 IF s=22 THEN aces=aces-1
280 IF s=22 THEN s=12
360 REM Input option-twist(t) or stick(s)
365 LOCATE 10,12: PRINT "Twist (t) or Stick (s)":FOR z=1 TO 500:NEXT:LOCATE 10,1
2:PRINT SPC(25):FOR z=1 TO 500:NEXT
370 x$=INKEY$:IF x$<>"s" AND x$<>"t" THEN 365
380 IF x$="s" THEN LOCATE 10,12:PRINT:GOTO 555
390 LOCATE 7,yc+5
400 yc=yc+1
410 GOSUB 740
420 s=s+f
430 IF f=11 THEN aces=aces+1
440 REM check score and aces
450 IF s<22 THEN 365
```



```
460 IF aces=0 THEN 500
470 aces=aces-1
480 s=s-10
490 GOTO 450
500 LOCATE 12,19
510 PRINT "You're bust"
520 PRINT:LOCATE 12,21:PRINT"Another game Y/N"
530 x$=INKEY$:IF x$<>"y" AND x$<>"n" THEN 530
540 IF x$="y" THEN RUN
550 END
555 LOCATE 27,6
556 GOSUB 740
557 t=t+f
558 IF f=11 THEN caces=caces+1
560 IF t=17 OR t=18 OR t=19 OR t=20 OR T=21 THEN GOTO 700
570 cc=cc+1
580 LOCATE 27,cc+4
590 GOSUB 740
600 t=t+f
610 IF f=11 THEN caces=caces+1
615 IF t>6 AND t<12 AND laces>0 THEN t=t+10:GOTO 700
620 IF T<21 THEN 560
625 IF t=21 THEN 700
630 IF cases=0 THEN 670
640 cases=cases-1:laces=laces+1
650 t=t-10
660 GOTO 620
670 LOCATE 12,19
680 PRINT "You win"
690 GOTO 520
700 LOCATE 12,19
710 IF t<s THEN 670
720 PRINT "The House wins"
730 GOTO 520
740 REM deal card
750 LET card=INT(RND(1)*52+1)
760 IF pack(card)=1 THEN GOTO 750
770 pack(card)=1
780 f=card-13*INT(card/13)
790 IF f=0 THEN f=13
800 IF f=1 OR f>10 THEN GOTO 845
805 FOR z=1 TO 1000:z=z+1:NEXT
810 IF f=10 THEN PRINT f;" ";
815 IF f<10 THEN PRINT " "f" ";
820 IF f>10 THEN f=10
830 PRINT suit$(INT((card-1)/13)+1)
840 RETURN
845 FOR z=1 TO 1000:z=z+1:NEXT
850 IF f=11 THEN PRINT "  J  ";
860 IF f=12 THEN PRINT "  Q  ";
870 IF f=13 THEN PRINT "  K  ";
880 IF f<>1 THEN GOTO 820
890 f=11
900 PRINT "  A  ";
910 GOTO 830
```

Any variations on this theme?

Graphics Mode

Half the fun of using the CPC464 is discovering and experimenting with graphics. Here are two programs that do just that. The first has been written by Len Deitman and is based entirely upon the triangle. The second piece, written by Sydney Brown, is a demonstration of a moving horizon with the use of a Joystick.

Len Deitman's contribution

```
10 REM fill-shapes
20 DIM x(3),y(3),f(3)
30 MODE 1:ORIGIN 348,240:INK 0,12:BORDER 12
40 REM start of data read
50 READ s,p:IF s=1 GOTO 380 ELSE IF s=2 GOTO 300 ELSE IF s=-2 GOTO 340
60 REM read in polygon sides
70 DIM w(s),z(s):FOR n=1 TO s:READ w(n),z(n):NEXT
80 IF s=3 OR s=4 THEN x(1)=w(1):y(1)=z(1):x(2)=w(2):y(2)=z(2):x(3)=w(3):y(3)=z(3
):GOTO 150 ELSE GOTO 110
90 REM find polygon centre
100 x(2)=w(4):y(2)=z(4):s=-1:GOTO 150
110 ma=w(1):mb=w(1):mc=z(1):md=z(1):FOR n=2 TO s:ma=MAX(ma,w(n)):mb=MIN(mb,w(n))
:mc=MAX(mc,z(n)):md=MIN(md,z(n)):NEXT
120 x(1)=(ma+mb)/2:y(1)=(mc+md)/2:x2=w(1):y(2)=z(1):FOR n=2 TO s:x(3)=w(n):y(3)=
z(n):GOTO 150
130 x(2)=x(3):y(2)=y(3):NEXT:x(3)=w(1):y(3)=z(1):s=-1:GOTO 150
140 REM make triangle sides to fill
150 f(1)=SGN(y(1)):a=(0.1+FIX(ABS(y(1))))*f(1):f(2)=SGN(y(2)):b=(0.2+FIX(ABS(y(2
))))*f(2):f(3)=SGN(y(3)):c=(0.3+FIX(ABS(y(3))))*f(3)

160 cy=MAX(a,b,c):ay=MIN(a,b,c)
170 zy=ABS(ay):ya=INT(zy):n=(zy-ya)*10:ya=ya*f(n):xa=x(n):g=n:zy=ABS(cy):yc=INT(
zy):n=(zy-yc)*10:yc=yc*f(n):xc=x(n):n=6-n-g:yb=y(n):
xb=x(n)
180 yab=yb-ya:IF yab<1 AND yab>-1 THEN yab=1
190 ybc=yc-yb:IF ybc<1 AND ybc>-1 THEN ybc=1
200 yac=yc-ya:IF yac<1 AND yac>-1 THEN yac=1
210 xab=xb-xa:xbc=xc-xb:xac=xc-xa:mxc=xac/yac:dxb=xab/yab:dxc=xbc/ybc
220 xab=xb-xa:xbc=xc-xb:xac=xc-xa:mxc=xac/yac:dxb=xab/yab:dxc=xbc/ybc
230 REM fill triangles
240 FOR n=0 TO yab:y=ya+n:x=xa+n*mxc
250 MOVE x,y:DRAW xa+n*dxb,y,p:NEXT
260 FOR n= 1 TO ybc:MOVE x+n*mxc,y+n:DRAW xb+n*dxc,y+n,p:NEXT
270 REM return from filling
280 IF s=4 GOTO 100 ELSE IF s>4 GOTO 130 ELSE IF s=-1 OR s=3 THEN ERASE w,z:GOTO
 50 ELSE RETURN
290 REM read for lines
300 READ mx,my:MOVE mx,my
310 READ dx,dy: IF dy<>999 THEN DRAW dx,dy,p: GOTO 310
320 IF dx=2 GOTO 300 ELSE GOTO 50
330 REM read for circles
340 READ r,sd,x(1),y(1)
350 DEG:f=360/sd:x(2)=r+x(1):y(2)=y(1)
360 FOR h=1 TO sd:x(3)=COS(h*f)*r+x(1):y(3)=SIN(h*f)*r+y(1)
370 GOSUB 150:x(2)=x(3):y(2)=y(3):NEXT:GOTO 50
380 PRINT(""):END
390 REM sky       ---  sample Data ---
400 DATA 4,2,294,36,294,161,-351,161,-351,33
410 REM right wall
```

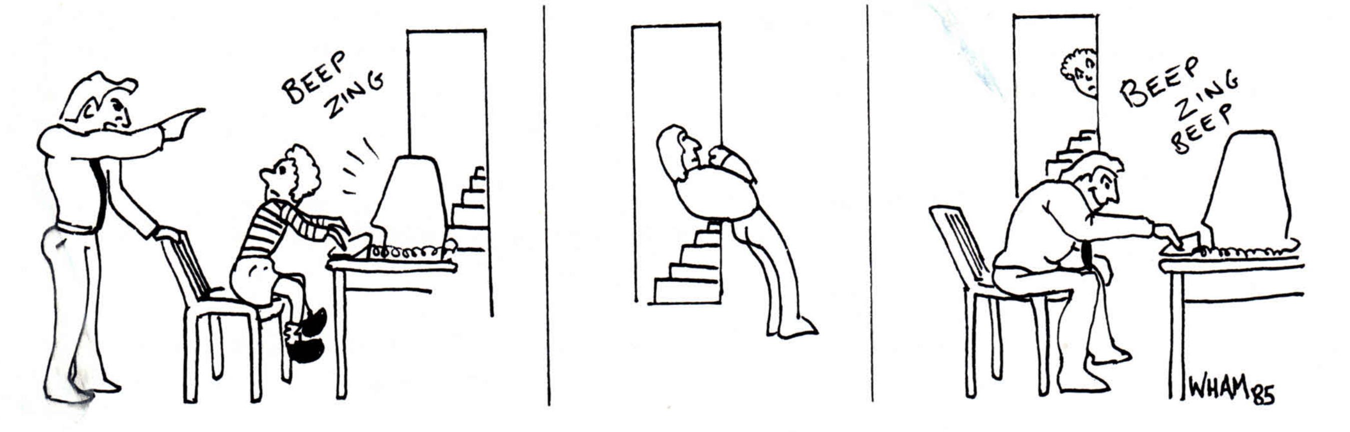
```
420 DATA 4,1,-17,-115,146,-5,159,100,-20,54
430 REM left wall
440 DATA 4,1,-189,87,-20,54,-17,-115,-171,-37
450 REM red roof
460 DATA 3,3,-189,87,-44,135,-20,54,4,3,-20,54,-44,135,38,140,159,100
470 REM road left
480 DATA 4,3,43,-206,-6,-258,-397,35,-365,40
490 REM road right
500 DATA 4,3,-13,-172,247,34,278,35,43,-206
510 REM line
520 DATA 2,0,-44,135,-20,54,-17,-115,-1,999
530 REM 4 circles
540 DATA-2,3,70,20,-260,-162,-2,2,70,30,196,-160,-2,1,30,15,220,98,-2,0,45,15,-2
70,98,1,0
550 REM
980 REM      -----------------------------------------------------
990 REM
1000 REM                FILLING POLYGONS AND CIRCLES
1010 REM
1020 REM Filling a square or circle with Move-Draw usually leaves dark dots at c
ertain angles. They may look like sunbeams in a sun
drawing but look like broken tiles on a house drawing.
1030 REM This program fills with horizontal lines which always light up.It fills
 triangles,so a triangle is filled in one go,a recta
ngle is two triangles.
1040 REM With a polygon of 5 or more sides,it will work out the centre of it and
 make triangles of each side and the centre.
1050 REM To fill a circle,you give it the radius,how many sides(up TO 255),the X
-Y centre position,PEN colour.    The program works
out the position of each side and fills in to the centre.
1060 REM For any shape of 3 sides and upwards the Data input starts with,the num
ber of sides,pen colour,then the X,Y,position of eac
h corner . At the end of the data list finish with  1,0.
1070 REM This will goto the End,or where you want to goto.  Take the data from t
he drawing in a circular direction either way.If you
 have a awkward backward angle side,make a seperate triangle of it.
1080 REM      Triangle :  Data 3,3,-189,87,-44,135,-20,54
              Rectangle:  Data 4,3,-20,54,-44,135,38,
140,159,100                         7 side polygon:  Data 7,3,x1,y1,x2,y2,----x7
,y7
1090 REM For circles the data input starts with -2,pen colour,radius,no of sides
 ,the centre X,Y,position.
                                        Circle   :  data -2,3,70,20,-260,-160 .

1100 REM For a line the data starts with a 2,pen colour,X1,Y1,X2,Y2,--Xn,Yn  .
 and other positions.At the end of line data put -1,
999 to get out of the line routine.
1110 REM If you need to move to a new line start put 2,999 at the end of the lin
e before the new one.        2 lines in different pla
ces                                          Line :  Data 2,3,x1,y1,x2,y2,2,999,x
1,y1,x2,y2,-1,999
1120 REM
1130 REM      -----------------------------------------------------
1140 REM
1150 REM         Do you feel the need of a ArcSin or ArcCos.
1160 REM
1170 REM The ArcSin and ArcCos functions are not supplied,but the following
         lines work very quickly :-
1180 REM         ASN(x) = ATN(x/sqr(1-x*x))
1190 REM         ACS(x) = 90-ATN(x/sqr(1-x*x))
1200 REM These lines can be DEF FN ,but have you noticed, the DEF FN must be def
ined in lines before you use it,unlike a GOSUB .
1210     -----------------------------------------------------
```

Sydney Brown's contribution

```
10 KEY 140,"mode 1:pen 1:ink 1,26"+CHR$(13)
20 REM*****************************************  Colour Demo By Sydney.H.Brown *
*****************************************
30 REM***************************************** Use Joystick to move the scene *
*****************************************
40 REM After the initial set-up nothing        is being plotted or erased only
       the colours are rotated.........

50 MODE 0:INK 0,0:INK 1,24:INK 2,3:INK 3,3:INK 4,3:INK 5,24:INK 6,3:INK 7,3:INK
8,3:INK 9,26:INK 10,26:INK 11,26:INK 15,24:CLS
60 INK 12,3:FOR w=0 TO 198 STEP 2:PLOT 0,w:DRAW 640,w,12:NEXT: PLOT 0,200:DRAW 6
40,200,15
70 xi=-320:FOR x=0 TO 640 STEP 32:FOR xx=0 TO 3:PLOT x+xx*8,198,xx+1:DRAW xi,0,x
x+1:xi=xi+16:NEXT:NEXT
80 yy=198:yi=2
90 FOR w=0 TO 3:PLOT 0,yy-(yi*w),w+5:DRAW 640,yy-(yi*w),w+5:NEXT:yy=yy-(4*yi):yi
=yi*2:IF yy-yi>-1 THEN GOTO 90
100 FOR w=0 TO 2:FOR ww=1 TO 20:ry=INT(RND(1)*200)+200:rx=INT(RND(1)*640):PLOT r
x,ry,9+w:NEXT:NEXT
110 x=1:y=1
120 IF INKEY(72)=0 THEN GOSUB 280
130 IF INKEY(73)=0 THEN GOSUB 340
140 IF INKEY(74)=0 THEN GOSUB 200
150 IF INKEY(75)=0 THEN GOSUB 260
160 IF RND(1)<0.5 THEN INK 9,0 ELSE INK 9,26
170 IF RND(1)<0.5 THEN INK 10,0 ELSE INK 10,26
180 IF RND(1)<0.5 THEN INK 11,0 ELSE INK 11,26
190 FOR w=1 TO 35:NEXT:GOTO 120
200 x=x+1:IF x>4 THEN x=1
210 IF x=1 THEN INK 2,3 :INK 3,3 :INK 4,3:INK 1,24
220 IF x=2 THEN INK 1,3 :INK 3,3 :INK 4,3:INK 2,24
230 IF x=3 THEN INK 1,3 :INK 2,3 :INK 4,3:INK 3,24
240 IF x=4 THEN INK 1,3 :INK 2,3 :INK 3,3 :INK 4,24
250 RETURN
260 x=x-1:IF x<1 THEN x=4
270 GOTO 210
280 y=y+1:IF y>4 THEN y=1
290 IF y=1 THEN INK 6,3 :INK 7,3 :INK 8,3:INK 5,24
300 IF y=2 THEN INK 5,3 :INK 7,3 :INK 8,3:INK 6,24
310 IF y=3 THEN INK 5,3 :INK 6,3 :INK 8,3:INK 7,24
320 IF y=4 THEN INK 5,3 :INK 6,3 :INK 7,3 :INK 8,24
330 RETURN
340 y=y-1:IF y<1 THEN y=4
350 GOTO 290
```



SAVE $450

Q. How many computer magazines would I have to buy to get at least 30 pages of information and program listings for the new Amstrad CPC464?

A. On average, most popular magazines will devote 2½ pages to the AMSTRAD CPC464. This means you will need to buy 12 magazines at a cost of around $40 per month, or **$480 per year!**

Q. Surely there must be a more sensible and cheaper way of getting the information I need?

A. There is.
THE AMSTRAD USER is a brand new monthly publication packed with articles, reviews, listings, hints etc. for Amstrad users only, and costs just **$30 a year**.

Q. How can I get a copy delivered to my home each month with an optional cassette containing all the program listings?

A. By completing and returning this order.

Please send me THE AMSTRAD USER for 12 months
Magazine only: $30 □ Magazine and cassette: $70 □
Payment by: Cheque □ Bankcard or Mastercard □
Card number __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ Expiry date .....
Name .... Phone ....
Address ....
.... Postcode ....
Signed ....

Return to THE AMSTRAD USER, Suite 4a, 33-45 The Centreway
Blackburn Road, Mt. Waverley, Vic 3149 Tel 03-233 9227

(OVERSEAS PRICES ON APPLICATION TO ABOVE ADDRESS)