apple user

Vol. 4. No. 3   March 1984   £1

-the new name for
Windfall

What follows Macintosh?

Cut-price Lisa launched

A sideways view of Visicalc

Play 20 Questions and
Noughts & Crosses

Create your own histograms

Studentdata, SunAccount,
Incredible Jack reviewed

Stringing along with Pascal

Critical path analysis on the Apple

Apple joins a pop group
Computer graphics help create a
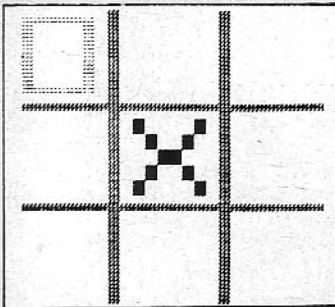new dimension in hi-tech music

apple user

Portable Apple to follow Macintosh

MACINTOSH has not even reached British shops yet — and already there is talk of another micro from Apple. It is a portable version of the Apple IIe.

The new portable has been called a variety of code-names during development, from Popcorn to Picasso.

As usual Apple refuse to give any details in advance and it is even rumoured that they changed the code name whenever they showed the prototype to a new group of people so that they would be able to identify the culprit should any leak occur.

However reduce the size of a IIe by half, give it a mouse, a built-in 5¼in disc drive, a 12-volt battery power supply and possibly even a flat screen with 80 column display and you could have Apple's next offering.

The new portable probably won't have the traditional seven interface card slots — it will utilise on-board I/O devices.

It is due for release in the spring and will cost "less than the IIe".

Apple expects to sell 350,000 Macintoshes this year.

Thousands of orders have already been placed in the UK — even though the machine will not be available until next month and no firm price has been announced.

For Lisa lovers a 70mbyte hard disc is on the cards and Apple is due to release a laser printer soon which can produce daisywheel quality output at speeds far faster than the fastest dot matrix machines on the market.

Already we have seen the upgrading (and the downpricing) of the souped-up Lisa range and Macintosh itself (see our report last month) will continue to generate excitement both through new software packages and through unusual applications.

Mice controls are now available for the II and III range and each month will see new software releases for these machines.

Apple IIe with mouse

Apple for Treasury Minister

JOHN Moore, MP for Croydon Central and Financial Secretary to the Treasury, has taken delivery of an Apple IIe system. He bought it to improve the service he gives his constituents.

He says he is delighted with the system and intends to use it for research, correspondence, information retrieval and record-keeping.
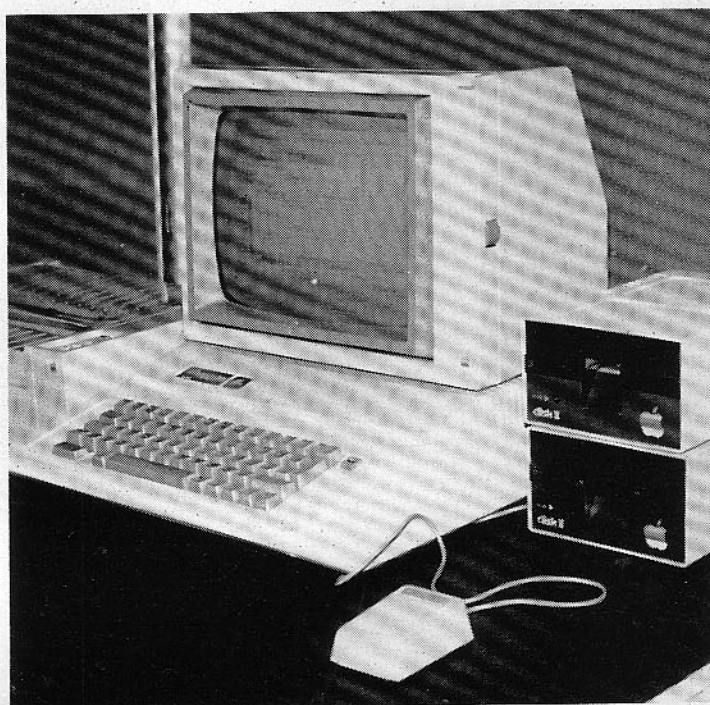
Now mice invade the IIe and III

A MOUSE on the IIe and III could be one of the biggest success stories in Apple history.

It means that on a humble early Apple users can run a remarkable equivalent to the LisaDraw package, called MousePaint, which incorporates pull down menus, overlapping windows and point-and-press cursor control.

Third party software houses are already working on word processing, database and other programs that will utilise the mouse on the Apple II and III in Lisa-like ways.

At around £150 including the MousePaint software, existing owners are likely to grab at the mouse — and so move into the mid-80s in style.

The mice themselves are,

in their own way, as impressive as Macintosh, and made as big an impact when demonstrated at the Macintosh launch.

The interface card connecting the mouse to the Apple contains 80k of operating system held in ROM plus its own processor.

This powerful combination of hardware devices gives the mouse on the Apple IIe and III a performance which has to be seen to be believed.

True it is a trifle slower than LisaDraw, and the resolution of the pictures produced is limited. But as with Macintosh, it allows users to perform surgery on individual pixels.

Portions of a picture can be selected and enlarged, the mouse being used to turn the pixels on or off.

Like Lisa, the package relies heavily on icons. Tools available are displayed down the left hand side of the screen in graphical form. Such everyday commodities as an indiarubber, pencil and magnifying glass are available.

There is even a "hand" icon which offers a novel approach to moving the graphics screen about.

One uses it to literally thumb through the MousePaint paper, just as one would with the real thing.

APPLE TO THE RESCUE

IF you want to get your hands on a new Macintosh quickly then it seems as though you'll have to break a leg — literally.

For Apple has come to the rescue of a road accident victim by sending him the company's latest — and most sought after machine — to "play with" while he's convalescing at home.

Mind you, he does also happen to be Pete Fisher — the Pete of Pete and Pam Computers — whose group now has an annual turnover in excess of £12 million.

"The arrival of the Macintosh is the only consolation I've had since the accident", admitted 34-year-old Pete.

It was on the evening of January 18 that he was knocked down by a car in the grounds of the National Exhibition Centre in Birmingham after visiting the Which Micro? show.

He sustained a broken left leg, damage to the toes on his right foot, a shattered right collar bone, severe lacerations and extensive bruising.

Rushed to Birmingham accident unit, he spent four days there before being transferred to hospital in Bury nearer his home.

Two attempts to reset his damaged leg failed and it became necessary to insert a steel plate.

He will be confined to his home until the plaster caste on his leg is removed in three months' time.

Unused to being housebound — he hasn't been ill for more than 20 years and had never been in hospital before — Pete Fisher is now helping run his business empire from the settee in his front room.

For he is linked by computer to his offices in Rawtenstall and London and so is able to take part in day to day decisions.

And — thanks to the generousity of Apple — he has also been kept busy appraising the new Macintosh.
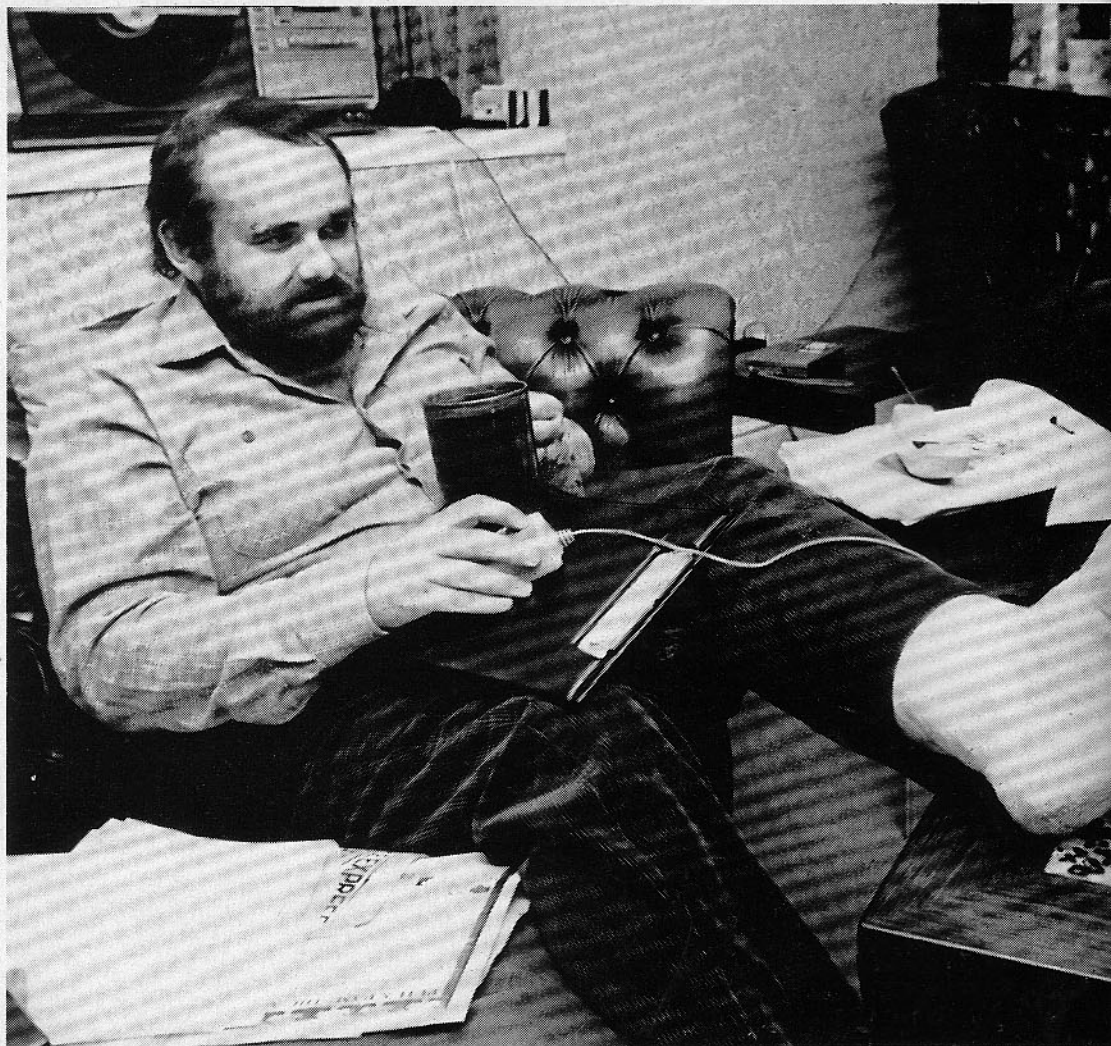
'Spectacular Lisa' scoops Rita award

APPLE'S Lisa has won the Rita award for the Office System of the Year, 1983.

The top honour (Rita stands for Recognition of Information Technology Achievement) is the British computer industry's equivalent of a Hollywood Oscar.

Lisa, one of eight award winners announced during the Which Computer? Show at the end of January, was described as the most innovative idea for office automation.

The judges praised its development potential, spectacular graphics, advanced technology and integrated software.

Sponsors of the award, Which Word Processor and Office System?, described Lisa as a worthy candidate in a world full of boring 16 bit personal computers equipped with WordStar and masquerading as small office systems.

Apple UK's managing director, Peter Cobb, who accepted the award agreed, naturally, that Lisa was a remarkable technological breakthrough.

However he added that it was only the beginning.

"The range and scoope of Lisa technology will be extended throughout 1984 to break down further the barriers between the individual and the computer", he said.

...and so does the 800

ANOTHER Rita award winner was Micronet 800 whose telesoftware venture was recognised for "System Innovation in 1983".

The company was praised for "using the technology to propagate the technology".

Micronet 800 is an information provider on British Telecom's Prestel service. An Apple can access the system via a modem to read pages of news, views and advice about micros and to download software programs.

APPLES BLOSSOM!

MORE than 100,000 Apple IIes were sold by Apple worldwide in December.

Naturally John Sculley, Apple president, was delighted.

"We were extremely pleased with the momentum demonstrated by the Apple II as we approached the announcement of our new 32-bit products", he said.

He attributed the record sales, in part, to a reduction in price and strong dealer promotion programs.

However the good sales figures couldn't hide relatively poor financial results achieved by Apple in the last quarter.

Net sales for the quarter were $316.2 million (£224.26 million), a 48 per cent increase over the same period a year ago.

Net income however was $5.8 million (£4.11 million – a huge drop of 75 per cent compared to the $23.5 million (£16.67 million) figure in the same period last year.

Despite this Sculley claimed that Apple ended the quarter in a strong financial position with $187.5 million (£132.04 million) in cash and cash equivalents, shareholders' equity of $386.2 million (£273.9 million) and total assets of $607.5 million (£430.85 million).
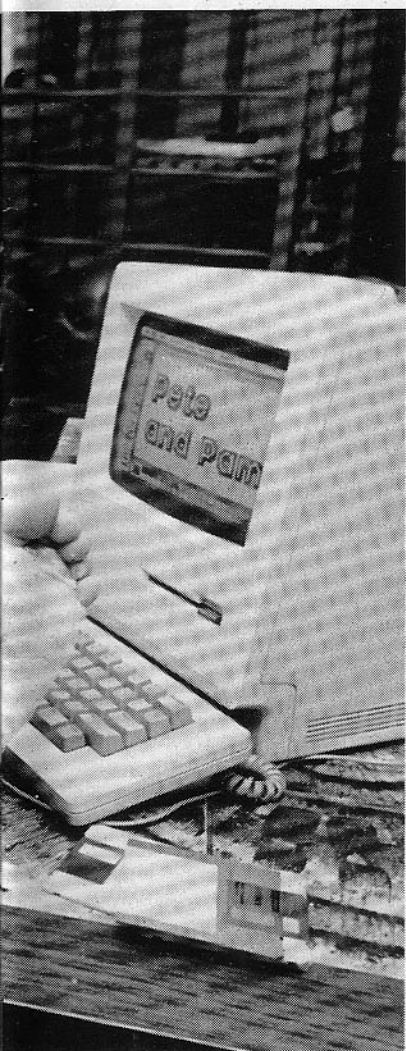
Robots have landed

THE androbots have landed, and the first are expected to arrive in UK dealer's shops this month.

Prism Consumer Products has announced the launch of two of what they call "the world's first generation of personal robots".

At present, they will only work under control of an Apple II, but software for BBC Micro, Spectrum and Commodore 64 is on the way.

The larger robot – a kind of electronic garden gnome called Topo – takes its orders from an Apple II keyboard or joystick via a wireless infra-red link. Its release date is this month.

According to Prism: "Once you've acquainted Topo with his new home, a simple computer command or joystick movement will start him off and running.

The other androbot is billed as a "creative" artist called Fred – Friendly Robotic Educational Device. It is due in he shops in March.

Fred has a 45 word vocabulary, and Prism envisages a big future for him.

Graham Daubney, Prism's director of development, points out that home robots are still in their infancy. But in a year or so, when more advanced software and hardware is available, they should be capable of doing many simple household jobs.
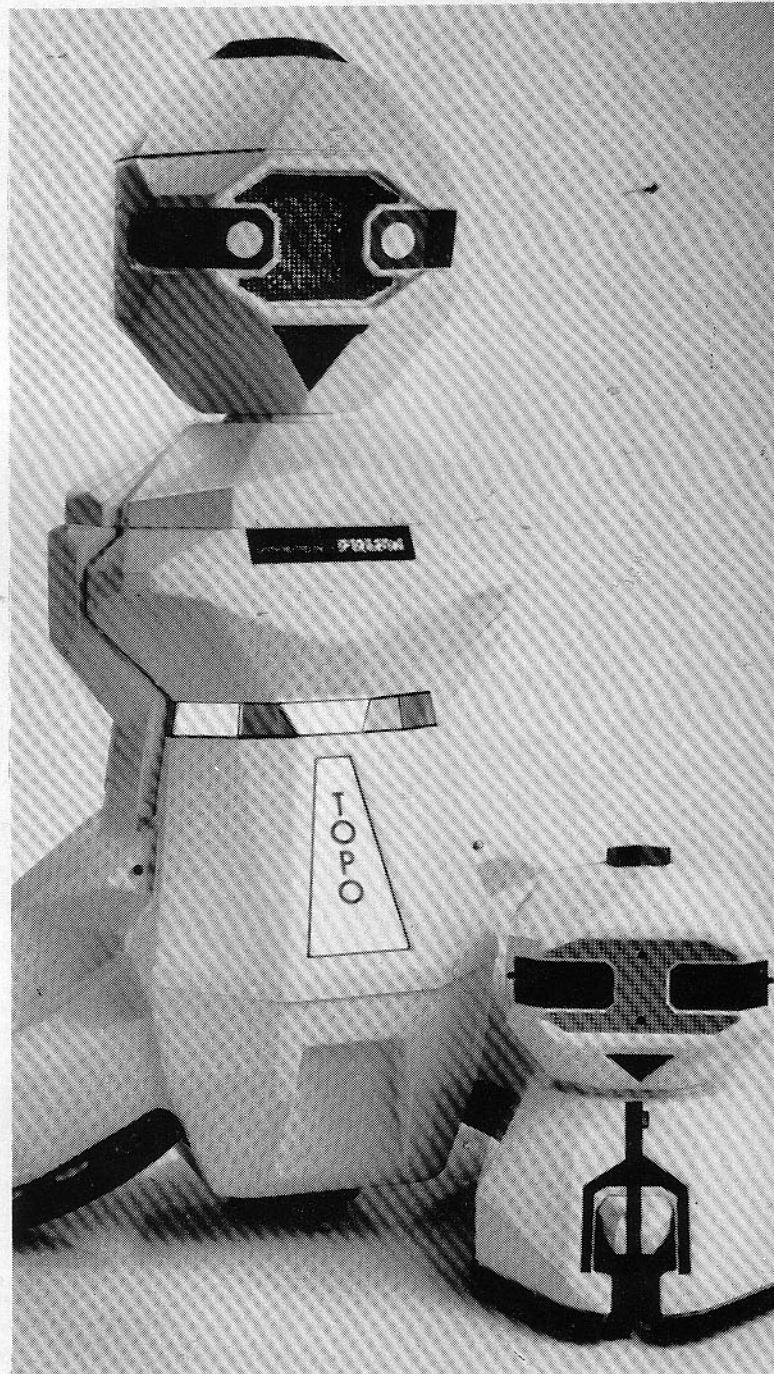
Two for charity

ON the receiving end of a couple of Apple IIes recently were comedian Ronnie Corbett and keep-fit specialist Diana (Green Goddess) Moran.

They accepted two Apple systems, donated by Apple UK at a special luncheon held in London, on behalf of two charities.

Youth Opportunities in Lewisham plan to use their new machine as a training tool at their centre.

The British Institute for Brain Injured Children, based in Bridgewater, Somerset, will use their Apple to ascertain the breathing rates of children with respiratory problems.

The organisers of a third charity, Sequal, also received a system.

Amaze your friends with 20 Questions

HERBIE BRENNAN confesses he cheats at games. And shows how you can, too.

AS a boy I was a sucker for those novelty advertisements that always seemed to start with the words "Amaze your friends . . ."

If you, too, like to amaze your friends, I would recommend this little program. It is as phony as a conjurer's hat, but the effect is so good it will fool even experts for hours.

The background (for the benefit of younger readers) is this:

A long time ago, on radio and later on television, there was an incredibly popular parlour game called "Twenty Questions".

The idea was that a series of objects were chosen and the team had to guess what they were.

They were allowed only 20 questions to do so.

Strictly speaking, their questions had to be framed to allow a simple "Yes" or "No" answer – although the rules were often bent a little.

The problem you now face is to program that parlour game into your computer. And do it in such a way that the end user can play it indefinitely without the mystery objects repeating.

You might consider an enormous database, if you have room for it. But however enormous, it's still limited eventually.

Or, then again, you might just decide to cheat. If the end user *thinks* he's playing Twenty Questions and *enjoys* playing Twenty Questions, where's the harm?

The result was an Applesoft version, written in an hour and typeable into your own Apple in less than 15 minutes.

It has no database, no need for disc drives, takes up miniscule memory space and the only artificial intelligence it uses is artificial cunning.

Program Notes

Line 10 sets the scorekeeping variables. SC is the user's score. CS is the computer's score.

Line 20 is just a title and a pause.

Lines 30 to 90 are user instructions. They're the biggest single chunk of code and rightly so since they set the user up very nicely for your little con game.

Line 100 sets another variable – Q. This lets the computer keep track of the number of questions the user has asked.

Line 120 is an outright lie. The computer doesn't have a mystery object ready. The computer *never* has a mystery object ready. The computer only *seems* to have a mystery object ready.

Line 130 sets up the user input display.

Line 140 pays tribute to the commonplace programmer conviction that end users are sloppy people.

They type in a question and sometimes they put in a question mark, sometimes they don't.

Since this sort of sloppiness can screw up your cunning little program completely, Line 140 scans for a question mark and if it's there gets rid of it, via the short subroutine at Line 280.

That way, it doesn't matter if the end user is sloppy or not.

Line 150 reveals the secret. It doesn't matter a hoot what questions the end user asks. If his input string ends with an "E", an "L" or an "S", he's going to get a "YES" answer to his question. (See Line 190).

Line 160 helps the computer keep accurate score if the user is honest. Dishonest users can, of course, cheat like mad, but this will keep them from examining your dishonest program too closely.

Line 170 prints a "NO" answer to every question that isn't picked up by Line 150, and (like Line 190) makes sure the question sequence stops when the 20 questions have been asked.

Lines 180 and 200 keep the question process rolling.

```
10 SC = 0:CS = 0
20 HOME : VTAB 12: HTAB 12:
   PRINT "TWENTY QUESTIONS":
   HTAB 12: PRINT "======
   =========": FOR J = 1 TO
   2000: NEXT
30 HOME : PRINT "IN THIS
   SPECIAL VERSION OF THE
   POPULAR": PRINT : PRINT
   "TV PARLOUR GAME, THE
   COMPUTER WILL": PRINT :
   PRINT "SELECT A MYSTERY
   OBJECT AND YOU HAVE TO ":
   PRINT
35 PRINT "GUESS WHAT IT IS.
   YOU ARE ALLOWED TO":
   PRINT : PRINT "ASK UP TO
   20 QUESTIONS TO DO SO. IF
   YOU"
40 PRINT : PRINT "GUESS
   RIGHT THE COMPUTER WILL
   TELL YOU": PRINT : PRINT
   "AND ADD TO YOUR SCORE.":
   PRINT : PRINT
50 PRINT "NOTE THAT THE
   RULES OF THE GAME WILL":
   PRINT : PRINT "ONLY ALLOW
   THE COMPUTER TO GIVE
   YES/NO": PRINT : PRINT
   "ANSWERS TO YOUR
   QUESTIONS.": PRINT :
   PRINT
60 INPUT "PRESS <RETURN>
   WHEN READY ";A$
70 HOME : INVERSE : PRINT
   "IMPORTANT:": NORMAL :
```

```
PRINT : PRINT "IF YOU
CORRECTLY GUESS THE
MYSTERY": PRINT : PRINT
"OBJECT IN LESS THAN
TWENTY QUESTIONS,": PRINT
75 PRINT "YOU SHOULD
TERMINATE THE QUESTION":
PRINT : PRINT "SEQUENCE
BY ENTERING AN ASTERISK
(*) AS"
80 PRINT : PRINT "YOUR ONLY
RESPONSE TO THE NEXT
QUESTION": PRINT : PRINT
"THIS ENSURES YOUR SCORE
WILL BE CORRECT:
90 PRINT : PRINT : PRINT :
INPUT "PRESS <RETURN>
WHEN READY";A$
100 Q = 1
110 HOME : PRINT "TYPE IN
YOUR QUESTIONS, THEN
<RETURN>"
120 PRINT : PRINT "COMPUTER
HAS THE MYSTERY OBJECT
READY.": PRINT : PRINT :
PRINT
130 PRINT : PRINT "QUESTION
";Q;: PRINT ": ";: INPUT
Q$: IF Q$ = "" THEN 130
140 IF RIGHT$ (Q$,1) = "?"
THEN GOSUB 280
150 IF RIGHT$ (Q$,1) = "E"
OR RIGHT$ (Q$,1) = "L"
OR RIGHT$ (Q$,1) = "S"
THEN 190
```

```
160 IF Q$ = "*" THEN Q$ =
"EEE": GOTO 210
170 PRINT : PRINT "NO!";
PRINT :Q = Q + 1: IF Q >
20 THEN 210
180 GOTO 130
190 PRINT : PRINT "YES!";
PRINT :Q = Q + 1: IF Q >
20 THEN 210
200 GOTO 130
210 FOR J = 1 TO 1500: NEXT
: HOME : VTAB 12: PRINT
"SCORES:-"
220 IF RIGHT$ (Q$,1) = "E"
OR RIGHT$ (Q$,1) = "L"
OR RIGHT$ (Q$,1) = "S"
THEN SC = SC + 1: GOTO
240
230 CS = CS + 1
240 PRINT : PRINT "YOU:
";SC: PRINT : PRINT
"COMPUTER: ";CS
250 VTAB 21: INPUT "ANOTHER
GO (Y/N)? ";A$: IF A$ <
> "Y" AND A$ < > "N"
THEN 250
260 IF A$ = "Y" THEN Q = 1:
HOME : GOTO 120
270 HOME : VTAB 12: PRINT
"THANK YOU FOR A
MAGNIFICENT GAME.": FOR J
= 1 TO 1500: NEXT : HOME
: END
280 X = LEN (Q$):Q$ = MID$
(Q$,1,X - 1): RETURN
```

Family favourite

There's naught like a good old game of Noughts and Crosses — especially played on an Apple.

NOUGHTS and Crosses is a long-standing family favourite. The game can be played on cigarette packets, bus tickets, edges of newspapers, beach sand — and on the Apple.

In this particular version the program has been made intentionally beatable, because, says its author, FRANK LEWIS, "it gets boring if the computer always wins!"

But he adds: "It has been tested to destruction by the junior games experts of the household and it seems to stand most tests".

If as a child you developed a winning strategy at the game and if you find your formula still works, try reading the program listing instead to see how it does what it does.

If you are still a child (in age, heart or mind) you will find plenty here to challenge you.

```
NOUGHTS & CROSSES
-----------------
THE GAME IS PLAYED ON THE USUAL GRID,
AND THE SQUARES ARE NUMBERED AS UNDER.
ENTER THE NUMBER OF THE SQUARE YOU WANT
TO PLACE YOUR NOUGHT IN (THE APPLE
ALWAYS PLAYS CROSSES), AND PRESS RETURN

        X   X
  1 X 2 X 3
        X   X
  XXXXXXXXXX
        X   X
  4 X 5 X 6
        X   X
  XXXXXXXXXX
        X   X
  7 X 8 X 9
        X   X
```

```
PRESS ANY KEY TO START
DO YOU WANT THE FIRST
MOVE (Y/N) ?
```
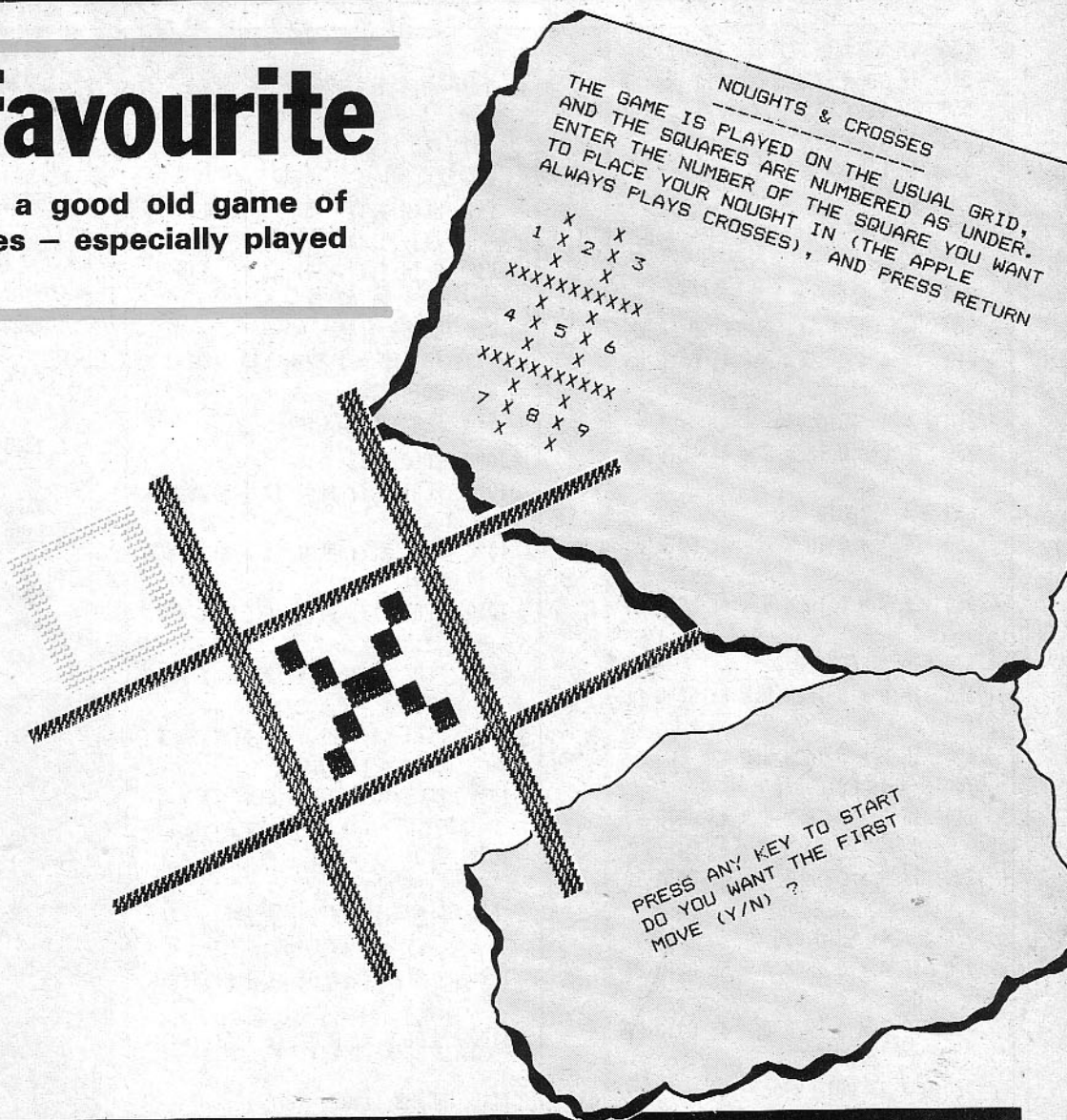
Noughts and Crosses listing

```
100  REM NOUGHTS AND CROSSES GAME
110  GOTO 1300: REM  SETUP PROCS
120  REM COMPUTER'S FIRST MOVE
130  FMC = 0: REM  1ST MOVE COUNT
140  IF E = 0 THEN E = 4: GOTO 87
     0
150  MC = INT ( RND (1) * 9) + 1
160  ON MC GOTO 170,200,180,190,1
     70,180,190,170,200
170  A = 4: GOTO 830
180  C = 4: GOTO 850
190  G = 4: GOTO 890
200  J = 4: GOTO 910
210  VTAB 22: CALL - 958: PRINT
     "WHAT'S YOUR MOVE ? ";: GET
     MOVE$
220  PRINT "";MOVE = VAL (MOVE$)

230  REM PLAYER'S MOVES
240  IF MOVE = 99 THEN 1300
250  IF MOVE < 1 OR MOVE > 9 THEN

     210
260  ON MOVE GOTO 280,300,320,340
     ,360,380,400,420,440
270  REM NOUGHTS SETUP ROUTINE
280  IF A = 0 THEN A = 5:X = 8:Y =
     2: GOTO 920
290  GOTO 450
300  IF B = 0 THEN B = 5:X = 17:Y
     = 2: GOTO 920
310  GOTO 450
320  IF C = 0 THEN C = 5:X = 26:Y
     = 2: GOTO 920
330  GOTO 450
340  IF D = 0 THEN D = 5:X = 8:Y =
     15: GOTO 920
350  GOTO 450
360  IF E = 0 THEN E = 5:X = 17:Y
     = 15: GOTO 920
370  GOTO 450
380  IF F = 0 THEN F = 5:X = 26:Y
     = 15: GOTO 920

390  GOTO 450
400  IF G = 0 THEN G = 5:X = 8:Y =
     28: GOTO 920
410  GOTO 450
420  IF H = 0 THEN H = 5:X = 17:Y
     = 28: GOTO 920
430  GOTO 450
440  IF J = 0 THEN J = 5:X = 26:Y
     = 28: GOTO 920
450  VTAB 23: CALL - 958: PRINT
     "YOU CAN'T GO THERE !": FOR
     I = 1 TO 750: NEXT : GOTO 21
     0
460  REM LINECHECK FOR X'S
470  IF E = 0 THEN E = 4: GOTO 87
     0
480  LINE = 8
490  IF DA = LINE THEN 610
500  IF DB = LINE THEN 630
510  IF HA = LINE THEN 650
520  IF HB = LINE THEN 680
```

```
530  IF HC = LINE THEN 710
540  IF VA = LINE THEN 740
550  IF VB = LINE THEN 770
560  IF VC = LINE THEN 790
570  IF LINE = 8 THEN LINE = 10: GOTO
     490
580  IF LINE = 10 THEN LINE = 5: GOTO
     490
590  GOTO 1590
600  REM CHECK IF SQUARE IS FREE
610  IF A = 0 THEN A = 4: GOTO 83
     0
620  J = 4: GOTO 910
630  IF C = 0 THEN C = 4: GOTO 85
     0
640  G = 4: GOTO 890
650  IF A = 0 THEN A = 4: GOTO 83
     0
660  IF B = 0 THEN B = 4: GOTO 84
     0
670  C = 4: GOTO 850
680  IF D = 0 THEN D = 4: GOTO 86
     0
690  IF F = 5 THEN 530
700  F = 4: GOTO 880
710  IF G = 0 THEN G = 4: GOTO 89
     0
720  IF H = 0 THEN H = 4: GOTO 90
     0
730  J = 4: GOTO 910
740  IF A = 0 THEN A = 4: GOTO 83
     0
750  IF D = 0 THEN D = 4: GOTO 86
     0
760  G = 4: GOTO 890
770  IF B = 0 THEN B = 4: GOTO 84
     0
780  H = 4: GOTO 900
790  IF C = 0 THEN C = 4: GOTO 85
     0
800  IF F = 0 THEN F = 4: GOTO 88
     0
810  J = 4: GOTO 910
820  REM CROSSES SETUP ROUTINE
830  X = 8:Y = 2: GOTO 1000: REM
     A
840  X = 17:Y = 2: GOTO 1000: REM
     B
850  X = 26:Y = 2: GOTO 1000: REM
     C
860  X = 8:Y = 15: GOTO 1000: REM
     D
870  X = 17:Y = 15: GOTO 1000: REM
     E
880  X = 26:Y = 15: GOTO 1000: REM
     F
890  X = 8:Y = 28: GOTO 1000: REM
     G
900  X = 17:Y = 28: GOTO 1000: REM
     H
910  X = 26:Y = 28: GOTO 1000: REM
     J
920  REM   NOUGHTS
930  COLOR= 3
940  HLIN X,(X + 5) AT Y: HLIN X,
     (X + 5) AT (Y + 9)
950  VLIN Y,(Y + 9) AT X: VLIN Y,
     (Y + 9) AT (X + 5)
960  GOSUB 1140
970  IF FMC = 1 THEN 130
980  GOTO 470
990  REM  CROSSES
1000 COLOR= 12
1010 PLOT X,Y: PLOT (X + 5),Y:Y =
     Y + 1
1020 PLOT X,Y: PLOT (X + 5),Y:Y =
     Y + 1
1030 PLOT (X + 1),Y: PLOT (X + 4
     ),Y:Y = Y + 1
1040 PLOT (X + 1),Y: PLOT (X + 4
     ),Y:Y = Y + 1
1050 PLOT (X + 2),Y: PLOT (X + 3
     ),Y:Y = Y + 1
1060 PLOT (X + 2),Y: PLOT (X + 3
     ),Y:Y = Y + 1
1070 PLOT (X + 1),Y: PLOT (X + 4
     ),Y:Y = Y + 1
1080 PLOT (X + 1),Y: PLOT (X + 4
     ),Y:Y = Y + 1
1090 PLOT X,Y: PLOT (X + 5),Y:Y =
     Y + 1
1100 PLOT X,Y: PLOT (X + 5),Y:Y =
     Y + 1
1110 GOSUB 1140
1120 PRINT : PRINT : PRINT : GOTO
     210
1130 REM RESET LINE VALUES
1140 DA = A + E + J:DB = C + E +
     G: REM DIAG'LS
1150 HA = A + B + C:HB = D + E +
     F:HC = G + H + J: REM HORIZ
1160 VA = A + D + G:VB = B + E +
     H:VC = C + F + J: REM VERT
1170 REM CHECK IF ANYONE'S WON
1180 WIN = 12
1190 IF DA = WIN GOTO 1570
1200 IF DB = WIN GOTO 1570
1210 IF HA = WIN GOTO 1570
1220 IF HB = WIN GOTO 1570
1230 IF HC = WIN GOTO 1570
1240 IF VA = WIN GOTO 1570
1250 IF VB = WIN GOTO 1570
1260 IF VC = WIN GOTO 1570
1270 IF WIN = 15 THEN 1290
1280 WIN = 15: GOTO 1190
1290 RETURN
1300 REM DRAW THE GRID AND GET
     STARTING ORDERS
1310 TEXT : HOME : VTAB 3
1320 PRINT  TAB( 12)"NOUGHTS & C
     ROSSES"
1330 PRINT  TAB( 12)"------------
     -------"
1340 VTAB (6): PRINT "THE GAME I
     S PLAYED ON THE USUAL GRID,"
     : PRINT "AND THE SQUARES ARE
     NUMBERED AS UNDER."
1350 PRINT "ENTER THE NUMBER OF
     THE SQUARE YOU WANT": PRINT
     "TO PLACE YOUR NOUGHT IN (TH
     E APPLE"
1360 PRINT "ALWAYS PLAYS CROSSES
     ), AND PRESS RETURN"
1370 POKE 32,15: PRINT
1380 PRINT " X   X  ": PRINT
     " 1 X 2 X 3 "
1390 PRINT " X   X  ": PRINT
     "XXXXXXXXXX"
1400 PRINT " X   X  ": PRINT
     " 4 X 5 X 6 "
1410 PRINT " X   X  ": PRINT
     "XXXXXXXXXX"
1420 PRINT " X   X  ": PRINT
     " 7 X 8 X 9 "
1430 PRINT " X   X  "
1440 POKE 32,0
1450 VTAB 23: PRINT "PRESS ANY K
     EY TO START ";: GET KEY$
1460 CLEAR : HOME : GR
1470 COLOR= 6
1480 HLIN 6,35 AT 13: HLIN 6,35 AT
     26
1490 VLIN 1,38 AT 15: VLIN 1,38 AT
     24
1500 FMC = 1
1510 VTAB 22: CALL - 958: PRINT
     "DO YOU WANT THE FIRST MOVE
     (Y/N) ? ";: GET ANS$
1520 PRINT ""
1530 IF ANS$ = "Y" THEN 210
1540 IF ANS$ = "N" THEN 130
1550 IF ANS$ = "Q" THEN 1620
1560 GOTO 1510
1570 IF WIN = 12 THEN  FOR I = 1
     TO 4: PRINT "": NEXT : PRINT
     "I WIN !": GOTO 1600
1580 FOR I = 1 TO 4: PRINT "": NEXT
     : PRINT "YOU WIN !!": GOTO 1
     600
1590 PRINT "STALEMATE !"
1600 PRINT : PRINT : PRINT "ANOT
     HER GAME (Y/N) ? ";: GET ANS
     $
1610 IF ANS$ = "Y" THEN 1460
1620 TEXT : HOME : VTAB 12: PRINT
     TAB( 18)"BYE": END
1630 REM WRITTEN BY FRANK LEWIS,
     SEVENOAKS
```
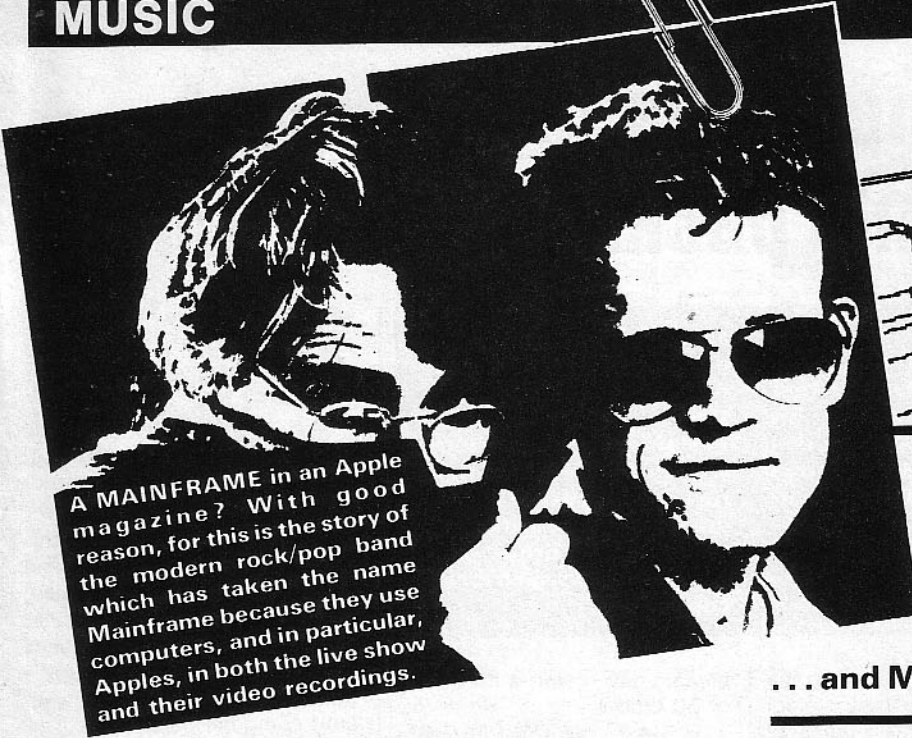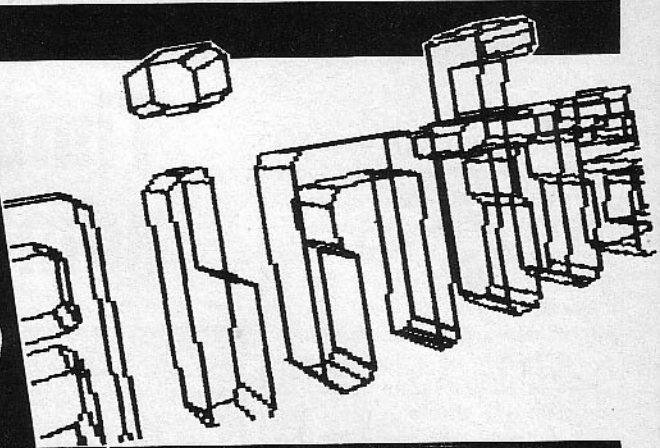
APPLE JOINS A POP GROUP

... and MIKE BADHAM watches the result

MAINFRAME is a two-person synthesiser/guitar group from Hemel Hempstead, not more than a stone's throw from the UK headquarters of Apple itself.

It started in June 1982, when musicians Murray Munro and John Molloy met in another band and decided that hi-tech music and computers were the thing to pursue.

On stage with the band are the group's trio of Apple computers, whose job is to create computer graphics which synchronise with the music.

These are projected onto large on-stage TV monitors to give the fans a visual dimension to the sounds they hear.

John Molloy is the leading light when it comes to the computing side. He is assisted by a couple of software enthusiasts; David Green and Colin Holgate, who is also the behind-the-scenes Apple operator at all live shows.

By October 1982 Mainframe's first single, "Radio" had been recorded and pressed with the "C" side — an Apple program cut in the grooves — believed to be the first pop single with a computer program.

Various problems prevented the release of "Radio" until February last year when the media gave it a good reception and the program was broadcast on Radio One's "Newsbeat" and the London LBC station's "Jellybone".

The current single has moved on apace and has some interesting sound-to-vision programs on the "B" side, created on the Apple and transported to ZX 81, Spectrum and the BBC Micro.

The software used to produce the graphics is engagingly diverse. Applesoft is the basis for much of the work, but two programs written by Paul Lutus, Appleword (a 3D design

Mainframe uses a wide variety of graphics images

package) and Graforth (a fast integer version of Forth) provide fine vehicles for the visual effects.

A real-time abstract graphics package called Ceemac by Brooke Boering completes the software for a sound-to-vision library of effects.

To interface with these packages and the applications programs designed with these tools, Colin has written a number of routines such as the creation of an Appleword file by means of joystick screen drawing.

The saving of program entry time and the easy creation of complex wireframe bodies such as their own logo has helped in the production of a wide selection of visuals to accompany the show.

David, having seen the need for speedy graphics, has created a special 3D package which uses Graforth to create an image.

The objective with these images such as the Mainframe logo is to create the "flying round the city" effect. Several ways have been tried to achieve this including video frame-by-frame animation. This produces the desired result but is very time-consuming.

Another package used is Ceemac, which is an abstract graphics language deserving some attention from serious graphics enthusiasts.

An important idea was the development of good moving displays on the video screen from music signals input via the cassette ports, the display to preferably have a frequency-dependent component in addition to the usual level criterion.

Music coupled with software is a growing field of interest, if for no other reason than the rapidly expanding ownership of personal computers by people who also have a great interest and knowledge of the world of pop.

The rapid move to synthesisers in all areas of the pop/rock scene will bring with it possibilities of computer control — all the more likely since the manufacturers of digital synthesisers have sensibly decided on a standard interface, the Midi, which will allow the networking of synths themselves and the peripherals of the rock band, drum machines, effects pedals and echo/delays.

It is on this note that Mainframe promise something very big for musical Apple-folk in the not-too-distant-future.

Left, Colin Holgate; centre, Murray Munro; right, John Molloy

Timetables test student admin package to the limit

COMPUTERS were used for school administration long before micros were generally available in schools.

Even the most traditional of teachers soon realised that a device which could deal with a great deal of structured data which had to be processed and cross-referenced was bound to save time in the long run.

Admittedly the run has been longer than some of us thought.

What few schools did have was an on-line, interactive administrative system. But since the coming of the micro many people have attempted to write one for the school computer with varying degrees of success.

Studentdata by Brian Healy of Educational Software is a commercial example of such a system. It represents the most ambitious attempt I have seen at a complete administration package.

It can cope with both student and teacher data, including the expected timetables, class/subject lists, exam results, teacher availability (for class cover in staff absence).

Included is a variety of school listings in alphabetic order, date of birth order, etc., address labels, fees paid, subject report forms, exam listings and so on.

With the exception of pupil registers, I've not been able to think of a useful facility not covered, bearing in mind that Studentdata is offered as an administration package and not a timetabling aid such as Rostar, or as an option selection utility.

The package, which costs £166, consists of 42 programs on six discs which, to speed up operation have been compiled from Applesoft.

Data on students is stored one year to a disc and there is a separate disc for the teacher data and the index.

Allowing for the usual back-up copies and necessary archive copies for pupils who have recently left the school, this means that at some schools up to 32 data discs could be required.

Fortunately once a pupil's details have been entered on the system little extra typing is required as the pupil progresses through the school levels.

At the end of each year the

Fundamental to the business of schools, which is teaching, is that of administration. An Apple package which will handle most school administration other than registration is Studentdata.

PHILIP BOLT, a Scottish teacher, describes it as a good attempt at a global administration package, although rather slow.

He found minor bugs and considers its usefulness limited by the maximum number of records and other parameters with which it can cope.

student data on disc can be re-indexed for the new school session so that basic personal information does not have to be re-entered each year.

Headmasters or administrators can decide who should have access to all this information as entry to the package is password controlled at three levels.

A master password gives unlimited access, teacher passwords give access to all student data and to the teacher's own data and student passwords allow pupils access only to their own data.

One minor fault is that the password is displayed on screen as it is typed instead of being hidden from prying eyes.

The programs are menu driven with at least two levels of menu.

As the various program discs are coded in relation to the menu options selection of the required disc is easy but, in any case, the disc is checked and prompts are given until the correct one is used.

I tested the package with a small sample of data based on the organisation of a Scottish school.

This point must be born in mind because ultimately its usefulness does depend on the individual school organisation.

Comments made from my experience may not apply to schools with different systems which may find Studentdata more or less suitable to their requirements.

The timetable structure is based on a system of blocking the week — up to 10 days of 15

periods/day — with a maximum of 50 blocks.

However as this structure has to apply across the whole school it is likely that a block would be used for a single period unless you operated in whole school half-day blocks or some similar system.

Up to 80 teachers and 48 subjects can be accommodated on a Studentdata timetable.

The number of pupils it handles depends on the storage capacity of your discs being 350 in any one year with a maximum of 700 for the whole school using DOS 3.3 discs.

The system can be tailored to individual school use to a certain degree by using up to five user-defined flags to record any desired parameter for both pupils and staff.

The instruction manual is well written and easy to use, detailing even the use of COPYA to make the back-up copies.

However, this is a large package and it would most probably require one member of the administrative staff to specialise in its use and be responsible for daily updating.

The main problems found in using the system were concerned with the maximum sizes of the various parameters.

While any teaching group is uniquely defined by subject/room/teacher, it is far more suitable to be able to designate teaching groups by a code which reflects the year of study and subject section, for example 3E1 for third year English, section 1.

It was this factor which proved to be the main problem

with the package which could not accommodate the number of teaching sections required in its original form.

It is also desirable to be able to designate different levels of study, particularly for senior years.

In our particular school this would require over 60 subject codes, even without indicating the sections, and this is not possible with the present issue of Studentdata although I understand that the 48 subject limit could be increased.

There is another limit in the internal grading system. It allows only up to nine subjects to be included. At S1/2 level we would require 15.

The limit of 80 teachers might also create problems in the larger schools, as will the maximum number of pupils to a year.

The system is slow in operation due to the frequent changes of disc and the fact that exiting from an option returns you to the main menu, not to the most recent.

While I did not attempt it, the start of session update of the student data, according to the manual, takes a whole day and this does not include the time that would be needed to enter in any new timetable information.

The amount of student data seems to be adequate for most schools' needs, but there may be a need for more teacher information, for example previous school, qualifications, years teaching experience, level of responsibility, etc.

Some of this could be entered by using user-defined flags, but five more items might still be insufficient.

There are one or two operational points I didn't like.

Some responses are single key, some require Return to be pressed. I've come to believe that the only single key response should be a Return.

The menu selection routine prints out the number of the selected option and asks for Y/N confirmation.

I prefer a system which highlights the chosen option and uses a Return for confirmation.

Studentdata represents a noble attempt at a global school administration package. Its utility will depend on the curricular structure and size of the school. If it fits it will be a help, but I fear that for many schools it will not fit in its present form.

The sample timetable *(right)* shows where Bolt B. is at any time and what he is doing, but it would be better if it indicated the section and level of study.

Two final points need to be made which do not just apply to Studentdata but to the very concept of running a school's administration on a small computer.

I am doubtful if this can ever be done efficiently on an 8 bit machine. I believe that the problems of speed and maximum parameter values can be related to 8 bit operation and

it is worth mentioning that the Scottish Scamp project is only implemented for 16 bit machines. (Anyone for Lisa?).

However, I would still have reservations even if a 16 bit machine is used. At the start of each session new timetables have to be entered for each pupil and member of staff.

With a manual system this is dealt with by having each person complete their own copies which are then centrally collected, collated and stored.

With any computer system all of this information has to be entered centrally, possibly by one person.

It seems to me that no package of this type will be completely successful using keyboard entry of data.

What is needed is a cheap, British (that is, quickly serviced), mark-sense reader to allow timetable data, test results, etc., to be encoded on cards by pupils and staff and then read into the system en masse.

```
TIMETABLE FOR:- BOLT               B
TUTOR GROUP 3/1
          1    2    3    4    5    6    7    8
MON     |MA  |EN  |B4  |B4  |LA  |LA  |GA  |GA  |
        |BB  |FF  |PP  |PP  |TT  |TT  |JJ  |JJ  |
        |G2  |G6  |F6  |F6  |F10 |F10 |G11 |G11 |
TUE     |MS  |MS  |MA  |EN  |P5  |P5  |AR  |AR  |
        |KK  |KK  |BB  |FF  |MM  |MM  |RR  |RR  |
        |G12 |G12 |G2  |G6  |F1  |F1  |    |    |
WED     |EN  |EN  |B4  |B4  |AR  |AR  |MA  |MA  |
        |FF  |FF  |PP  |PP  |RR  |RR  |BB  |BB  |
        |G6  |G6  |F6  |F6  |    |    |G2  |G2  |
THU     |AR  |P5  |MR  |MR  |EN  |MS  |B4  |SE  |
        |RR  |MM  |EE  |EE  |FF  |KK  |PP  |DD  |
        |    |F1  |G5  |G5  |G6  |G12 |F6  |    |
FRI     |MS  |MS  |P5  |P5  |EN  |MR  |MA  |MA  |
        |KK  |KK  |MM  |MM  |FF  |EE  |BB  |BB  |
        |G12 |G12 |F1  |F1  |G6  |G5  |G2  |G2  |
```

A sample timetable prepared with Studentdata

Product: Studentdata
Description: Schools administration program
Price: £166
Distributor: Education Software, 12 Barclay Road, Walthamstow, London E17.

CLIFF McKNIGHT contemplates how copy protection sometimes can be the ruin of your discs . . .

HAVE you ever ruined a disc? I mean **really** ruined it so it couldn't be written to or read again? I haven't, but it happened to a friend of mine recently.

She was new to computing and as far as we could work out when we held the autopsy, she had moved the disc while the "in use" light was on.

Several tracks were unreadable, even using the CIA Files. Fortunately, she had a back-up, so the information wasn't lost.

I tell you all this, not to point out that some of my best friends are Wallies, but because I was reminded of it a few days ago.

I had been playing a game (no, honest) and had left the machine on when I finished.

When I wandered back into the room I noticed that the machine was still on and the game's title screen was being displayed.

Vast electricity bills floated before my eyes, so I reached over to remove the disc before switching off – just as the game accessed it and went into demo mode!

When I recovered from the shock of nearly committing a cardinal sin worse than coveting my neighbour's ox, I poured myself another drink and gave the matter some thought. It went something like this:

Most arcade games will fit into 48k with no difficulty, so they don't **need** to access the disc once they are loaded.

The usual reason for repeated access is to check that the original disc is still there.

Often this check involves counting the number of nibbles at a particular point on the disc.

The counting is a form of copy protection, because the number of nibbles present will depend on the speed of the drive in which the original disc was written.

By writing the original at a non-standard speed it ensures that most copies made on normal speed drives will not contain the correct number of nibbles in the crucial section, and the program will therefore abort itself.

This means that a system designed to copy-protect the disc actually increases the likelihood that it will be physically damaged.

Also, the fact that it is a very successful protection scheme means that many modern games use it.

The way to avoid damage is easy, if a little tedious.

Most games have a pause facility, so you can restart the game and then pause it.

In pause mode none of the games I know access the disc, so the chances are that it can now be removed safely.

I'd like to know how many discs have been damaged inadvertently like this, but it's probably something the manufacturers don't want to talk about.

You'll be all right if your face fits

EVERY year I tell a new group of students not to take it personally if I ignore them in the corridors of the college. By way of explanation, I admit that I have a terrible memory for faces.

However, next year I might not have to make this confession because Facemaker and Police Artist might be useful for improving my facial recognition.

Title: Facemaker
Publisher: Spinnaker Software

Facemaker comes from Spinnaker Software, publishers of the best-selling Snooper Troops cases (Windfall, December 1983). It allows a face to be built up by selecting a particular feature (nose, eyes, etc). and then choosing from the available examples.

Once you've designed a face, you can animate it in certain ways. For example, it can wink, frown, stick its tongue out or
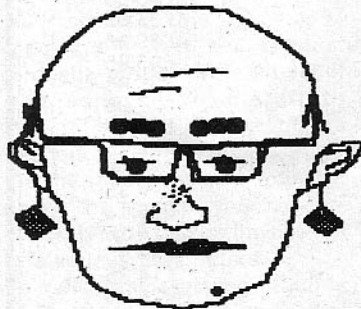
wiggle its ears in addition to smiling and crying.

Single letters are used to specify the action, and a "program" of actions can be written to produce a sequence of actions.

All this is good fun, but there is also a game element which
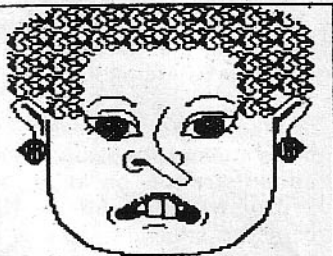
can be selected. This is a sort of Simon game using the face you've designed, with another action being added to the string each time you correctly remember it.

A count is kept of the number

of items you remember, and your highest score is also displayed.

The animations are each accompanied by a particular sound, so Spinnaker say that the program not only improves visual but also auditory sequential memory.

Keyboard familiarity is also learned, and the specifying of a sequence of animations is seen as a "gentle introduction to programming". The appropriate age range is given as four to eight years.

Title: Police Artist
Publisher: Sir-Tech

Police Artist comes from Sir-Tech, renowned for Wizardry among other things. It consists of three separate games all revolving around the identikit idea.

In Police Lineup, the face of a culprit appears briefly on the left of the screen. Then one by one a series of suspects appears on

the right of the screen. You must identify the culprit and reject innocent people.

There are four levels (bystander, eye witness, star witness and eagle-eye) and you can choose which level you start on. Success leads to higher levels, unless of course you start at the hardest.

In Police Artist, you press the space bar for the culprit to appear on the left of the screen. When you think you know the face, pressing space bar causes it to disappear. You must then construct the culprit's face by choosing the appropriate features from a pool.

The number of seconds you peek at the culprit is your score,

and so you are aiming for the lowest score. You can peek again while you're trying to reconstruct the face, but it puts your score up.

Again there are four levels, with only two face parts per group on level 1 but 16 face parts per group on level 4.

Off Duty simply allows you to build faces for fun. You have access to the whole bank of face parts, so there are 1,048,576 possible combinations.

Each face in all three sections of Police Artist is given a name, and there are also various sounds used throughout. The appropriate age range is given as from seven years upwards.

Both of these packages proved extremely popular with the children who tried them. Police Artist is manageable by children younger than seven, and Facemaker proved popular with children older than eight.

One of the favourite games was "let's make a face like daddy/mummy/teacher" and the ability to make the face stick its tongue out was a real hit! Maybe if I get my students to stick their tongues out I might recognise them.

Cliff McKnight

Medical eduventure

Title: *Microbe*
Authors: *Alan Zalta MD and Robert Clardy*
Publisher: *Synergistic Software*
Requirements: *Apple II*

Title: *Adventures in Flesh*
Developer: *Professor Fred Williams*
Publisher: *Krell Software*
Requirements: *Apple II*

PICTURE the scene . . . a crowd has gathered around someone who has collapsed in the street and various onlookers are offering advice. "Loosen his collar", "Give him some brandy", they say.

Suddenly, a tall (well, maybe not) figure with a stethoscope hanging our of his pocket pushes his way through the crowd. "Let me through", he says authoritatively, "I've played Microbe and Adventures in Flesh".

Far fetched, of course, but in fact both these games are medically accurate, although both can be played simply for fun. They represent two quite different approaches to the same subject, that is the human body, but both use the "Fantastic Voyage" type of scenario.

Microbe is described as an "eduventure" which combines adventure, arcade, strategy and simulation qualities. It comes on a double-sided disc, one side of which is the game and the other side is a demonstration program.

It is essential to watch the demo at least once in order to understand what is required.

Four manuals are provided, each corresponding to one of the crew functions — captain, navigator, technician and physician. You'll have to read these, too, especially if more than one person is playing.

With a single player, the computer provides most of the information, but with several players much more use must be made of the manuals.

If you specify your background as "gamer", the sub's physician will provide you with diagnostic and treatment infor-

mation. However, "student" and "physician" players are provided with less information. Three skill levels are available.

You can also select the type of cases to work on. The default option of "health/safety" provides patients who are children or young adults injured in the home or in a driving accident.

"Medicine" cases are generally older people with either medical emergencies, like heart attacks, or injuries sustained during various attacks.

It's impossible to do justice to a game like Microbe in a page or two of review. Firstly, the range and variety of graphics means we could devote a page to screen dumps and still not show everything.

Secondly, the complexity of the game parallels the complexity of its subject matter. Thirdly, the difficulty means that a lowly reviewer like myself could spend six months getting far enough "into" the game to achieve a reasonable score.

The fact that the game is medically accurate means that it could be used as a teaching aid. However, it would be a fairly

progressive medical school which encouraged its students to spend the necessary amount of time playing the game.

There's so much competition to get into the profession that colleges don't need to worry about motivating their students.

Presumably then, Microbe is intended primarily as a game — after all, the games market is much bigger than the medical education market. As such, it should appeal to anyone who likes complexity and difficulty.

When I first saw the title Adventures in Flesh I thought "say no more, nudge nudge, wink wink". However, it's not at all like that. Like Microbe, you are travelling through the body in a miniaturised submarine. But Adventures in Flesh is more like a traditional adventure game.

For a start, it's text-only. It comes supplied with "lavish documentation" — a Human Anatomy Colouring Book from Dover Publications!

After the initial surprise of being provided with a colouring book (but no crayons, I might add), I must admit I found it a very useful reference book. Of
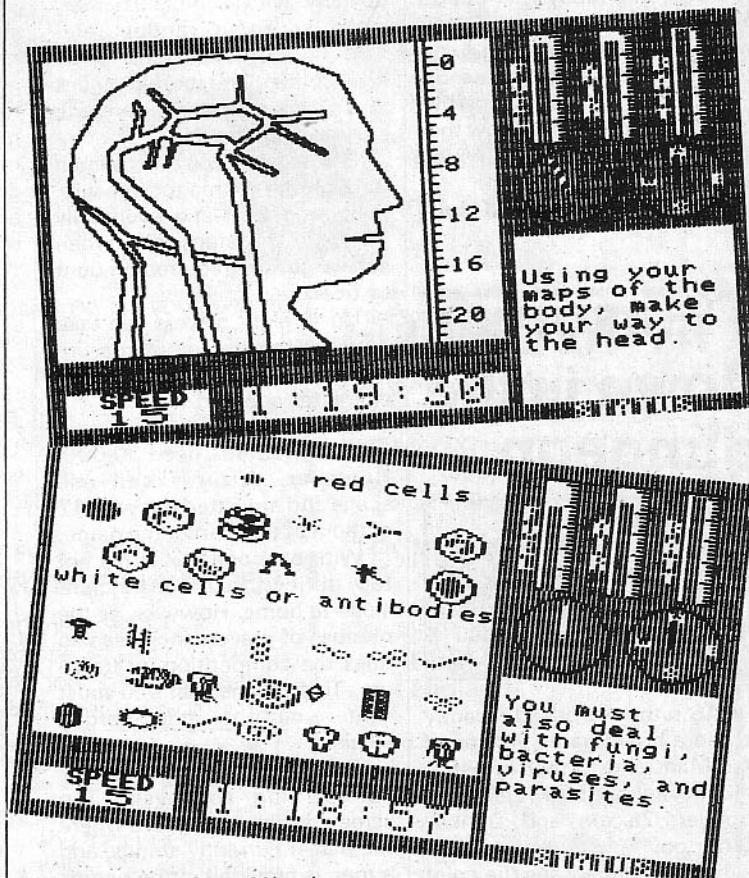
Screendumps from Microbe

course, if your knowledge of anatomy is up to scratch you might not need it.

The adventure format means that you can Look Cell, Go Artery and generally behave as an adventurer. However, there is an important additional command: Diagnose.

Rather than have to race to the brain to remove a tumour, you must explore the whole system in search of symptoms. You must use these clues to diagnose the eleven ailments with which the patient is suffering, these having been chosen at random from a larger possible set.

A random element also provides hazards as you move around. This means that even if you know where you are going, there might be a surprise or two in store. Although random, the hazards are biologically valid for the location in which they appear.

I found Adventures in Flesh far more "accessible" than Microbe and consequently learned more from it in the same amount of time.

If you're prepared to work at it, Microbe will certainly keep you busy and repay your efforts eventually. However, if you want some fun now then I can recommend Adventures in Flesh. There are no graphics, but it's the only program I know that uses the disc drive for sound effects!

Cliff McKnight

Treasures down in the dungeon

Title: Dungeon!
Authors: Bruce Nesmith, Keith Enge and Stephen D. Sullivan
Publisher: TSR
Requirements: Apple II

I WAS rather surprised recently to see a board game version of Pac-Man, and further investigation revealed board versions of Frogger, Zaxxon and Donkey Kong too.

I couldn't really see the point
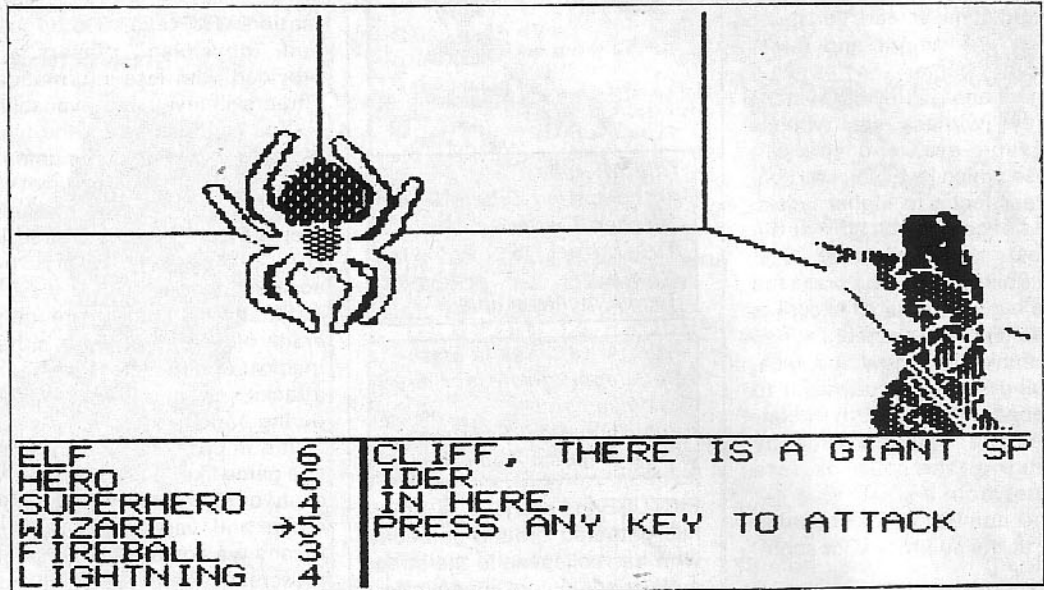
Dungeon chambers harbour many perils

of converting computer games to board games, unless real addicts couldn't live through a power failure.

Dungeon! represents the more usual direction of movement, from board to computer. There are several arguments in favour of automating play — score-keeping is easier, particularly with complex games, more sensitive random elements than multiple dice can be incorporated and it often means that a single player can compete against the machine.

In Dungeon! you must search through the rooms for treasure, fight monsters, and eventually return to the start point when you've got the required amount of treasure.

Up to eight players can take part, and the amount of treasure each needs depends on the character adopted.

Elves need 10,000 units of gold but Wizards need 30,000. However, Wizards can use spells and are kitted out with 12 of them at the start of the game.

With only one player it's not too difficult to find treasure close to home. However, as the number of players increases so does the competition for treasure. This means that you must venture out onto other, harder, levels.

There are nine "boards" representing six levels. The screen displays the board where the player currently stands, and a map is provided of the boards

and the way they interconnect. Movement of your player (a coloured square) is achieved using U-D-L-R keys.

In monochrome it is sometimes quite difficult to locate yourself, so you need to remember roughly where you were while the other players take their turns.

When you enter a room or chamber a monster that you must fight is displayed on the hi-res screen. Fighting is by the roll of computer dice, with the number needed to defeat the monster being displayed. If you don't kill it, it may well kill you or at least cause you to drop some of your treasure.

As the illustration shows, the

text is sometimes split in odd ways. Also, some of the monsters are not what you'd expect to find . . . like Black Puddings. No doubt the Lancastrian martial art of Ecky Thump works best on these.

Dungeon! isn't a brilliant game, and it sometimes feels like a board game. However, it is easy enough to play, and is therefore suitable for children.

Indeed, our daughter and her friends spent a delightful rainy afternoon finding treasure and zapping Black Puddings. Judging by the ambient noise level, they had a whale of a time. At £12.95, it's good value for money.

Cliff McKnight
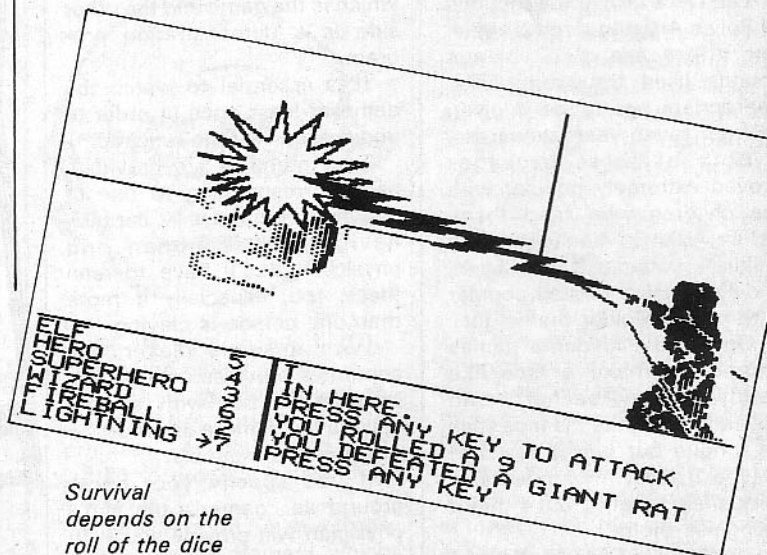
Survival depends on the roll of the dice

Apple Pascal has extensive string handling capabilities. This month GORDON FINDLAY defines a string and looks at declarations, assignments and input-output relating to strings.

Don't get tied up in knots over Pascal strings

THUS far in our exploration of Pascal we have manipulated only numbers, representing numerical data such as hours worked and wages. But numerical data isn't everything — we need to be able to handle names and addresses, questions and answers, and all sorts of non-numerical stuff.

This sort of data is represented in Pascal by strings.

A string is just a sequence of characters, all strung together as it were. Examples are "P.O. BOX 827" and "WHAT IS YOUR NAME?".

Standard Pascal doesn't have facilities for handling strings, but Apple Pascal has extensive string handling built in.

Programmers can easily store, input, output and compare strings, and alter them in many ways.

Strings may be stored by assigning the variables, just like numbers: for example, X: = 'HAPPY BIRTHDAY'. The quote marks are only to delimit the string — that is, to separate it from the rest of the line.

In this statement, X is a variable, and of course it must be declared like any other. STRING is a type, just like REAL or INTEGER, and strings are declared just as numbers are:

```
VAR
    HOURS, RATE: REAL;
    IDENTIFICATION: INTEGER;
    NAME, ADDRESS: STRING;
```

Remember that the order of declarations is not important — any other order would have been perfectly OK.

Strings tend to vary in length, which will almost certainly change during a program. But the compiler needs some sort of information about lengths before the string even has an initial value.

For this reason strings are declared with a maximum length. This will be 80 characters, unless you specify otherwise in the declaration — 80 is the "default" length.

Thus the string's name and address above can be up to 80 characters long each. They may well be shorter than 80, but cannot become longer. The default length corresponds to the width of the screen.

If you want a different length you may specify it by including the maximum in square brackets as part of the declaration, like this:

```
VAR SHORT:STRING[10];
    MEDIUM:STRING;
    LONG:STRING[120];
    EXTRALONG:STRING[255];
```

These declarations result in the creation of strings with various maximum lengths. Short can be no more than 10 characters long, medium may go to 80 characters by default and long may have up to 120.

The absolute maximum is illustrated by extralong, with up to 255 characters. This is as high as you can go in a string declaration. Remember that the square brackets are typed with CTRL-K and shift-M.

Always remember that the declared length is a maximum, and that during the execution of a program the strings may not necessarily reach their maxima. Be sure that your declarations allow sufficient room.

There are two schools of thought within the Pascal community about the degree of help the system should give the programmer.

One says that the system should leave everything possible up to the programmer — this gives maximum control, with of course a maximum amount of work.

The other says that the system should contain built-in

conveniences to save programmers constantly reinventing the wheel.

The first point of view is illustrated by the Apple's refusal to allow backspaces when READing. If you want to allow backspacing, write it yourself is the message. The second viewpoint prevails in the Apple's string-handling facilities.

Since the length of a string varies during the execution of a program ("dynamic" lengths) there needs to be a method for finding the current length.

Here's how Pascal does it:

```
L:= LENGTH(X);
or
WRITELN(LENGTH(X));
```

In the first, L must be a declared variable, either real or integer. The length function returns an integer, which may be assigned to a real if required, but L will most often be integer.

Joining strings together is a common operation. In Basic (shudder!) strings are joined together with +. In Pascal on the other hand they are joined together with CONCAT, which is an abbreviation for concatenation. Try this:

```
PROGRAM JOINER;
    VAR X,Y,Z:STRING;
    BEGIN
        X := 'APPLE ';
        Y := 'USER';
        Z := CONCAT(X,Y);
        WRITELN(Z)
    END.
```

This creates two strings, and joins them together to make a third. You may join as many as you like in the one CONCAT, and the strings may be variables (as in joiner) or given directly in the concatenation statement:

```
X := 'APPLE ';
Y := 'USER';
Z := CONCAT(X,Y,' MAGAZINE',
    ' IS THE BEST');
```

gives you "APPLE USER MAGAZINE IS THE BEST". Remember to include blanks where needed — CONCAT doesn't!

As well as joining strings together, we may need to take them apart. In Pascal, the COPY function is used to extract part of a string.

In using COPY, specify the

string to be used, the starting place, and the number of characters to be taken. Those of you who are familiar with Applesoft will recognise the MID$ (function).

If X is a string, with value 'BOOKKEEPER', and PART is another string, the following gives PART the value 'OOKKEE':

```
PART := COPY(X,2,6);.
```

The first letter to be taken is the second, and six letters are taken in all. Here are some other examples:

```
COPY('COMPUTER',3,2) = 'MP'
COPY('APPLESOFT',6,4) ='SOFT'
COPY('MISSISSIPPI',4,3) = 'SIS'.
```

Be careful to keep the starting position and length reasonable: COPY('THE',2,8) isn't very sensible.

There are three other ways of using strings which will not be so familiar to Applesoft users.

POS finds one string in another. The value of POS(X,Y) is an integer, which is the place at which X occurs in Y. Get the order right – the substring comes first.

INDEX: = POS('ISS','MIS-SISSIPPI') gives INDEX the value two. If the first string cannot be found at all, the result is zero. This enables us to find whether or not one string is in another.

Suppose we have READ a string called FILENAME which is the name of a file we are working on. This is, presumably, a data file, with the suffix '.DATA'.

We can make life easy for the user by testing to see whether or not the suffix has been given, and if not adding it with CONCAT:

```
IF POS('.DATA',FILENAME)=0
THEN CONCAT(FILENAME, '.DATA');
```

That way it doesn't matter

whether the user gives the '.DATA' or not.

INSERT and DELETE are two self-contained procedures. INSERT takes a string, a substring to put into it, and a place to put it. DELETE needs to know the string, the place to start deleting, and how many characters to delete.

INSERT('ARE',X,5) puts the word 'ARE' in the string X, starting at the fifth position (strings have no zero position).

DELETE (X,POS('BASIC',X),5) removes 'BASIC' from the string.

String built-ins can be nested to perform quite complex tasks in one line: INSERT('APPLE',X, POS('PASCAL',X)−1)) would change PASCAL to APPLE PASCAL in any string named X.

Try to decide what would happen if X was the string 'BASIC IS NOT NICE'. Try with a machine to see if your idea is correct.

Here is a sample program to

manipulate a string typed in: can you see what it will do? Try it!

```
PROGRAM RIPPLE;

VAR

    X,Y,Z,W:STRING[30];
    I,L:INTEGER;

BEGIN

    READLN(X);
    L := LENGTH(X);
    FOR I:= 1 TO L DO
    BEGIN
        Z := COPY(X,I+1,L-I);
        Y := COPY(X,1,I);
        W := CONCAT(Z,Y);
        WRITELN(W)
    END
END.
```

Accounting for Lisa's taste

LET us examine a few of the design requirements deemed important in the development of SunAccount.

The principal aim was to produce a set of accounting systems with the highest possible specification exceeding the features available on mainframe systems.

To stand any chance of recovering the investment in the shortest possible time, Cobol was chosen as the language providing the greatest choice of compatibility between operating systems.

It also provided an indexed access method within the language and the capability of several users to share files.

The choice of MicroFocus as supplier was made because of its rigid adherence to the ANSI Cobol standard, its reputation and market penetration.

These factors alone provided two significant benefits.

Within three days of attending a recent Apple software developers conference we were able to port SunAccount onto Lisa and demonstrate to ourselves and thus to Apple that we had a viable solution.

What then are the cornerstones of this approach? There are three basic criteria — flexibility, sophistication and ease of use.

Flexibility has been assured by the choice of Cobol. Data can be transferred from other disc files, computer systems or programs.

For example, budgets prepared with LisaCalc operations can be transferred into the ledger for comparison with actual results.

Similarly, transfer out to user-readable files can be achieved for word processing by LisaWrite.

The specific advantages of Cobol have enabled us to develop the package so that the appearance and format of reports using Lisa's mouse can be altered.

Entry and exit from Sun Account can also be mouse controlled as can page scrolls, cash matching, menu selection options and simultaneous reference windows on data entry.

At the time of development, Cobol was undergoing a popularity resurgence due to the effective application of the language by MicroFocus.

MicroFocus Cobol gives low occupancy, fast compile times and fast performance. Five years ago Cobol compilers took half an hour. They now take three minutes.

The principal features of sophistication are ones of size, the efficient use of screen facilities, layout, user driven set up and inherent depth of analyses.

The single ledger which holds up to 65,000 available accounts can record up to 65,000 transactions — if the computer's disc file size can cope with this volume!

SunAccount can actually utilise more storage memory than is provided by the 5mbyte conventional Lisa disc configuration.

However, when Apple releases expanded configurations, including the eagerly awaited larger hard disc drive, the existing capability of Sun Account will be able to exploit that also.

The layout of financial statements and analyses are user defined.

The user selects both the lines down the page in terms of ranges of account codes and the columns across the page by selecting one of 34 available column codes.

Separate financial statements can be produced for different user defined categories such as division, department and product group.

To cater for an international market SunAccount allows the use of dates expressed in any national format and of different sales tax structures.

The manuals are also compiled in terminology of wide international understanding. Hence the use of text files for screen wording permits programs to appear in any language and be consistently used by international corporations in each country of operation.

Ease of use is a fundamental feature. Every function is presented on a single menu screen. Help is given at any time by option prompts.

A consistent screen layout is used throughout with headings, item descriptions and error messages occurring in fixed locations.

The package documentation comes in three parts.

The tutorial manual is based on a complete demonstration ledger supplied with the system which can be used for practice.

The reference manual which describes each function, contains examples of all screens and reports detailed error message.

A reference card summarises the key information in the reference manual and an installation manual provides step-by-step installation procedures for Lisa and information about the operating system which will allow you to process your own ledger.

The system offers both public and private user hooks in the system.

The public hooks are published in the installation manual.

Private hooks are disclosed only in the case of a proven technical need, but we do bear in mind that the needs of the user come first.

The most common case of public user hooks is in form printing in case the user requirement exceeds the scope of the in-built parameters.

Name: SunAccount
Publishers: Systems Union
Price: £1,250

THIS month we shall look at two topics. I shall continue the graphics package with a set of routines to plot histograms, but first a look at a perennial problem – moving a program past the graphics pages.

A constant complaint from Apple users is "Why are the graphics pages in such an awkward place?"

The memory map in Figure I illustrates the problem. The graphics pages are in the middle of memory in a modern Apple.

Historically, the reason for this is easy to understand – in the early days memory was expensive and the basic Apple had only 16k of memory.

Hi-res page 1 was then at the top of memory. The next step was 24k where the top of memory was occupied by hi-res pages 1 and 2.

Unfortunately, as the memory grew the pages stayed put – creating maximum inconvenience. None of this would matter if the Apple made a distinction between graphics memory and program memory, but it doesn't.

It thus falls upon the user to make sure that the program and variables don't stray into the graphics pages if you intend to use them.

The first revelation of this problem for many people is very striking. It occurs when they add graphics to a long program only to find that the first call of HGR, which clears the graphics memory, brings the program to a grinding halt. Worse still is to come – listing the program reveals that half of it is now erased!

It's as well to consider this problem now since the graphics package started last month will grow to such a length that action will have to be taken.

Several different approaches have been used in the past, but I shall consider only the simplest.

This consists of moving the

> ### 'Make sure variables don't stray into graphics pages'

Solving a little problem...

PETER GORRY delves into the Apple's memory in search of its graphics pages

whole Applesoft program to start past the required graphics page.

It does leave the memory below the graphics page empty, but it can be used for machine code routines and shape tables – so it's not entirely wasted.

All that's needed to move the start of an Applesoft program

are a few simple POKEs:

POKE 103,1
POKE 104,64 Past hi-res page 1
POKE 16384,0

POKE 103,1
POKE 104,96 Past hi-res pages 1 and 2
POKE 24576,0

It is more convenient if a program can relocate itself and Listing I gives a short routine that can be incorporated into a program to achieve this.

The variable HP must be set to the desired page (1 or 2) and PN$ must contain the program name – so remember to change it if you rename the program.
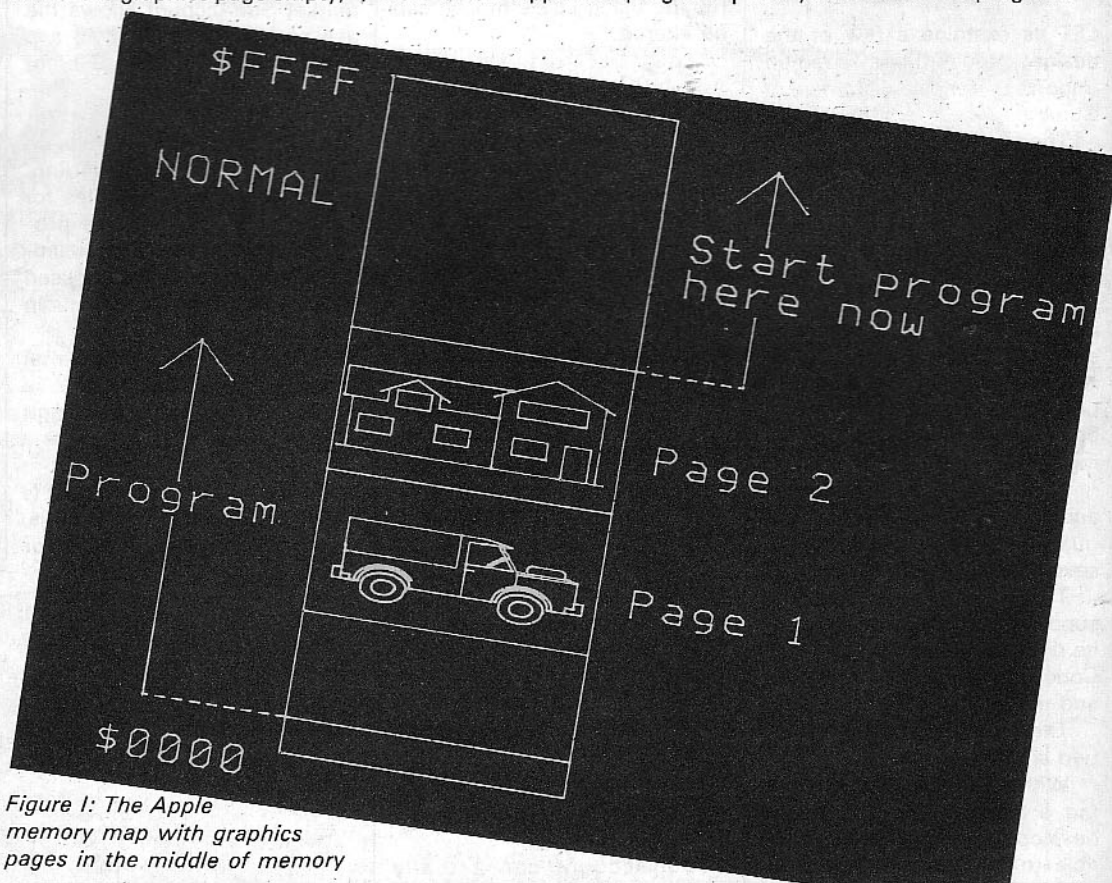
Figure I: The Apple memory map with graphics pages in the middle of memory

```
100  REM  PROGRAM "PROGTEST"
110  HP = 2:PN$ = "PROGTEST": GOSUB
     60000
120  :
130  :
140  :
150  :
160  END :

60000  REM

PROGRAM RELOCATOR


60010  ON HP GOTO 60020,60030: RETURN

60020  IF PEEK (103) < > 1 OR PEEK
       (104) < > 64 THEN  POKE 103
       ,1: POKE 104,64: POKE 16384,
       0: PRINT  CHR$ (4)"RUN";PN$
60025  RETURN
60030  IF PEEK (103) < > 1 OR PEEK
       (104) < > 96 THEN  POKE 103
       ,1: POKE 104,96: POKE 24576,
       0: PRINT  CHR$ (4)"RUN";PN$
60040  RETURN
```

Listing I
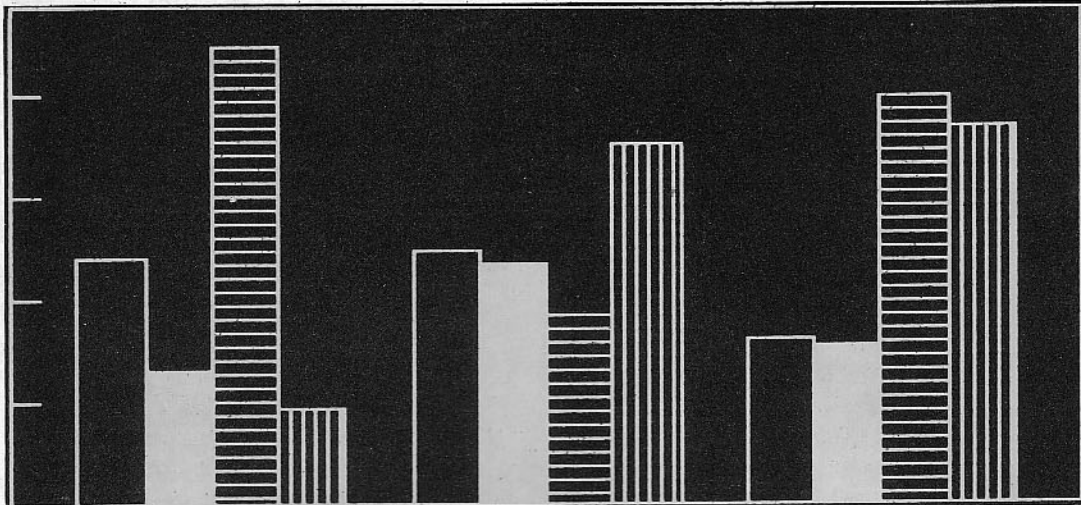
A histogram generated by the listing on Page 34

...and constructing problem solvers

Histograms can be useful business tools — we show how to make them

WE now come to the graphics package. Last month saw the first routines in what will eventually be a large *Apple User* library of integrated graphics routines.

The routines presented here provide for a wide variety of histogram plotting options with minimum effort by the user.

The philosophy is the same as last time — the maximum amount should be done in subroutines, hence keeping the controlling program as short as possible.

Before looking at the routines themselves it's worth considering what requirements a good histogram plotter should meet.

Back in *Windfall*, March 1982, Robert Beynon discussed the problems very thoroughly and the present routines are very much in the same spirit, although the implementation is a little different.

The list of requirements I drew up for the routines to meet is:

● Bars can be adjacent or separated by a space.
● Bars can be grouped in sets.
● Bars can be of different colours and shading.
● Bars will be "clipped" at the plotting area edges — no checking by the user is required.
● Calculation of the bar width is automatic.
● Bars can be stacked on top of each other.
● Vertical and horizontal bars are allowed.

Implementing these options means that a considerable amount of information is needed by the histogram routines. This information is stored in an array ZH.

ZH 1 Number of groups.
   2 Number of bars in group.
   3 Present group number.
   4 Present bar number in present group.
   5 Bar base value — user units.
   6 Bar height — user units.
   7 Bar shading 0-3.
   8 Format: 0 vertical, 1 horizontal.

The first two values determine the layout of the complete histogram, the rest supply information about the particular bar presently being plotted.

Suppose we want to display the sales of screws, nails and hammers for the years 1980, 1981, 1982 and 1983. We have three bars to plot for each year and four years so our first two numbers will be:

ZH(1) = 4 Number of groups (years)

ZH(2) = 3 Number of bars in group (items)

In principle there is no limit to the values you can use, but in practice the resolution of the Apple screen soon determines an upper limit.

The only special value is ZH(1)=0 which forces all the bars to be next to each other, with no separating spaces.

The only other value which needs comment is ZH(5). This is used to set the base value of the bar.

Normally this would be set to make the bars lie along the bottom of the plotting area. However, if you want to produce "double decker" bars you only need to set ZH(5) to the height of the lower half and the second half will be drawn on top.

We can now look at each routine in turn to see how this information is used. They are numbered to follow on from last month and, since they make use of some of last month's routines, they should be typed in after them.

Since the basic element of any histogram display is the bar itself, the starting point is the routine, at 40800, to draw a box.

Box routine

The coordinates of the box are all in user values and converted into screen points internally. The information required is the bottom left hand corner of the box (ZX,ZY), its width (ZW) and its height (ZH).

Finally a shading option (ZS =0,1,2,3) allows for the box to be open, solid, horizontally shaded or vertically shaded.

The box is plotted in the presently set colour.

BOX can be used independently of the histogram section to draw boxes in general and it also provides an ERASE function if a solid black box is chosen.

Histogram Plotter

This routine controls the plotting of an individual histogram bar.

Its principal task is to use the
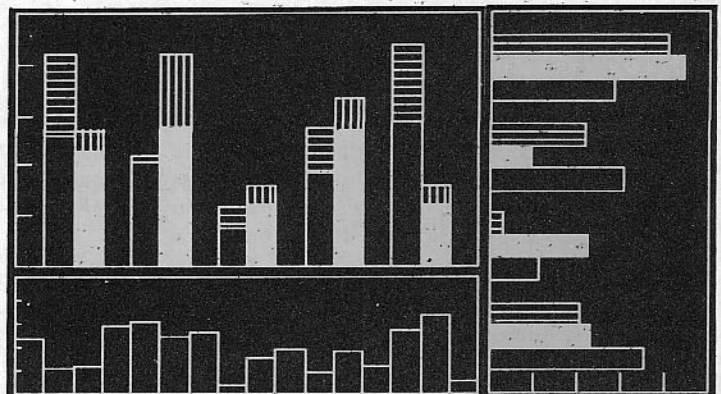
Example of the flexibility created by calling the routines three times

GRAPHICS

parameters in the ZH array to calculate ZX, ZY, ZW, ZH and ZS. It then uses BOX to perform the actual drawing.

Histogram Controller

This controls the plotting of the complete histogram. Its job is to ensure that all the bars are called and in the correct order. It sets up various parts of the ZH array before calling Histogram Plotter.

The routine layout is flexible and by no means unique. It is designed to be easily modified to suit your particular colour shading, stacking or grouping requirements.

Line 41480 would normally set the bar height from an array of data, rather than the random number generator used in the demo.

Example Program

This sets up the user and screen values, draws a border and then calls the histogram controller. Altering the values in line 180 will produce a wide variety of histogram plots.

So have fun.

```
100  REM

EXAMPLE PROGRAM

110  ZC = 3:ZP = 1:ZF = 1: GOSUB 4
     0000: REM  SET PAGE
120  ZM(1) = 0:ZM(2) = 1:ZM(3) = 0
     :ZM(4) = 1: REM  USER VALUES

130  ZM(5) = 0:ZM(6) = 260:ZM(7) =
     170:ZM(8) = 0: REM  SCREEN V
     ALUES
140  GOSUB 40200: REM   SET MAPPI
     NGS
150  ZG(1) = 0:ZG(2) = 5: GOSUB 40
     400: REM  BORDER AND TICK MA
     RKS
160  REM  NOW PLOT HISTOGRAM
170  REM   3 GROUPS, 4 BARS IN GR
     OUP, ZERO OFFSET, VERTICAL B
     ARS
180  ZH(1) = 3:ZH(2) = 4:ZH(5) = 0
     :ZH(8) = 0
190  GOSUB 41400: REM  PLOT HISTO
     GRAM
200  END :


40800  REM

BOX ROUTINE

40810  REM  DRAW A BOX CLIPPED AT
       THE PLOTTING LIMITS
40820  REM  ZX,ZY BOTTOM LEFT COR
       NER
40830  REM  ZW,ZH BOX WIDTH AND H
       EIGHT
40840  REM   ZS = 0,1,2,3 OPEN, S
       OLID, HORIZONTAL, VERTICAL S
       HADING
40850  XP = FN XCN(ZX):YP = FN Y
       CN(ZY): REM  SCREEN VALUES
40860  XP = INT (XP + .5):YP = INT
       (YP + .5)
40870  ZA = FN XCN(ZX + ZW):ZB =
       FN YCN(ZY + ZH)
40880  ZA = INT (ZA + .5):ZB = INT
       (ZB + .5)
40890  IF XP < ZM(5) THEN XP = ZM
       (5): REM  CLIP VALUES
```

```
40900  IF ZA > ZM(6) THEN ZA = ZM
       (6)
40910  IF YP > ZM(7) THEN YP = ZM
       (7)
40920  IF ZB < ZM(8) THEN ZB = ZM
       (8)
40930  IF XP > ZA OR YP < ZB THEN
       RETURN : REM  NOT VISIBLE
40940  HPLOT XP,YP TO ZA,YP TO ZA
       ,ZB TO XP,ZB TO XP,YP
40950  ON ZS GOTO 40960,40970,409
       80: RETURN
40960  FOR ZI = XP TO ZA: HPLOT Z
       I,YP TO ZI,ZB: NEXT : RETURN
       : REM  SOLID
40970  FOR ZI = ZB TO YP STEP 4: HPLOT
       XP,ZI TO ZA,ZI: NEXT : RETURN

40980  FOR ZI = XP TO ZA STEP 3: HPLOT
       ZI,YP TO ZI,ZB: NEXT
40990  RETURN :


41000  REM

HISTOGRAM PLOTTER

41010  REM   PLOTS HISTOGRAM BAR
       ACCORDING TO VALUES IN ZH(1)
       -ZH(8)
41020  REM  1  NO OF GROUPS
41030  REM  2  NO OF BARS IN GROU
       P
41040  REM  3  PRESENT GROUP
41050  REM  4  PRESENT BAR IN GRO
       UP
41060  REM  5  BASE VALUE
41070  REM  6  BAR HEIGHT
41080  REM  7  BAR SHADING , 0-3
41090  REM  8  0,1 VERTICAL, HORI
       ZONTAL BARS
41100  IF ZH(8) = 1 THEN GOTO 41
       180: REM  HORIZONTAL BARS
41110  IF ZH(1) = 0 THEN GOTO 41
       140: REM  NO GROUPS
41120  ZW = (ZM(2) - ZM(1)) / (ZH(
       1) * ZH(2) + ZH(1) + 1)
41130  ZX = (ZH(3) + ZH(4) - 1 + (
       ZH(3) - 1) * ZH(2)) * ZW + Z
       M(1): GOTO 41160
```

```
41140  ZW = (ZM(2) - ZM(1)) / ZH(2
       )
41150  ZX = (ZH(4) - 1) * ZW
41160  ZY = ZH(5):ZS = ZH(7):ZH =
       ZH(6)
41170  GOSUB 40800: RETURN : REM
       PLOT BAR
41180  IF ZH(1) = 0 THEN GOTO 41
       210: REM  NO GROUPS
41190  ZH = (ZH(4) - ZH(3)) / (ZH(
       1) * ZH(2) + ZH(1) + 1)
41200  ZY = (ZH(3) + ZH(4) - 1 + (
       ZH(3) - 1) * ZH(2)) * ZH + Z
       M(3): GOTO 41230
41210  ZH = (ZH(4) - ZH(3)) / ZH(2
       )
41220  ZY = (ZH(4) - 1) * ZH
41230  ZX = ZH(5):ZS = ZH(7):ZW =
       ZH(6)
41240  GOSUB 40800: REM  PLOT HOR
       IZONTAL BAR
41250  RETURN :


41400  REM

HISTOGRAM CONTROLLER

41410  REM  CONTROLS PLOTTING OF
       COMPLETE HISTOGRAM
41420  REM  USES INFO IN ZH ARRAY

41430  FOR ZK = 1 TO ZH(1): REM
       GROUPS
41440  FOR ZJ = 1 TO ZH(2): REM
       BARS IN GROUP
41450  ZA = ZJ - 1
41460  IF ZA > 3 THEN ZA = ZA - 4
       : GOTO 41460: REM  SELECT SH
       ADING
41470  ZH(7) = ZA: REM  SHADING
41480  ZH(6) = RND (1): REM  HEIG
       HT - SUBSTITUTE ACTUAL VALUE
       NORMALLY
41490  ZH(3) = ZK:ZH(4) = ZJ: REM
       GROUP AND BAR NO
41500  GOSUB 41000: REM  PLOT HIS
       TOGRAM
41510  NEXT ZJ: NEXT ZK
41520  RETURN :
```

Index your Forth discs

By MAX PARROTT

THE one thing which worries me about Forth is the profligate use of disc space when saving programs (I'm always short of discs) and the lack of an easy way to see what has been saved on them.

I decided if you can't beat 'em, join 'em and so I now dedicate the first block of each disc to an index, maintained by the user, of the contents.

The first command is INITCAT, which should be used with caution as it fills block 1 with zeroes without any second warning being given, but as it is a command only used once with each new disc there shouldn't be any real problem.

The index is maintained by the command ADDTOCAT which is followed by the first block, then the last block of the program and finally the name, thus:

ADDTOCAT 101 107 CALENDAR

The block numbers must be in the range 1-140 to be valid, otherwise a beep is given, the stacks are cleared and no further action is taken.

The name can have up to 22 characters in it. Any more will be truncated.

Up to 42 filenames may be stored, an attempt to add a forty third will elicit the response "CATALOG FULL".

Filenames may be deleted by the command DELETE, followed by the name, thus:

DELETE CALENDAR

will delete the first match found in block 1.

The index is displayed by the command CATALOG. This first emits a formfeed to clear the screen, so if you are planning to print an index for your disc jacket plan accordingly.

If any readers, more familiar with Forth, know of other approaches to disc indexing I, and I'm sure other readers, would be most interested in hearing about them.

```
First screen is 2

  0    ( CATALOG, uses BLOCK 1 to hold an index of the disc )
  1    DECIMAL
  2    : INITCAT 1 BLOCK 1024 0 FILL UPDATE ;        ( fill with zeroes )
  3
  4    : FINDSPACE DUP BEGIN DUP C@ SWAP 1+ C@ OR WHILE
  5              24 + DUP 1 BLOCK 1020 + U< NOT IF
  6                 ." CATALOG FULL" ABORT
  7                 THEN
  8              DUP REPEAT ;
  9    : CHECK DUP DUP 1 < IF 7 EMIT ABORT THEN
 10              140 > IF 7 EMIT ABORT THEN ;
 11    : ADDTOCAT PAD 40 32 FILL 1 WORD COUNT PAD SWAP CMOVE
 12    1 BLOCK FINDSPACE DUP DUP 0 0 PAD 1- CONVERT SWAP DROP
 13    SWAP CHECK ROT C! 0 SWAP 0 SWAP CONVERT SWAP DROP SWAP CHECK
 14    ROT 1+ C! 1+ SWAP 2 + 22 CMOVE UPDATE ;
 15    3 LOAD 4 LOAD

-------------------------------------------------------------------

 16    ( second screen of CATALOG )
 17
 18    : CATALOG 12 EMIT 0                        ( Positional counter )
 19    ." FIRST  LAST" 28 SPACES ." FIRST  LAST" CR       ( Heading )
 20    1 BLOCK 42 0 DO DUP C@ IF DUP 1+ C@   ( See if first byte zero )
 21                 IF SWAP 1+ DUP        ( & then second )
 22                    2 MOD IF CR THEN   ( check counter )
 23                    SWAP DUP C@ 4 .R 1+ ( print bytes )
 24                    DUP C@ 7 .R 3 SPACES 1+
 25                    DUP 22 TYPE 3 SPACES     ( & title )
 26                    22 + ELSE 24 + THEN     ( or loop to )
 27                 ELSE 24 + THEN            ( next entry )
 28              LOOP CR DROP DROP UPDATE ;
 29
 30
 31

-------------------------------------------------------------------

 32    ( Last screen of CATALOG )
 33
 34    : FINDFIT DUP 0 SWAP PAD BEGIN    ( Match names in PAD & BLOCK )
 35    DUP C@ ROT DUP C@ ROT = WHILE
 36       ROT 1+ SWAP 1+ ROT 1+ REPEAT
 37    DROP DROP 21 > IF 1 ELSE 0 THEN ;
 38
 39    : DELETE PAD 40 32 FILL          ( Prepare PAD for name of file )
 40    1 WORD COUNT PAD SWAP CMOVE          ( Move name into PAD )
 41    CR PAD 22 TYPE                      ( Print the file name )
 42    1 BLOCK 22 - BEGIN 24 + DUP 1 BLOCK - 1020 > IF CR
 43                    ." not found" ABORT THEN
 44              FINDFIT UNTIL
 45    ." deleted.  UPDATE CATALOG? Y/N" KEY 89 = IF 2 - 24 0 FILL
 46    ." REMOVED" ELSE ." NOT UPDATED" DROP THEN CR UPDATE ;
 47

-------------------------------------------------------------------
```

LISA's first birthday has been marked by the launch of three completely new version's of Apple's revolutionary micro — together with a dramatic reduction in price.

In last month's *Apple User* we reported how in its first 10 months' life the price of Lisa had dropped from £7,999 to £5,500. Now its cheapest version is £2,625.

There are three models in the new Lisa 2 range, offering flexibility in both price and performance.

All have 512k of internal memory, plus a built-in 3½in microdisc drive which provides 400k storage — far more than the conventional 5¼in floppy.

The basic machine is the Lisa 2. Additional mass storage is provided by Lisa 2/5, which has

THERE'S A NEW LOOK FOR LISA...

an external 5mbyte hard disc — the Profile — and Lisa 2/10; which has a built-in 10mbyte hard disc.

RAM can be doubled to 1mbyte with an additional chip.

All three can run existing Lisa software, as well as programs designed for the new Macintosh.

The compact 3½in microdisc drive used in the new Lisa range is identical to the one used in Macintosh.

The faster speed of the microdiscs, together with an upgraded operating system, vastly increase the performance of the new machines. They are three times faster than the original Lisa.

Lisa 2/5 and 2/10 have sufficient mass storage to run many third-party software packages and a wide range of programs based on the Unix operating system.

When expanded to the full 1 mbyte of RAM, both the 2/5 and 2/10 can run the complete Lisa Office System, with its fully-integrated software.

The Lisa Office System allows simultaneous operation of multiple Lisa programs, as well as cut-and-paste integration that lets users easily move information — text, calculations and graphics — from one application to another.

Lisa 2/5 is being launched at £3,375, and the 2/10 at £4,135.

The basic Lisa 2 is expected to appeal to users who wish to run Macintosh software but prefer to have expanded memory and mass storage. Macintosh itself is limited to 128k memory.

A large number of new Lisa software packages are now being developed to run on the Lisa range. Among UK software houses involved in this development are Blyth Computers, Systematics, LBT and Intelligence UK.

The UK company Logica is one of the developers of multi-user applications that will run under the Unix operating system.

The new Lisa (left) compared to the original version (below)

... but don't worry if you bought the first version

FULL marks to Apple for looking after its end users — particularly those people who, in the past year, shelled out £6,500 or even £7,500 to buy Lisa hardware that is now "old-fashioned".

The company is providing free hardware upgrades to existing Lisa owners to bring their machines into line with the newly released Lisa 2 series.

The early buyers had the courage to gamble on a pioneering machine that although revolutionary, had an unproven market. The up-grade to the new technology is offered free of charge until June 1. It can be performed by authorised Lisa dealers in less than a half an hour and replaces the two 5¼in drives with the higher-speed 3½in drive featured in Lisa 2. This will allow early Lisas to run Macintosh software.

An upgrade which offers additional hard disc storage capacity will also be provided for a charge.

"When people buy Apple computers, they must be assured the value of their investment will be protected", said Keith Hall, Apple's marketing director.

"With the free upgrade, current Lisa owners will be given the same advanced disc storage technology found in our newest family of products.

"They'll be able to take advantage of the many software packages being written for Macintosh, as well as Lisa".

I WAS recently able to print a Visicalc model which consisted of 43 columns (A to AR) of nine characters each. In other words, I printed a document which was 387 characters wide.

Considering that even the best printers cannot print a document which is more than 255 characters wide, how was it possible to print 387 characters in a single continuous line?

The secret lies in using a program called Sidevise, produced by Vriegate Software of Cape Town, which as the name implies, prints out large worksheets down the page instead of across it.

Using continuous stationery, the particular document I printed was 44 inches wide, and reading from it was like reading from the Scrolls.

There are a few things that every prospective user of Sidevise ought to know before using the program.

First of all, any worksheet to be printed must first be saved in print format, that is saved with /PF or /PD.

This is a very simple process and if you have not used /PF before, just follow the instructions on page 3-47 of the VC manual.

But what is of utmost importance is to make sure that Sidevise is compatible with your printer and the printer interface card in your Apple (see illustration).

Even if your printer and interface card are not specified

VISICALC - a sideways view

IN this month's Spreadsheet NICK LEVY reviews Sidevise, a program which allows you to print Visicalc models of any width, offers some more practice in replicating and introduces the @CHOOSE command.

And VIC TANN delves into project planning and the use of Visicalc in critical path analysis.

in the Sidevise configuration, it is still worth trying the program (without obligation) to see if it works with your equipment.

Incidently, saving a VC file with either /PF or /PD has another very useful application. If for example you write a report using a word processor and want to include a Visicalc file, then provided that that file was saved in print format (/PF or /PD

as well as with /SS) the word processing program will treat that file as if it was written with the word processor.

I cannot guarantee that this will work with every word processor in the market, but the Applewriter IIe certainly has no difficulty in loading VC files saved with /PF or /PD.

Allied to the kind of problems that can be solved by using

Sidevise is one that sooner or later almost every Visicalc user is likely to be faced with.

Suppose you have a row of figures spread over 20 columns so that only a few of them are visible to you at any one time.

How can you then bring those figures into your screen so that you can view them all together in a single column?

Experiment with the following exercise. Insert a figure at random in cell J10. Then move to cell K10 and enter the formula 12+J10 (or any other formula linked to the number in J10).

Following that replicate K10 from L10 to AC10 (imagine these are the totals of the columns above those figures).

Next go to A1 and enter 1 in A1. Go to A2 and type 1+A1. Replicate the formula in A2 from A3 to A20 (Relative).

Next move to B1 and write the formula: @CHOOSE(A1, J10. AC10). The number which you inserted in J10 will be copied into B1.

Next replicate the formula in B1 from B2 to B20 (Relative, No change, No change) and the whole row of figures from J10 to AC10 will be displayed to be viewed at a single glance from B1 to B20.

You could of course, go to B1

```
SIDEVISE                    CONFIGURATION

  1)...PRINTER
  2)...INTERFACE CARD
  3)...PAPER SIZES
  4)...ALL
  5)...NONE


CHANGE WHICH ? (1-5) ■
```
Set-up menu
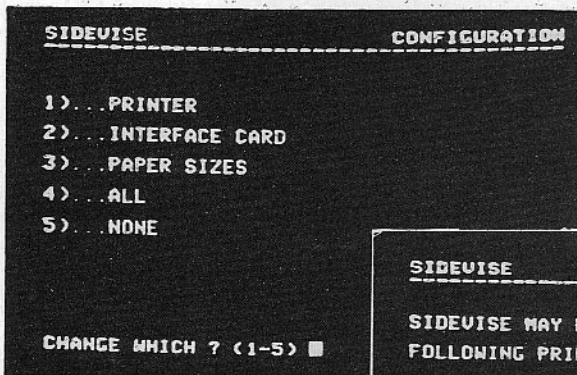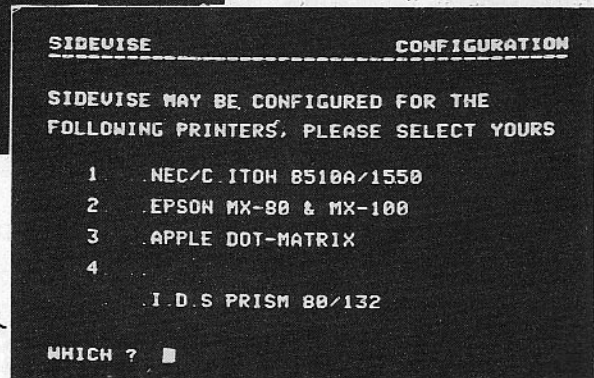
```
SIDEVISE                    CONFIGURATION

SIDEVISE MAY BE CONFIGURED FOR THE
FOLLOWING PRINTERS, PLEASE SELECT YOURS

    1    .NEC/C.ITOH 8510A/1550
    2    .EPSON MX-80 & MX-100
    3    .APPLE DOT-MATRIX
    4
         .I.D.S PRISM 80/132

WHICH ? ■
```
Choose printer by number
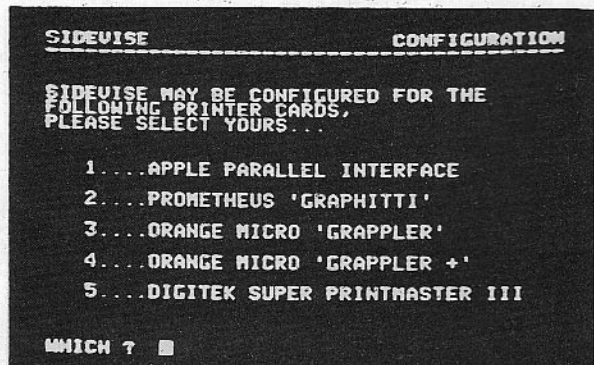
```
SIDEVISE                    CONFIGURATION

SIDEVISE MAY BE CONFIGURED FOR THE
FOLLOWING PRINTER CARDS,
PLEASE SELECT YOURS...

    1....APPLE PARALLEL INTERFACE
    2....PROMETHEUS 'GRAPHITTI'
    3....ORANGE MICRO 'GRAPPLER'
    4....ORANGE MICRO 'GRAPPLER +'
    5....DIGITEK SUPER PRINTMASTER III

WHICH ? ■
```
Choose printer interface by number
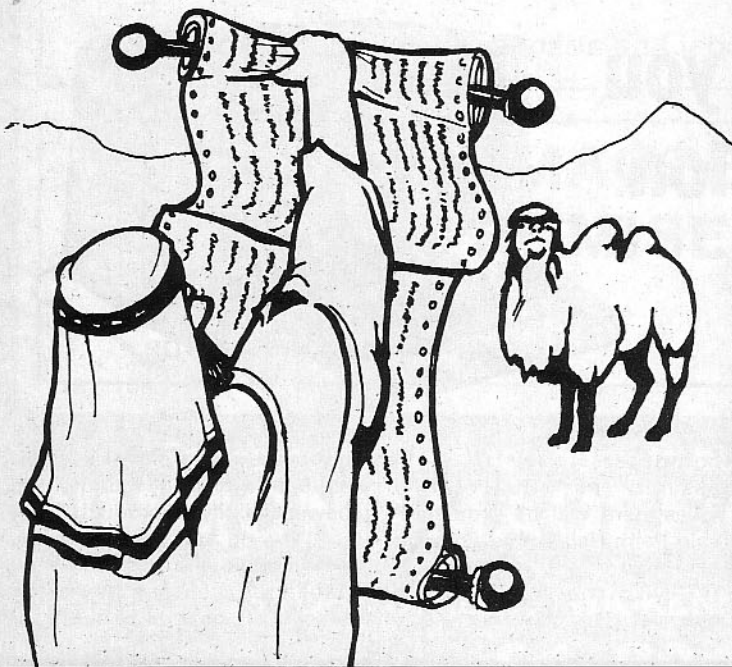
and type +J10 and then move to B2 and type +K10 and so on, but that method could be most laborious.

Why not try the above exercise, it will give you some replicating practise as well as an example of using the @CHOOSE command which may be you have been using for the first time.

By the way, I noted that many Visicalc users get worried on typing a range such as J10. AC10, when they discover that VC will display their entry as J10 . . . AC10.

If that happens don't let it concern you, it wasn't caused by a typing error. That's how Visicalc was programmed — to respond with three ' . . . ' to your single ' . '.

Happy Sidevising.

Take the right path towards planning

SURPRISINGLY, the operations used in preparing a critical path analysis are very similar to those used in Visicalc. This is because in both the end product is a network of interrelated items.

The method of creating a CPA is also very suitable for Visicalc — that is, setting up a network or related items on a calendar. This program has been developed in order to make the best use of Visicalc techniques in preparing a CPA.

When the spreadsheet has been prepared all other planning functions can be carried out — checking progress and updating the program, extracting parts for different departments, preparing budgets and estimates of labour requirements etc.

We all know the advantages of Visicalc over using a rubber and pencil when changing spreadsheets. These advantages are also of great help when amending and updating planning charts.

The updating can be carried out with a minimum of effort, several options can be considered and yet the original chart can be kept intact.

The first requirement in planning a project is to define the main elements. This can be done in a block diagram. (Figure I).

Visicalc is not of much assistance here, but if the diagram is "drawn" onto a spreadsheet it can serve as an index for the full program by inserting the grid square references of each part at a later stage. Figure I shows such a flow chart.

This is purely a typing exercise, but serves to show the symbols used, for example, <<<,>>>,VVV, in corelating the elements.

These are inserted by typing " (shifted 2) so that a label is expected and then typing the symbol. The symbols can be preceded by blank spaces if required.

Before starting on the program itself it is necessary to give some thought to a coding system.

In order to show as much of the program on screen as possible (and later on the printer) much use is made of narrow columns down to the minimum 3 (/GC3).

It is therefore necessary to devise a three letter coding system.

The first letter indicates the critical items:

- * indicates a critical item
- — indicates a normal work item
- F indicates spare time (or a float)

The second letter indicates the type of work being carried out. This might be:

- E Excavation
- C Concreting
- B Brickwork
- J Joinery

The third letter identifies the area of work:

- 1 Phase 1
- 2 Phase 2
- A Alterations to existing building
- E External work (outside the factory)

These three letters are combined to give, for example:

- *EE Critical path item, excavation, external.
- —J1 Non-critical joinery item, phase 1.
- FB2 Float (spare time) to complete brickwork, phase 2.

Starting with a blank Visicalc sheet, give it a title and leave a few lines blank for notes. The next move is to establish a calendar down the left hand side.

● At A11 enter '1'
● At A12 input the formula (A11+1)
● Replicate this formula down to A60 and, assuming these to be weeks, this will give room for a year's work.
● If starting from a fixed date the next step is to insert Christmas and Easter holidays and any factory shutdowns.

If Christmas is on week 23

Figure I: Getting the overall picture. This helps to plan out the Visicalc sheet

type "Xmas" at B33 and replicate across to Z33. This serves as an automatic reminder as the program is developed across the sheet.

We now have the basic calendar.

As the CPA is developed you will wish to indicate related activities by showing cross links. These are indicated by combinations of arrows and dashes. They can be inserted very easily in the program without upsetting the calendar.

● GOTO A34. Type /IR. This gives you a free line in a newly-created space in the calendar. This line can now be filled with "- - - - - VVV - - >>>" etc.

The week numbers above and below are unaffected, as is the data already entered. The chart can then be continued from the next week, without any adjustment.

The coding system takes up three letters, then insert a blank and type in a full description of the work.

For example, "DA detailed design by architects"

Enter this into a blank cell on the spreadsheet and then key /GC3. You will see '–DA' in the cell, but above, in the top window, you will see the full description. This will be useful as reference while compiling the program.

Now key '/GC39' and you will see the full title in the cell. This will be used later to print out columns in full.

Having established our coding system we can now begin to enter data. If we have items for design work by several offices our coding will be:

–SC Scheme design by civil engineers

–SA Scheme design by architects

–SE Scheme design by electrical engineers

–SM Scheme design by mechanical engineers

–IP Initial work by planning dept.

All the details are entered with a first symbol '–' to start with. When parts of the critical path are found the first symbol is changed to '*'. Items with spare time are then prefixed with 'F'.

As shown in Figure II, the costs and scheme drawings are required within 11 weeks and after discussing the work involved with the staff concerned the following program is drawn up (printed out at /GC3).

The critical path has been identified and the flow of information between departments is shown after five weeks.

The item in square C11 is actually as follows: "–SC civil scheme design", as explained above.

In entering the data much use is made of replicating techniques to save typing time. For example, the entry is made in C11 and replicated C11...C14, then C18...C19.

This way the full length description is stored at each location. Alternatively the entry could be replicated straight from C11 to C19 and the floats, etc, inserted later.

After week five it was necessary to show an important exchange of information.

Two additional rows were inserted (/IR) which moved the numbered weeks down the sheet, but did not change the calendar set up in column A. Column B has been left blank to be used for noting key events in the program.

As the program is developed it can be useful to retain columns A and B on the screen by using titles.

Position the cursor at B1 and key /TV. This will keep columns A and B on the screen as a reference as the program is developed across many columns. Items and dates of particular importance are shown in column B as they arise.

They should be fully typed out (up to 39 characters) and later printed out (/GC39) alongside the calendar as a very useful guide to the project.

The critical path is established graphically rather than by calculation.

This is a departure from the method of earliest and latest start and finish dates and is more suitable to this Visicalc approach.

As each section of the CPA is analysed the critical items are identified. By editing the cells (/E) the first symbol of critical items is changed to '*'.

The float periods are next inserted to fill all the gaps and spaced to suit the individual departments. Earliest start dates can be established for non-critical parts of the program by deleting all the Fs before the work, and latest start dates by inserting all the Fs before the work.

Joining up the stars on the printout gives us the critical path so far. The Fs show the float for each department and the arrows show the exchange of information – that is, the dependant activities.

As more activities are entered into the program the critical items will become apparent and adjustments will have to be made in order to achieve key dates.

This, of course, is the normal planning procedure, and can be carried out very efficiently using Visicalc.

Moving an activity forward four weeks is simply achieved by blanking the four lowest entries and then replicating the item upwards for a further four squares.

Now for a really nice surprise. Turning the printout through 90


```
1    -SC*SAF  -SMF   -IP
2    -SC*SAF  -SMF   -IP
3    -SC*SA-SEF  F    F
4    -SC*SA-SE-SM-SO-IP
5    F   *SA-SE-SM-SO-IP
     U    U    U    F
     -->--<-----<UUU-IP
6    -SC-SA-SE-SM*SO-IP
7    F   -SAF  -SM*SQF
8    F   -SAF  -SM-SQ-IP
9    F   -SAF  -SM*SQF
10   F   -SAF  F   *SQF
11   F   F    F   F   *SQF
     COS<=====================
12
13
14
     GO ======================
15   -IC-PA-EE
              U------------CB
16   -IC-PA
17   -IC-PA
18   -IC-PA
19   -IC-PA
20        -PA
21
22
23XMAXMAXMAXMAXMAXMAXMAXMA
24XMAXMAXMAXMAXMAXMAXMAXMA
```
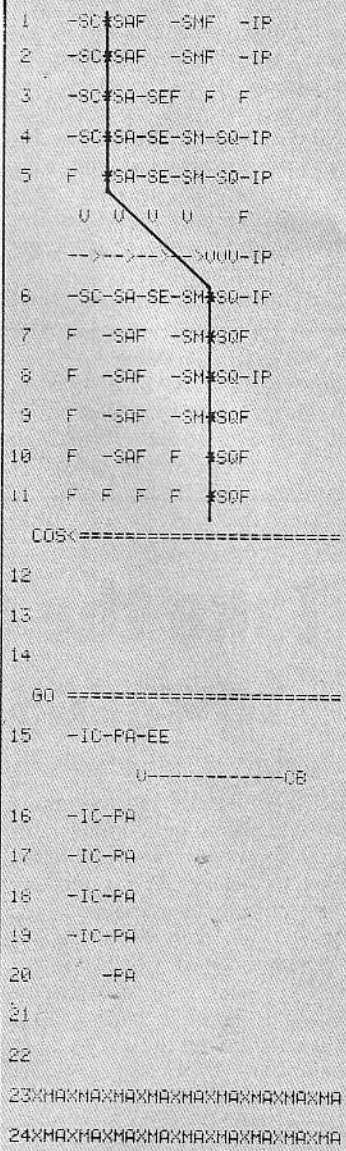
Figure II: Printout at three columns wide. Join up stars to obtain critical path.

degrees immediately gives us the bar chart!

The critical items can now be joined up and the bar chart can be used instantly. All that is needed is a key to the coding system used.

The program can now be developed to your individual needs. There are many possibilities at this stage – for example, by varying the column widths patches of the program can be printed out separately for each department.

In addition, if several of these project charts are being prepared for the same client they can be combined using DIF file techniques to form individual planning charts for each department.


```
1   -SC CIVIL SCHEME DESIGN   *SA ARCHITECT SCHEME DESIGN F
2   -SC CIVIL SCHEME DESIGN   *SA ARCHITECT SCHEME DESIGN F
3   -SC CIVIL SCHEME DESIGN   *SA ARCHITECT SCHEME DESIGN -SE ELECT SCHEME DESIGN
4   -SC CIVIL SCHEME DESIGN   *SA ARCHITECT SCHEME DESIGN -SE ELECT SCHEME DESIGN
5   F                         *SA ARCHITECT SCHEME DESIGN -SE ELECT SCHEME DESIGN
    U                          U                           U
    --                        --                          --
6   -SC CIVIL SCHEME DESIGN   -SA ARCHITECT SCHEME DESIGN -SE ELECT SCHEME DESIGN
7   F                         -SA ARCHITECT SCHEME DESIGN F
8   F                         -SA ARCHITECT SCHEME DESIGN F
9   F                         -SA ARCHITECT SCHEME DESIGN F
    F                         -SA ARCHITECT SCHEME DESIGN F
    F                         F                           F
    <==                       ==========================================
```

Figure III: Printout at 26 columns wide

INCREDIBLE Jack claims to handle word processing, letter writing, form letters, record retrieval, mailing labels and calc analysis in one package.

That is, no more swapping between word processor, spreadsheet and database programs to produce the kind of output you really require.

This a novel approach to me as an Apple II user —

generally when reviewing software I play with it for a while getting to know it, then boot my word processing program to start typing my thoughts.

Of course, as soon as I want to check something about the package I have to leave the word processor and re-boot the program under review, check and then return to the word processing.

With Jack I am able to check the file handling or whatever from within the part of the package.

Furthermore, letters and reports can have embedded calculations, as can records.

Jack comes in one small plastic disc library case along with manual, registration card, a quick reference card and an addendum to the manual.

Most of the manual is a tutorial in seven lessons. It is followed by a reference guide, some appendices and a reasonably clear index.

Although Jack is a single program, I will describe it as separate parts. Bear in mind, however, that the operations carried out in one part are, in general, equally applicable in all.

Incredible Jack will suit the micro novice

MAX PARROTT puts this integrated software package through its paces

I SPENT several afternoons using Incredible Jack on an Apple II Plus fitted with the Vision-80 card, a 16k expansion card, an Epson dot matrix printer and two floppy disc drives.

This is the minimum configuration which can be realistically used with Jack, although a 40 column screen is supported via the use of CTRL-A to flip from the left hand half of the screen to the right hand half and back again.
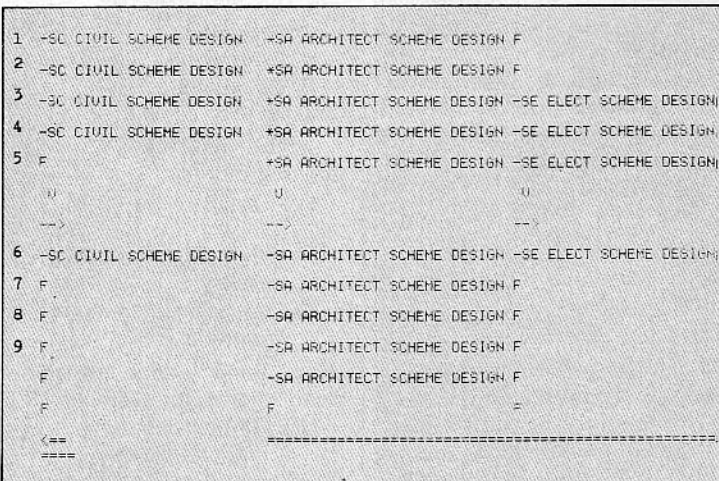
In case you haven't recognised it, this is typical of the Apple's UCSD Pascal system. In this case the UCSD Pascal runtime system was used. As the program disc is booted it automatically finds and uses an 80 column card if fitted.

The disc is protected, however, and makes a strange head noise as it carries on loading and subsequently verifying the program disc.

The screen presented consists of three main parts:

● Two lines at the top where the currently available options are displayed, known as the command line.

● A line at the bottom where current information on the file status is displayed — the status line.

● The main part of the screen, separated from the other two by two tabulation rules. Here is where all information, of whatever sort, is entered and is known as the work area.

The command line options consist of key words.

Pressing the key for the first letter of the word, either with or without holding down the CTRL key at the same time, brings up another set of options.

Moving backwards through the hierarchy of menus is simply accomplished by pressing ESC.

Once some form of keyboard entry is begun, the cursor moves to the work area and now only control characters will activate the menu options.

These menus and options are exactly the same whether you are word processing or recording data or calculating. Indeed nothing changes — you DO use the same screen for all purposes.

The first set of options available after booting the disc are Disc, File, Select and Print.

Before beginning any work, a file must be selected from the current work or data disc which resides in the second drive (or third or fourth if desired).

Normally the program disc stays the whole time in the first drive (except when copying files from disc to disc).

Pressing D or ^ D (for disc) brings the options Load, New-disc, Save, Remove, Copy (this takes five seconds).

Pressing L or > L displays the

files currently on the data disc in the work area unless it's a new, unformatted disc in which case Jack (after checking for valid data which takes eight seconds) will present the query "New disc (Y/N)".

Pressing Y begins the formatting.

Selecting a file is simple. Merely position the cursor over the file name and press Return. A new file is created by moving the cursor to a clean part of the work area using the cursor key, typing in the new name and pressing Return.

Inadvertant use of a name already on the disc isn't flagged as an error, but just brings up the old file.

Upper and lower case are treated as different, review and REVIEW are quite separate files, although this is not mentioned in the manual.

The first lesson of the tutorial covers the use of Jack as a word processor, so I shall begin here as well.

The best thing I can say about it is that it's rudimentary

when compared with packages such as Wordstar and Applewriter II.

Jack is written for the Apple IIe and makes full use of the four cursor keys and the TAB key — but it does provide alternatives for use on the Apple II Plus.

No other fast cursor moving commands are given except for pressing Return, which moves to the beginning of the next line.

For example, to move from the beginning of the article to this point took 1min. 18sec. By any standards this is very slow. Most of the time was taken up by constant disc accessing.

On reaching the end of a screenful of text the disc is accessed, presumably to save one block of text and load the next.

However, the file must still be saved when finishing by pressing ∧S, otherwise any changes made to the file will be lost.

Jack· is very slow at certain tasks. My other big moans about its word processing capabilities involve the lack of formatting facilities, either on screen or at print time.

For example, no easy means of centering text on the line, no global search for text strings, no global replacement of text, no means of underlining, indeed no means of embedding printer control codes in the text.

On the plus side, Jack allows double or triple columns of text to be set up and edited by setting the margins successively to the required position.

Once set, text outside them is untouchable without first repositioning the margins to include them.

Unhappily, once margins have been set, text cannot be

reformatted by simply changing their positions.

Upon printing, text may be left, right or full justified but there is no hyphenation facility for breaking words when using full justification.

Before printing starts a setup option may be selected and changed to suit the file.

This controls such things as the justification, the line spacing, which column at which to begin printing, etc.

In this way mailing labels can be printed from records and form letters are printed. The printing setup details can be saved along with the file if so desired.

Fields are the key to Jack's handling of records, calculations and form letters. In the course of entering text, typing a colon produces a colon, 10 spaces, then a caret. This forms the definition of a field.

The field space is alterable. One can delete or insert spaces as normal within the field space or write over the caret and print it again elsewhere.

In the absence of other commands the field is known by the single word immediately preceding the colon.

Once a record or form letter has been set out on the screen the text may be locked so that the field names are protected and data entry may proceed.

The TAB key (^I on the Apple II+) moves from field to field allowing data entry. Pressing the Return key moves the cursor to the start of the current field. This is convenient and allows easy editing of entries.

If the text is unlocked new fields can be entered into existing records and existing fields can be changed in length. This is a facility not found in many database programs.

Arithmetic and logic operations may also be carried out on the fields by invoking other field names plus appropriate operators into a "footer" for the field in question.

Merely position the cursor in a field and press > F. The cursor moves to a new area created at the bottom of the screen and the required calculations are entered.

For example, suppose I create a simple record such as:

Total:
Invoice Number:
Date:
Discount Class:
Net Amount:
Amount Paid:
Amount Due:

Then in field Net AMOUNT (with fields upper and lower case are treated identically) I can enter the following calculation:

AMOUNT: IF CLASS-
=1 THEN TOTAL*.6 ELSE
IF CLASS=2 THEN
TOTAL*.75 ELSE IF
CLASS=3 THEN
TOTAL*.9 ELSE IF
CLASS='' '' THEN
TOTAL ELSE "ERROR"

The calculation aspect of Jack is now shown because as soon as TOTAL and CLASS have numbers entered in their fields the correct value will be inserted into Net AMOUNT.

Records are searched in the GOTO (^G) command. Whatever selection rule has been entered will operate on the search until the selection rule has been changed. If no rule is entered all records are selected.

For example, with my small record given above I can search for all records where I'm owed more than £100 by entering SELECT: DUE > 100.00

If I only want to search those invoices up to a certain number (say 12000) then I can use:

SELECT: DUE > 100.00
AND NUMBER < 12001

Strings may be searched for, indeed part strings may be searched for using two dots as a wild card.

Thus .. 'di' .. will search the required field and find directory, disc, displays, non-digital, but not Digital, because case is distinguished.

There appears to be no easy way of searching without case distinction. A high speed search may be made on the key field, which is defined as the first field entered into the record design, not the first on the screen.

Only simple equality can be tested in the fast, key search. After making a selection rule it cannot be easily cleared. The only way I could subsequently view all records was to make a dummy rule such as SELECT: NAME = ''.

Finding the last record out of six using a unique field value took 14 seconds. Finding the first record took 6 seconds.

In total, Jack appears to me to be a free format database allowing calculated fields. Records may be accessed in various ways so that form letters and letters with calculated values within them may be printed.

Records may be copied from one file into another as long as the same fieldnames are used for both files.

The maximum capacities (quoting from the manual) are 1000 records for a file, depending on available disc capacity. Files cannot stretch across different discs.

Records can be sorted (numerically or alphabetically) but only at print time.

Conclusion

Jack is not a comprehensive word processor, but it may be used to create simple pieces of textual output, and if you are prepared to compromise on price/performance Jack could suit.

For a price tag of under £130, however, Jack does offer a wide range of facilities, readily accessible in one program, without the need to read bulky manuals or learn new skills for each facet.

The program's limitations may irritate the more experienced user. But it could suit the newcomer to the Apple.

```
File: Goto Clear Lock Save
+---+----+----+----+----(----+----+----+----+----+----+----+----)---+----+
       Dear : _ JOHN^,
             Last month you achieved the following
       sales : 12500^
       -------------------------------------------
             3% on the first $5000  : 150.00^
             5% on excess           : 375.00^
       -------------------------------------------
       Your total commission earnings $: 525.00^

       :Congratulations on a fine month.                    ^
---+----+---1----+----2----+----3----+----+----5----+----6----+----7----+----8
File:Start Demo   Record:  1 Line:   1 Colon: : Format: 7.2^  Lock:Y Blank:Y
```
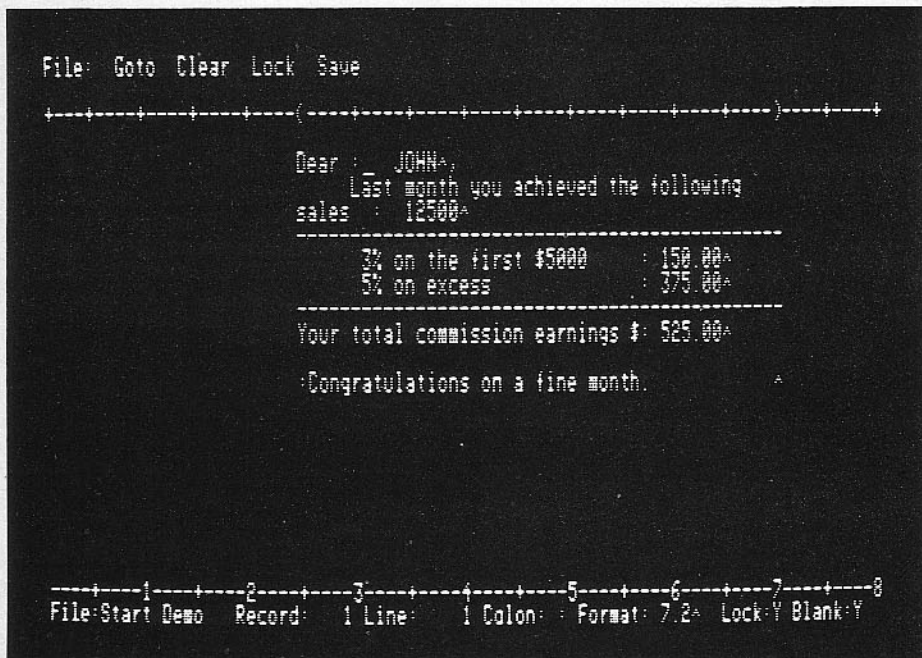
An example of interaction — words and numbers being processed together

Advanced Visicalc keeps its own counsel . .

I WAS very interested in Nick Levy's review of Advanced Visicalc for the Apple IIe in the January 1984 issue of Apple User.

He answered most of the questions I have had about this new version of Visicalc except one — can "ordinary" Visicalc read data files created by Advanced Visicalc?

I realise that the advanced features would not be operable, but if you only used ordinary commands to create an Advanced Visicalc, I wonder if it would be readable and useable by ordinary Visicalc.

I ask this because I use an Apple IIe at work and an Apple II Europlus at home. I could upgrade the work Visicalc, but I can't upgrade my home Apple.

I wonder if I were to upgrade the work Visicalc if I would be making it impossible to do work-related Visicalc at home.

I also would like to know if you are aware of any way to make 80 columns show on an Apple IIe with an Apple 80 column extended memory board when using ordinary Visicalc.

I need something similar to Videx's pre-boot disc for their 80 column card.

Thanks for the fine articles in Apple User. Nick Levy's columns in particular help Apple User to be one of the best Apple magazines available anywhere — **Robert M. Schneider,** Edinburgh.

● The ordinary and advanced versions of Visicalc are not compatible, even if only ordinary commands are used in the advanced version.

As a result, an upgrade of your work Visicalc would make it impossible to do work-related Visicalc at home.

The Apple IIe will not display 80 columns on screen when using an Apple extended memory card with ordinary Visicalc. As far as we know — short of some major alterations to the original program (which are not planned) — regretfully there is no way to achieve this.

Disabling reset bug

AS a regular reader I find your articles range from the amusing and ridiculous through to the sublime and extremely useful.

One such useful article was Dave Miller on disabling reset in the March 1983 issue. There is, however, one small bug in the code as it stands, in that it does not work after a string is used.

This is due to the fact that the code resets the HIMEM top of memory pointer, but does not reset the pointers to the start of string storage which the Basic command HIMEM: would do.

There are two simple solutions:

One is to set the string storage pointers at the same time as the top of memory pointers within the code. This involves a small change to the original code (see listing).

The second is to set HIMEM: 38373 ($95E5) immediately after BRUNing the program without altering the code from Dave Miller's article. Note that this makes the four machine code instructions, which set the top of memory in the code, redundant.

This is definitely the easier way out if you have already spent the time typing in the code and incorporating it within a program.

For someone who has spent hours wondering why sometimes their reset hangs and other times it seems to work, this could be the answer.

But beware if the program is run for a second time. Even after a "warm start" or "soft boot" —

call it what you will — the program will store the amended reset vectors in place of the originals and thus an immediate mode reset will probably hang.

This can be avoided by first PEEKing location 38375 ($95E7) which will contain 166 ($A6) if the code has already been run. Remember to reset HIMEM again as before. **W.G.A. Tulloch, Edinburgh**

```
]CALL-151

*95BDLL

95BD-   AD F2 03    LDA   $03F2
95C0-   8D E5 95    STA   $95E5
95C3-   AD F3 03    LDA   $03F3
95C6-   8D E6 95    STA   $95E6
95C9-   A9 E7       LDA   #$E7
95CB-   8D F2 03    STA   $03F2
95CE-   A9 95       LDA   #$95
95D0-   8D F3 03    STA   $03F3
95D3-   49 A5       EOR   #$A5
95D5-   8D F4 03    STA   $03F4
95D8-   A9 E5       LDA   #$E5
95DA-   85 6F       STA   $6F
95DC-   85 73       STA   $73
95DE-   A9 95       LDA   #$95
95E0-   85 70       STA   $70
95E2-   85 74       STA   $74
95E4-   60          RTS
95E5-   00          BRK
95E6-   00          BRK
95E7-   A6 76       LDX   $76
95E9-   E8          INX
95EA-   F0 11       BEQ   $95FD
95EC-   A6 33       LDX   $33
95EE-   E0 DD       CPX   #$DD
95F0-   F0 0B       BEQ   $95FD
95F2-   A6 D8       LDX   $D8
95F4-   10 07       BPL   $95FD
95F6-   A2 FF       LDX   #$FF
95F8-   86 DE       STX   $DE
95FA-   4C 12 D4    JMP   $D412
95FD-   6C E5 95    JMP   ($95E5)
*BSAVE RESETFIX.OBJ0,A$95BD,L$43.
9600-   00          BRK
9601-   00          BRK
9602-   00          BRK
9603-   00          BRK
9604-   00          BRK
9605-   00          BRK
9606-   00          BRK
9607-   00          BRK
9608-   00          BRK
*
```

Magic graphics commands

I RECEIVED an Apple for Christmas, and, in order to improve my fledgling programming skills, have decided to try to write my own game.

As there are two hi-resolution graphics pages, I would like to draw on one page while displaying another and then swap pages to achieve some degree of animation.

Using the HGR or HGR2 commands clears the screen, so my graphics attempts to date have been still-born.

I understand that there are some magic commands which I can use to achieve the desired effect.

Although I have an Apple IIe, I have access to the old Apple II reference manuals, the appropriate literature for the IIe being on order from my dealer.

When I have written my game, I shall send it to you for publication. — **Andrew Barnes, Purbrook.**

● It is possible to plot data on one hi-res screen while displaying the other so that the user doesn't see the screen being changed.

The POKEs listed on pages 132 and 133 of the old Applesoft Reference Manual control which screen is displayed and location 230

(decimal) controls which screen will be plotted on by the hi-res commands.

POKE 230,32 will cause Applesoft to draw on screen 1 (HGR) and POKE 230,64 will draw to screen 2 (HGR2).

To draw on Page 1 and display Page 2 hi-res:

POKE 230, 32
POKE −16299, 0

To draw on Page 2 and display Page 1 hi-res:

POKE 230, 64
POKE −16300, 0

Good luck. We are looking forward to seeing your completed games program.

Database for vets

I HAVE an Apple II, 48k, with two disc drives and an Epson MX 100 printer.

My experience is limited to producing my own crude programs.

I wish to have a suitable database which would handle a list of procedures and products used in veterinary practice.

About 200 would be the limit, constantly being changed, updated and re-priced. Also filing for approximately 200 farmer clients, much more static.

From this database I wish to print a price list in alphabetical order and also access the data so that it can be used in my own programs. Any ideas? – **Donald Edwards, Trowbridge.**

● Everyone's needs for organising information are different. They range from carefully tabulated names and addresses to free format sentences and paragraphs.

Several good database programs are advertised regularly in *Apple User*. Ask your dealer to recommend one – and to show it to you before you commit yourself to a purchase, to make sure it fits your needs – or else contact the distributors direct for further information.

Your local user group should be able to give you good advice.

Short of space

THANK you for publishing my Appletip in the February 1984 Apple User, but there is an unfortunate error in line 9020.

You will see from my letter and from the printout I enclosed that there should be a space between the '' and S of SAVE, line 9020.

The reason for this is that when the program ends and prints the SAVE statement on line 14, the POKE37,12 not only shifts the cursor up to line 14 but also the Applesoft square bracket, which leaves the cursor on the second column of that line, which without the space after the '' puts it over the A of SAVE, the S deleted by the bracket.

I hope most operators will see the error and put it right, but the standard regarding errors is generally very good in your magazine. – **Frank H. Mallett, Shoreham, West Sussex.**

FOR–NEXT problem

YOUR reply to R.A. Whittle in the December 1983 Windfall regarding FOR – NEXT loop problems was of great interest to me.

My Apple II+ had a nasty habit of seizing up after running one of my data handling programs.

Is there no other way to solve the problem except by the method you describe? Some of the routines I used have to be extensively modified to solve this problem.

Regarding your reply to Patrick Massey in the January 1984 Apple User, the Apple II Reference Manual, Page 10, gives details for Eurapple 50Hz modifications.

One should also note that a different crystal is required, and that warranty is voided if you change the crystal yourself.

Similarly, an 80 column card bought in the States requires changes, one being in ROM.

Mind you, the only inconvenience with the ROM problem is the vertical hold being different between 40 and 80 columns, though some monitors may handle it. – **J. Buchanan, St. Peter Port, Guernsey.**

● With regard to your siezed-up Apple, I am not aware of any more elegant solution to the FOR and NEXT problem than those previously outlined. The solution to your problem may lie in a close inspection of how you are using sub-routines. Similar symptoms occur if a sub-routine is not cleared with a RETURN statement. If you are forced to leave a sub-routine at an unexpected point, or wish to return to an unusual area of the programme, make use of the POP command.

Unreliable cassette

I HAVE just purchased a second-hand Apple II. As the original owner has upgraded his system to a IIe, he has retained the disc drives so until I can accrue the necessary cash I am forced to use the cassette

I HAVE recently bought a second hand Apple II and disc drive and for the second month have read Apple User with interest, especially the games reviews.

But could you please advise where I can purchase some of the games you review, for example "Dawn Patrol".

It sounds brilliant but you don't say from where it can be bought.

I hope that you can oblige with some outlets. – **K.P. McKeever, London.**

HAVING purchased an Apple IIe for business purposes, which I might add has been very

Missing on patrol

successful, to amuse visitors I wanted to purchase one or two games programmes.

As an avid reader of Windfall, now Apple User, I came across a review of Dawn Patrol by TSR.

In the light of my experiences since, I must say your reviews would appear to be an utter waste of time unless you publish sufficient information to enable the item to be purchased.

I have spoken to six Apple dealers, two of whom have a reasonable selection of games. Not one of them had heard of TSR, or had the faintest idea where I might purchase the programme. – **K.B. Bromige, Wraysbury, Staines.**

● You don't say whether you liked the actual reviews.

We hope you do, as they have been extremely popular over the years.

It is unusual, in our experience, to have so much trouble in locating a product.

Dawn Patrol is obtainable from TSR UK Limited, The Mill, Rathmore Road, Cambridge. Tel: 0223 212517.

storage system.

It doesn't, however, prove to be particularly reliable, especially in reading data from within Applesoft programs. I will be grateful if you have any tips for improving its performance. – **Phil Reynolds, Dyffed.**

● The choice of tape recorder is quite important to ensure integrity of data. In particular, recorders having an automatic volume control system should be avoided if they are not specifically designed to be used with computers.

Our experience has been that recorders at the cheaper end of the market are most likely to be successful.

There are several possible reasons for a cassette tape load to fail. The following list is presented in the order of the most probable cause.

☐ Try adjusting the volume and tone controls on the recorder.
☐ Try moving the recorder at least 18 inches away from the TV.
☐ Try another cassette to check if yours is defective.
☐ Try another recorder to see if yours is compatible and aligned.
☐ Try another Apple to see if yours is OK.
☐ Try cleaning the recorder head.
☐ Try unplugging the cable to the MIC jack.

Microsoft packages ready for Macintosh

FOUR applications packages for Macintosh are being released by Microsoft. They include enhanced versions of Multiplan and Microsoft Word, and two new applications – Microsoft File and Microsoft Chart.

Microsoft will also be making its Basic available for the Macintosh.

They chose Apple's launch of its new computer as the time to release two new applications programs onto the market.

These are a database management program, Microsoft File, and a graphics program, Microsoft Chart.

Multiplan on the Macintosh provides features of other versions of Multiplan with additional enhancements.

An UNDO command allows reversal of the last change to the spreadsheet.

Recalculation of data is claimed to be much faster, and printing options have been extended to include headers and footers and automatic page numbering.

Multiplan also employs Macintosh's "cut and paste" concept, allowing data to be transferred between different spreadsheets.

Microsoft Word uses the full graphics capabilities of the Macintosh screen to allow full visual representation of text and graphics on the screen.

This includes proportional spacing and support for Macintosh's character fonts.

Moving and copying portions of text, including moving between documents, is accomplished with the standard Macintosh edit functions.

Microsoft Chart is a business graphics program that can display information in graphical form.

Chart can format graphs in a number of formats – pie charts, line graphs, bar graphs, scatter diagrams and histograms.

Any chart can be reformatted into a different format.

Microsoft File is a database management program using forms to enter and view data.

It can then be used to supply data to any application that can read and write text files, or to Macintosh's clipboard.

Microsoft Basic takes advantage of the addressing capability of the 68000 microprocessor employed by the Macintosh, and includes a decimal maths package with 14-digit precision, string variables, and string expressions of up to 32,767 characters each.

Microsoft Basic incorporates the Macintosh interface and presents the user with up to three kinds of windows – one for command entry or editing a listing, one for viewing the program listing and one for the output of a running program.

Basic also provides many of the extended graphics capabilities of Microsoft's leading GW-Basic and has access to the Quickdraw routines supplied with Macintosh.

The Basic also supports Macintosh's font manager.

● *Microsoft Limited, Piper House, Hatch Lane, Windsor, Berkshire. Tel: 07535 59951.*

Image processing

HAVE you tried making a film with your Apple IIe using, as "actors", bit-map shapes that you build with simple keyboard commands?

A package that allows you to do this is available from Accent Software.

It is a graphics editor and animation system, called TGS, that combines graphics and text in a manner similar to making a movie.

Distributor Pete and Pam claims TGS can be an invaluable tool for developing training aids to film story boards and marketing surveys.

The program runs on a 64k Apple II with Applesoft in ROM and a single disc drive.

Its features include powerful editing capabilities, dual lo-res/hi-res operation, multiple backgrounds and hidden-line animation to show the proper overlay of objects when one object crosses another.

● *Pete & Pam Computers, New Hall Hey Road, Rawtenstall, Rossendale, Lancashire BB4 6JG. Tel: 0706 227011.*

Command performance

A HIGH resolution digital framestore which can be memory mapped into the Apple II for image processing applications has been developed by Foster and Freeman, a company specialising in the design and manufacture of micro based instrumentation for forensic science laboratories.

The basic framestore offers 768 pixel × 512 line resolution, of which a central 512 × 512 area is stored with 6 bit resolution.

The output to the monitor can be switched between the live and the stored image and a load button is provided to allow real time grabbing of the video input.

The 256k memory is bank selected by the Apple in 16 × 16k blocks. A user can READ and WRITE to the whole framestore image through machine code or Basic PEEK and POKE instructions. Normal operation of the Apple is not impeded.

The basic framestore is available in a case with cable connection to an Apple peripheral card. Additional slots can accommodate optional cards to provide a variety of extra features.

A 6502 processor card is available to run Apple developer programs faster in stand alone applications.

With 64k of RAM and room for 2 × 4k eproms, a number of user developed programs can be packaged.

Mass storage of images is provided by a 67mbyte removable cassette tape store controlled either through the Apple or the dedicated processor card.

The Apple itself may save parts of a framestore image on a standard floppy disc using BSAVE and BLOAD commands.

The basic framestore costs £3,000.

● Foster & Freeman, Tunnel Industrial Estate, London Road, Grays, Essex. Tel: 04026-7692 7714.

Styled for industry

A complete industrial-style Apple compatible computer based on the U-COM2 motherboard has been launched by U-Microcomputers.

The U-COM2 has been selling for more than a year as an OEM product.

U-Micros claims because it has written its own code for the U-COM2 ROMs copyright problems with Apple have not occurred.

The U-COM2 System One consists of a rectangular case with a 64k U-COM2 motherboard installed with illuminated

power switch, mains wiring and power supply.

The retail price is £349.

Options for U-COM2 include 40 or 80 column displays, disc drives, interface cards and monitors.

● *U-Microcomputers, Winstanley Industrial Estate, Long Lane, Warrington, Cheshire WA2 8PR. Tel: 0925 54117.*

Aid for accountants

A FULL suite of Apple IIe-based programs for the professional accountant has been developed by Southern Computer Software.

The suite, which is modular, consists of four programs – Incomplete Records, Time Ledger, Professional Services Invoicing and a Client's Ledger.

The Incomplete Records module has a rapid entries routine, allowing a free format of the chart of accounts and, in the final accounts print run, correctly sorts current assets and liabilities and produces a source and applications of funds report.

Coupled with Apple Multiplan, Quickfile and Applewriter, the system provides a package for the smaller practice.

At £1,250 the system can be installed on an Apple IIe for less than £3,250 with a dot matrix printer, or £3,750 with a daisy wheel printer.

● *Southern Computer Systems, 7 Park Hill Road, Torquay, Devon. Tel: 0803 212957.*

Plan range extended

AN addition to the Plan range of networking systems has been announced by Zynar.

The Plan 3000 runs on the same software as their existing 2000 and 4000 networks and forms an intermediary system on the path of network upgrading from the Plan 2000 entry system.

It uses 5¼in disc technology
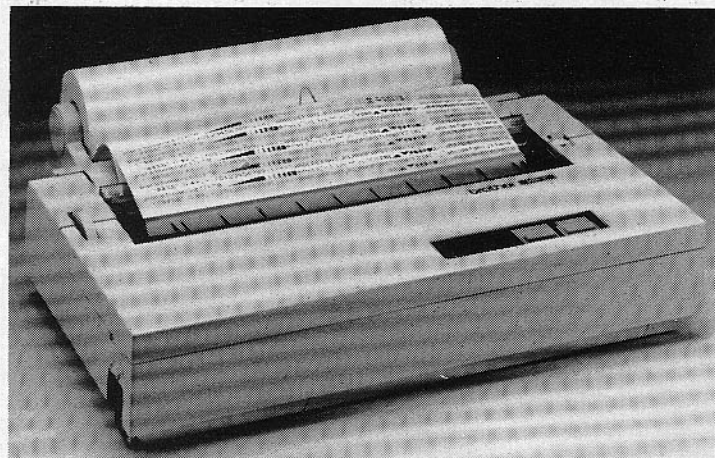
THE smallest printer offering from Brother, the HR5, is aimed at the home user and the OEM requiring a low cost, good quality print from a compact mains or battery operated printer.

The HR5 has either Centronics or RS232C serial interface and features 96 Ascii characters, 63 block graphic characters and 17 international characters as standard.

Print is produced by a 9 × 9 dot matrix and uses a transfer ribbon with a life of 50,000 characters at a speed of 30 cps.

Print pitch is 10 or 12

to provide a central memory store of 15mbytes and upwards.

A tape unit is included for system archiving and the software will support the IBM PC, Apple II and III as well as all their operating systems.

Rob Thorpe, Zynar's marketing manager said: "With a lower disc capacity than our 4000 system, the 3000 offers small and medium size businesses, an affordable solution to their networking needs.

"Smaller groups of users can benefit from sharing a common database and peripherals such as disc drives and printers".

Priced at under £8,500 for the 15mbyte capacity with a tape streamer for backup, the 3000 File Server can support a maximum of 255 workstations and servers per network.

Individual stations linked to the network may be placed up to 22,000 feet apart.

The 3000 is fully compatible with the existing Plan 2000 and 4000 systems and can utilise all

Younger Brother

characters per inch enlarged or condensed.

Text printing is unidirectional, while graph printing is done in a unidirectional mode at a noise level of less than 55 dB.

The HR5 weighs 3½ lb and is powered by standard flashlight batteries or mains power. Price: £14.

● Thame Systems, Thame Park Road, Thame, Oxon OX9 3XD. Tel: 084-421 5471.

products in the Series, including the Printer Server, File Transfer Server, and the Messenger electronic mail.

● *Zynar, 122 High Street, Uxbridge, Middlesex UB8 1JT. Tel: 0895 5983.*

1-2-3 for Macintosh

ANOTHER major company backing Macintosh is the Lotus Development Corporation, which is marketing a custom version of its 1-2-3 integrated software program for the machine.

"The new version of 1-2-3 is designed to take advantage of Macintosh features such as icons, and the cut-and-paste facility", according to Mitchell Kapor, president of Lotus.

"We are making a major commitment to Macintosh, which is natural, intuitive and in

line with how people think and work", Kapor said. "Macintosh sets a whole new standard, and we want our products to take advantage of it".

1-2-3 combines spreadsheet analysis, graphics, and information management functions in one program, which includes a high-level language capability, called the Macro Command language, which lets users tailor applications and store them to run with 1-2-3.

The 1-2-3 program was named product of the year by US magazines Fortune and Infoworld, and since its introduction in the UK last September has received high acclaim.

1-2-3's spreadsheet has 2,048 rows of 256 columns.

Sweet-P plotter

THE six pen Sweet-P Model 600 Six-Shooter graphics plotter from Enter Computers in San Diego will soon be available in the UK and Europe. It can be used with Apples and the IBM PC.

Standard features include 19 language character sets, 2k of buffer memory storage, a plotting speed of 14 inches per second and both serial and parallel interfaces.

The plotter electronically senses the interface to which it is connected and automatically switches to the port in use.

It is also compatible with HPGL (Hewlett-Packard Graphics Language), so most graphics software written for the HP 7400 series of plotters can also be used.

The Sweet-P 600 automates multipen usage in plotter graphics, and six pens are housed in the plotter's rotating carousel.

The pens can be purchased in any of 12 colours and can write on paper or acetate.

The system can also use radiograph-type drafting pens for CAD applications.

The plotter is expected to retail at around £850.

● *Pete and Pam Computers, New Hall Hey Road, Rossendale, Lancashire BB4 6JG.*