TRS-80 SYSTEM 80 VIDEO GENIE PMC-80 HITACHI PEACH TRS-80 COLOUR COMPUTER

Vol. 3, Issue 6, May 1982

Press 'I' for instructions, 'ENTER' for game!
Geronimo!

FOR TRS-80 & SYSTEM 80

Also in this issue:

PROGRAMMING:

Writing Instructional Programs
Screen Dump of LC Characters from NEWDOS 2·1

REVIEWS:

Micro-80 Lower Case Modification for the System 80

SOFTWARE:

- Matrix Manipulation—Level II
- Jumbled Players—Level II
- Reset—Disk
- Magic Cube—Level II
- Income Tax Calculator—Colour
- 3-D Cube—Colour

MICRO-80

P.O. BOX 213, GOODWOOD, S.A. 5034. AUSTRALIA. TELEPHONE (08) 211 7244. PRICE: AUS. $2.50, N.Z. $4.00, U.K. £1.50

***** ABOUT MICRO-80 *****

EDITOR:	IAN VAGG
ASSOCIATE EDITORS:	
SOFTWARE :	CHARLIE BARTLETT
HARDWARE :	EDWIN PAAY

MICRO-80 is an international magazine devoted entirely to the Tandy TRS-80 microcomputer and the Dick Smith System 80/Video Genie. It is available at the following prices:

	12 MONTH SUB.	SINGLE COPY
MAGAZINE ONLY	$ 26-00	$ 2-50
CASSETTE PLUS MAGAZINE	$ 65-00	$ 4-00 (cass. only)
DISK PLUS MAGAZINE	$ 125-00	$ 10-00 (disk only)

MICRO-80 is available in the United Kingdom from:
U.K. SUBSCRIPTION DEPT. 24 Woodhill Park, Pembury, Tunbridge Wells, KENT. TN2 4NW

Prices:		
MAGAZINE ONLY	£ 16-00	£ 1-50
CASSETTE PLUS MAGAZINE	£ 43-60	N|A
DISK PLUS MAGAZINE	£ 75-00	N|A

MICRO-80 is available in New Zealand from:
MICRO PROCESSOR SERVICES, 940A Columbo Street, CHRISTCHURCH 1 N.Z. Ph. 62894

Prices:		
MAGAZINE ONLY	NZ$ 43-00	NZ$ 4-00
CASSETTE PLUS MAGAZINE	NZ$ 89-00	NZ$ 5-00
DISK PLUS MAGAZINE	NZ$ 175-00	NZ$ 15-00

MICRO-80 is despatched from Australia by airmail to other countries at the following rates:

(12 MONTH SUB.)	MAGAZINE	CASS + MAG	DISK + MAG
PAPUA NEW GUINEA	Aus$ 40-00	Aus$ 83-00	Aus$ 143-00
HONG KONG/SINGAPORE	Aus$ 44-00	Aus$ 88-00	Aus$ 148-00
INDIA/JAPAN	Aus$ 49-00	Aus$ 95-00	Aus$ 155-00
USA/MIDDLE EAST/CANADA	Aus$ 55-00	Aus$ 102-00	Aus$ 162-00

Special bulk purchase rates are also available to computer shops etc. Please use the form in this issue to order your copy or subscription.

The purpose of MICRO-80 is to publish software and other information to help you get the most from your TRS-80, System 80 or Video Genie and their peripherals. MICRO-80 is in no way connected with either the Tandy or Dick Smith organisations.

** WE WILL PAY YOU TO PUBLISH YOUR PROGRAMS **
Most of the information we publish is provided by our readers, to whom we pay royalties. An application form containing full details of how you can use your TRS-80 or System 80 to earn some extra income is included in every issue.

** CONTENT **
Each month we publish at least one applications program in Level I BASIC, one in Level II BASIC and one in DISK BASIC (or disk compatible Level II). We also publish Utility programs in Level II BASIC and Machine Language. At least every second issue has an article on hardware modifications or a constructional article for a useful peripheral. In addition, we run articles on programming techniques both in Assembly Language and BASIC and we print letters to the Editor and new product reviews.

** COPYRIGHT **
All the material published in this magazine is under copyright. That means that you must not copy it, except for your own use. This applies to photocopying the magazine itself or making copies of programs on tape or disk.

** LIABILITY **
The programs and other articles in MICRO-80 are published in good faith and we do our utmost to ensure that they function as described. However, no liability can be accepted for the failure of any program or other article to function satisfactorily or for any consequential damages arising from their use for any purpose whatsoever.

***** CONTENTS *****

	PAGE
EDITORIAL	2
PEEKING (UK) (From our U.K. Correspondent)	3
SOME PRINCIPLES AND GUIDELINES FOR WRITING INSTRUCTIONAL PROGRAMS ON MICROCOMPUTERS	4
MODEL III GETS THE RUN-AROUND AS A STATUS SYMBOL	9
USER GROUPS	11
DUNJON QUEST - TEMPLE OF APSHAI (REVIEW)	12
FS1 FLIGHT SIMULATOR (REVIEW)	13
MICRO-80 LOWERCASE MODIFICATION FOR THE SYSTEM 80	14
SM ELECTRONICS CONVERSION OF THE OLYMPIA ES100 ELECTRONIC DAISYWHEEL TYPEWRITER	16
INPUT/OUTPUT - LETTERS TO THE EDITOR	16

SOFTWARE SECTION

AUSTRALIAN INCOME TAX CALCULATOR	CC	19 & 24
AUSTRALIAN INCOME TAX CALCULATOR	PEACH	19 & 25
3-D CUBE	CC	20 & 25
3-D CUBE	PEACH	20 & 26
SKYDIVER	L2/16K	20 & 26
JUMBLED PLAYERS	L2/16K	21 & 31
MATRIX MANIPULATION	L2/16K	21 & 33
RESET	32K/Disk	22 & 34
MAGIC CUBE SOLVER	L2/16K	23 & 29

NEXT MONTH'S ISSUE	35
CASSETTE/DISK EDITION INDEX	36
ORDER FORM	36

MICRO-80 is registered by Australia Post - Publication SQB 2207 Category B

AUSTRALIAN OFFICE AND EDITOR: MICRO-80 P.O. BOX 213, GOODWOOD, SOUTH AUSTRALIA, 5034. TEL. (08) 211 7244

U.K. SUBSCRIPTION DEPT: 24 WOODHILL PARK, PEMBURY TUNBRIDGE WELLS, KENT TN2 4NW

Printed by: Shovel & Bull Printers, 379 South Road, MILE END 5031

Published in Australia by: MICRO-80, 433 Morphett Street, Adelaide.

***** SPECIAL OFFER TO NEW READERS AND READERS RENEWING THEIR SUBSCRIPTION *****
***** SOFTWARE LIBRARY, VALUED AT OVER $100 – FREE!!! *****

MICRO-80 has developed a new Library of Software consisting of 7 programs and a comprehensive user manual. The Software Library, on cassette, will be sent FREE to every new subscriber and to every subscriber who renews his subscription for another 12 months. Disk subscribers will receive their Software Library on a diskette. The new Software Library contains the following Level II/Disk Programs. All programs will also operate on the Model III.

Level I in Level II
Convert your Level II TRS-80 or System 80 to operate as a Level I machine. Opens a whole new library of software for your use.

Copier
Copies Level II System tapes, irrespective of where they load in memory. Copes with multiple ORG programs.

Z80 MON
A low memory, machine language monitor which enables you to set break points, edit memory, punch system tapes, etc...

Cube
An ingenious representation of the popular Rubick's cube game for Disk users.

Poker
Play poker against your computer, complete with realistic graphics.

Improved Household Accounts
Version 3.0 of this useful program. One or two bugs removed and easier data entry. This program is powerful enough to be used by a small business.

80 Composer
A music-generating program which enables you to play music via your cassette recorder and to save the music data to tape. This is an improved version of the program published in Issue 17 of Micro-80.

***** EDITORIAL *****

There is some fascinating news for Tandy users which has just come to hand. In the USA, Radio Shack, which has so staunchly refused to allow any but its own stores and Authorised Distributors to sell TRS-80 computers, has made a major change in marketting policy. Henceforth, The TRS-80 Color Computer will be sold to 60 distributors supplying over 2000 independent retail outlets. This move is of great significance since it is a tacit admission that the Colour Computer at least, is very different from Radio Shack's other products. Is this the thin end of the wedge? We await with great interest news of Tandy UK's and Tandy Australia's reactions to the new policy.

As if the above is not sufficient, Radio Shack has also announced a 5 Mbyte hard disk drive for the Model 3. The chosen Operating System is LDOS. So, for the first time, Radio Shack is recognising the contribution which can be made by independent organisations. We heartily applaud this move and believe that more co-operation between Tandy and many independents supporting Tandy products can only strengthen the position of the TRS-80 in a fiercely competitive market place.

The Model 16 microcomputer is due for release in Australia in November 1982. This computer has two microprocessors, a Z80A and a 68000. The Z80 means that it will be compatible with all existing Model 2 software, while the 68000 gives a full 16 bit capability. The standard machine also has two double-sided, double-density, 8 inch disk drives mounted in the cabinet for a total disk storage capacity in excess of 2 Mbyte. All this will sell for about $7000. The Model 16 will be a formidable contender in the small business computing field. It is pleasing to see a major manufacturer remaining with 8 inch disk drives. These drives are undoubtedly more reliable than the ultra high density 5 inch drives being used in so many new computers. The problem is not so much with the drives as with the media which is being stretched to its limit in 80 track dual-headed drives operating in double-density. No doubt improved media will eventually become available but, for the present, 1 Mbyte 8 inch drives are significantly more reliable for data storage than 3/4 Mbyte 5 inch drives.

Those who went to the Data '82 exhibition in Sydney recently, could have seen the new Hitachi 16 bit computer. this is Hitachi's IBM look-alike and will operate under both MSDOS (the disk operating system developed by MICROSOFT for the IBM) and CP/M-86. The processor used in the Hitachi is the Intel 8088, just as in the IBM. This is to all intents and purposes a 16 bit processor but with an 8 bit data bus. Rumours still abound as to the release date in Australia of the IBM PC (Personal Computer) and it is quite likely that the Hitachi will be available here before the IBM since deliveries are expected before the end of 1982. Despite statements in some other magazines, the price of the Hitachi 16 bit computer has not yet been fixed. We believe that, with Sales Tax paid, it will be somewhere in the region of $5500, including a high resolution colour monitor. We will keep you posted as more news comes available.

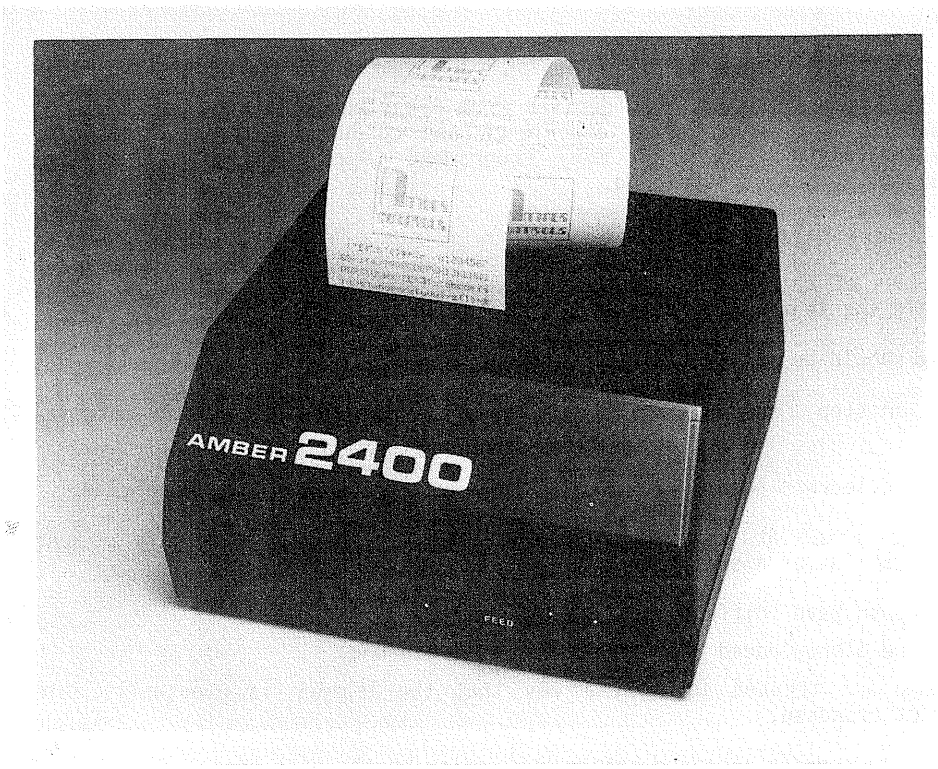
Seasoned readers of MICRO-80 will know that most of the programs we publish are written by our readers. This makes for a great variety of programs and helps readers make their hobby pay. Whilst waiting for readers to send in programs for the Peach and the Colour Computer, Editorial staff are converting programs previously published for the '80 computers. This month we came unstuck on one of the programs - Measurements. This program has so much graphics content that it would need to be completely rewritten to operate satisfactorily on the colour computers with their very much higher resolution than the '80. Perhaps some enterprising reader will take on this task which we have now abandoned. We would, of course, pay a publication fee. In the meantime, we have written a simple program which makes use of the high resolution and line-drawing capabilities of the colour computers. In converting The Australian Income Tax Calculator, we have taken the opportunity to bring it up to date with the latest rates and the method of calculating rebates. Whilst on the subject of programs in this issue, '80 readers get an unadvertised bonus in the shape of the program Matrix Manipulation. Such a program has been on our list of requests for a very considerable time.

- 0000000000 -

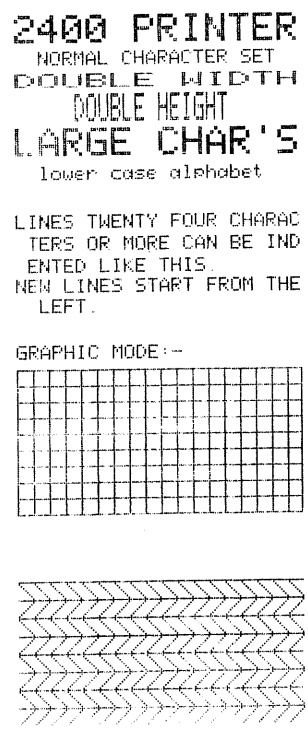
***** PEEKing (UK) by our U.K. Correspondent Tony Edwards *****

The prices of micro-computers in the UK are falling so quickly that it is bewildering. Clive Sinclair led the way with his 16K colour computer the 'Specrum' at £125 ($A250) (it is one of the ZX series). Following this the cost of colour computers with high resolution graphics has plummeted. Texas Instruments and Atari have both reduced their prices by some £100 ($A200) (30%) as the Atari 400 and the Texas TI99/4A both sell in the UK for less than £200 ($A400). A newcomer, the Dragon, also sells for about this figure. The new Commodore machine, the VIC-20, now sells for only £125 ($A250) and this will soon fall it is rumoured (perhaps before you read this even). As if these reductions were not enough I now hear that Binatone will be introducing a new machine in January 1983. It will be a 16K machine (expandable to 64K) with colour and sound and its projected price is less than £50!! ($A100). Remember the days when (early this year) a ZX-80 at about £80 ($A160) was considered cheap for a 1K black and white machine. Binatone's machine will be marketed through major High Street chain stores such as Woolworths and Argos as well as by mail order and is expected to sell 400,000 units in 1983. It is of interest to us at MICRO-80 as this Binatone Unit uses Tandy TRS-80 software.

Not only have computers fallen in price at an accelerating pace but so have peripherals, especially printers. Many are now available at under £200 ($A400). However I have just heard of the next jump, a printer with both series and parallel interfaces as standard selling at £70 ($A140). It is the Amber 2400. It has 24 characters/line with lower case and dot graphics. It uses cheap plain roll paper 160mm wide and has a selectable baud rate between 75 and 9600. It is available at such a low price because it uses a novel needle array. Most dot matrix printers form characters with a vertically aligned array of print needles which traverse the width of the paper on a carriage. In the Amber 2400 there are only four needles aligned horizontally which oscillate from side to side, each covering a quarter of the paper width. The paper is incremented one dot height as the needle returns. This printer will never be used in word processing but it is a very good printer at a rock bottom price for many users who do not need the full width output. Below is an example of the print output from this machine showing its various modes. The printer controls are built into its hardware so there is no difficulty in handling the unusual print head orientation, and it is available for most micros including the '80 range.



PHOTOGRAPH OF AMBER PRINTER



SAMPLE OF OUTPUT
(App. 2/3 actual size)

- 0000000000 -

******* MASTER DISK DIRECTORY - $19.95 + $1.00 p&p *******
FIND THAT PROGRAM FAST!! PAYS FOR ITSELF BY RELEASING REDUNDANT DISK SPACE!!

MASTER DIRECTORY records the directories of all your individual disks onto one directory disk. Then it allows you to examine them, find an individual file quickly, list files alphabetically, weed out redundant files, identify disks with free space, list files by extension etc. etc.. This program is invaluable for the serious disk user and will pay for itself many times over.

******* THE FLOPPY DOCTOR/MEMORY DIAGNOSTIC - NOW AVAILABLE FOR THE MODEL 3 TOO! *******
Model 1 Disk $35.50 + $1.00 p&p. Model 3 Disk $42.50 + $1.00 p&p.

Computer professionals have long known the importance of regular use of diagnostic software in verifying the integrity of computer hardware. The TRS-80 is no exception; good diagnostics are a must in any situation where valuable data files are maintained. The new double-density recording techniques available for the Model I and used in the Model 3 together with high track count and double-sided disk drives stretch the hardware to its limits and make it even more important than ever to thoroughly check out the system prior to trusting your valuable data to it.

THE MICRO CLINIC offers two programs designed to thoroughly check out the two most trouble-prone sections of the TRS-80 - the disk system (controller and drives) and the memory arrays. Both programs are written in Z80 machine code and are supplied together on diskette for a minimum 32K, one disk system. Specify Model 1 or Model 3.

***** SOME PRINCIPLES AND GUIDELINES FOR WRITING INSTRUCTIONAL PROGRAMS ON MICROCOMPUTERS - by Dr. John F. Barrett *****

The task of writing good instructional programs is both simple and complex. It is simple in that our goal of having someone learn "something" while at the computer keyboard is obvious. It is complex in that one has to know a great deal about the psychological processes of learning and also know how to write programs that follow established learning principles.

In this article I will introduce some principles for developing instructional computer-based programs, knowing full well that various "learning theorists" would be critical of the simplicity of the presentation.

To begin, let me endorse the Editorial of MICRO-80, issue No. 3 of February 1980. Generally, "teaching programs should be interesting" as well as building the confidence and self-esteem of the user. As also mentioned, the satisfactory treatment of user errors is a very important aspect of any form of instruction. I hope this presentation will convey the idea that the computer is capable of providing a more complex instructional role than telling how many one got correct out of 10 attempts!

Theory

In this brief introduction I will keep to principles derived from Behavioural Theory, particularly the work of B.F. Skinner who has had considerable experience and conducted extensive research in the area of behaviour modification from which many Programmed Learning materials and CAI (Computer-Assisted Instruction) packages have developed.

Principles of Learning

1. The goals or objectives of the program are clearly defined and all experiences or steps in a program are to aid the learning in achieving these goals.

2. Material to be learned should be in small steps of a number of frames of information.

3. Following each frame or step a number of questions which assess learning of the material should be presented.

4. Related to 2,3 above the learner should be actively involved in the learning experience.

5. Material should be presented at a level appropriate to the learner's ability - such that he/she can get about 70% of answers correct.

6. Responses (answers) should have "differential treatment";

   (i) Correct answers are acknowledged and are reinforced (rewarded)

   (ii) Incorrect answers are treated in such a way that the learner is eventually able to get the correct response.

7. The learner has to get the correct response before proceeding to the next step.

8. Related to 4,5, 6 above - The user is provided with immediate knowledge of results - a very important principle most often ignored in classroom teaching and a major strength of CAI - when so designed!

9. "Tests" for competency (knowledge or problem solving) may be used.
   Basically they take three forms:

   (i) As a "Pre-Test" - before a new sequence of instructions to gauge the entry status of the learner - i.e. does the learner have significant "knowledge" to begin the program?

   (ii) As a "Post-Test" - for assessing performance levels after instruction (i.e. the usual "TEST" procedures).

   (iii) For Diagnosis where errors are analyzed and instruction is provided for remedial purposes.

   Please note that testing "in and of itself" is not predominantly instructive. It may provide useful information which allows learning but this has to be built into the learning system.

10. After the learner has attained the correct response and is rewarded (not every time!) then further practice is given.

    Such practice enables "overlearning" the correct response. However the practice exercises are at a similar level of difficulty and only employ previouslylearned material and skills. Contrast this with many lessons at school -

After the teacher has shown the correct procedure on the chalk board, (how actively involved is each pupil?) often demonstrating a number of complex steps; he/she now says -

"Now do Nos. 3, 5, 7, 9 of the graded exercises!"

These exercises turn out to be performance tests not practice!

11. Continuous reinforcement (reward statement such as "Correct" or "Well Done!") are effective only for a limited time in the initial phases of new instruction. After this a ratio of reinforcement (i.e. reward statements of one in say 5 correct) followed in a later phase by Random Reinforcement (where rewards are randomly distributed to correct answers)will prove to be more effective.

12. Enable the learner to move at his/her own pace. This has two aspects -

(i) Quickness of response - the computer is infinitely patient with slow movers and can keep ahead of the best!

(ii) Those doing well (you define this in your program) may "jump" sections of the work to get to the goals of the program more quickly. In this case, you have changed from a Linear Program (like reading the pages of a book) to Branching Programs. It is obviously imperative that you know where such branching will finish up!

Guidelines for Developing CAL

How can we use these principles for development of instructional programs?

First, keep it simple!

Until you have a 'feel' for the process it is better to select an area with which you are quite familiar. This is particularly important as you are trying to accomplish skill in 3 aspects:

- remembering the aim and objectives of your program - what exactly do you wish your learner to be able to do (or know) as a result of your program?

- keep track of the flow of ideas and the sequence of the learning tasks (this applies especially to the treatment of incorrect answers).

- have programming skills (and the appropriate hardware) to accomplish what you intend.

Programming and Flowcharting

Let's examine a block diagram of what a simple program may contain (see Fig. 1).

In this first stage an introduction must be given. Note that the instructional step should be kept short and written in an accurate and free-flowing style using words appropriate to the reading level of users. Of course, presentation on the screen as to tabulating and spacing is important but these may be edited at a later date.

Answers are treated as correct or incorrect. Incorrect answers simply repeat the instruction/assessment loop. (Note the infinite loop - we will have to do something about that!) Correct answers are informed that this is so, and are rewarded in some way. Rewards are of two types: intrinsic - just getting the correct answer is rewarding and this can be consolidated by giving a running total of correct answers; specific, such as "bells and whistles", exciting graphics etc. These are especially effective for younger children but be careful not to overdo it and detract from the original objective of your program. Note that the learner is not chastised in any way - he is told, say,

"No, let's try again!" As you see, this treatment has a built in fault. Our learner has now "learned" the correct response possibly without knowing why "it is so"!

If you continue to develop your program following this sequence you will have the framework of a good Linear program. Even though it is a very simple structure you have the basis for building a more complex program - one more suited to computer applications.

STAGE II

You may use refinements such as those suggested in Figure 2, to further incorporate some of the previous mentioned principles.

Explanation In this stage a frame is presented and three possible answers are presented. The correct answer follows the previous example but in this case a "NUMBER CORRECT" counter is initiated.

Incorrect answers are differentially treated. A simple mistake gains a second attempt, however we have got rid of the infinite loop by allowing a second attempt only. Third attempts and "serious mistakes" are given remedial instruction. That is, they are shown how the mistake may have occurred and how it can be corrected.

INSTRUCTIONAL ASPECT

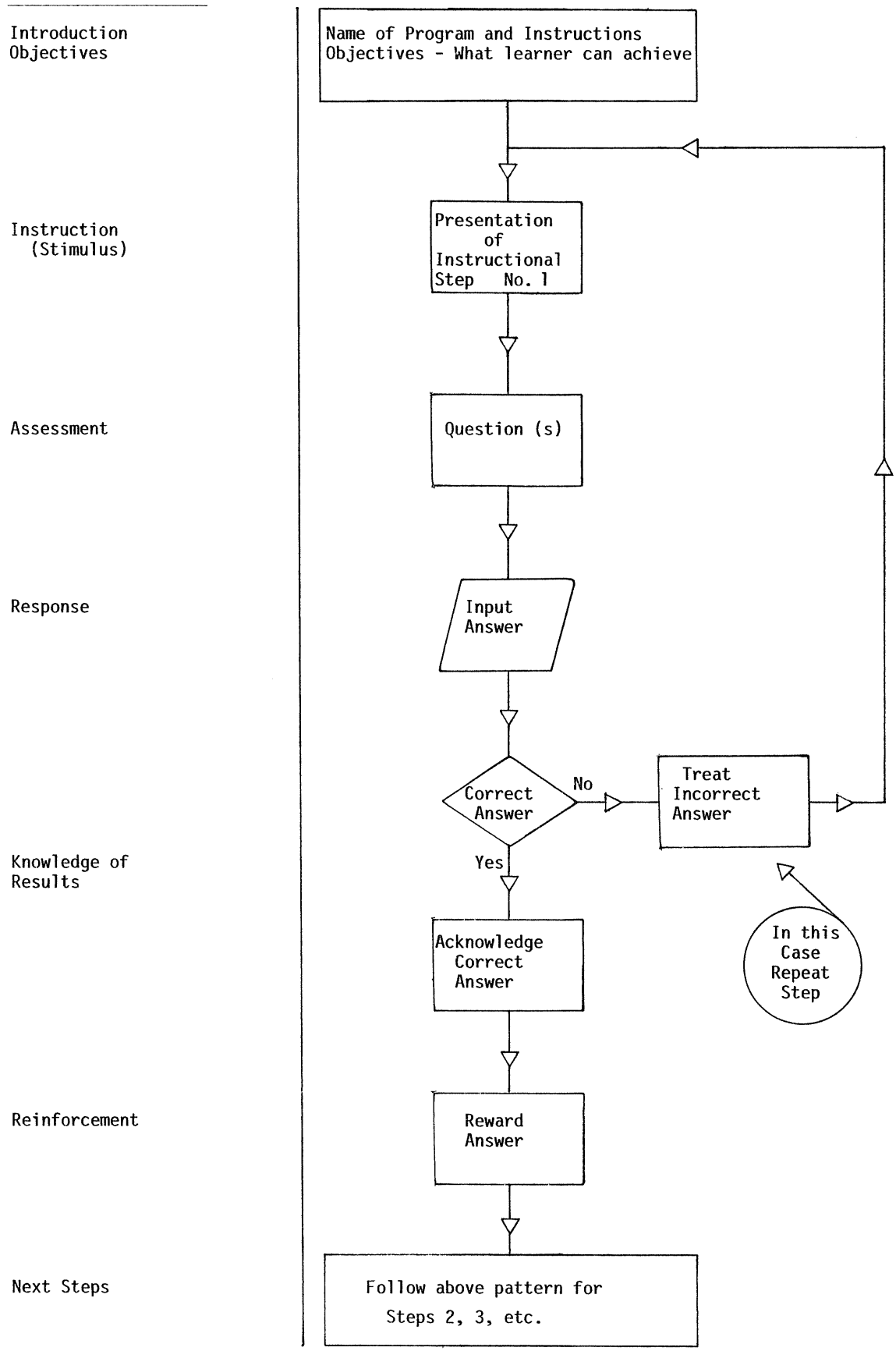


Fig. 1 ASPECTS OF SIMPLE CAI PROGRAM

The Remedial Instruction block is shown as a sub-routine. They may well be referred to later as part of the program where similar mistakes are made. Also, consider the idea of storing these for remedial work as a result of performance on a final TEST (i.e. the "POST-TEST").

In this case after remedial instruction the learner could have been re-routed to Step 1. Again the loop would have to be considered.

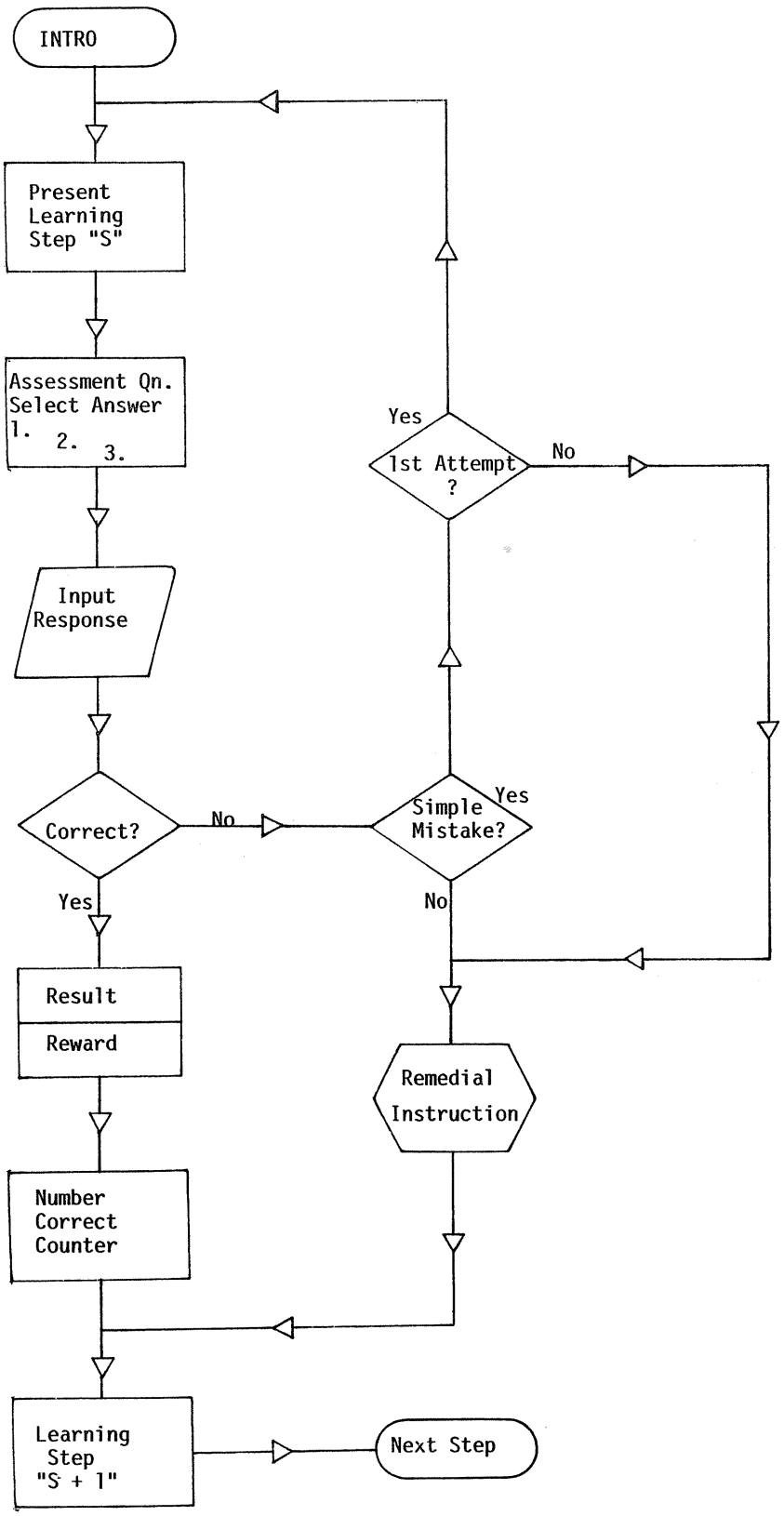


Fig 2. - STAGE TWO

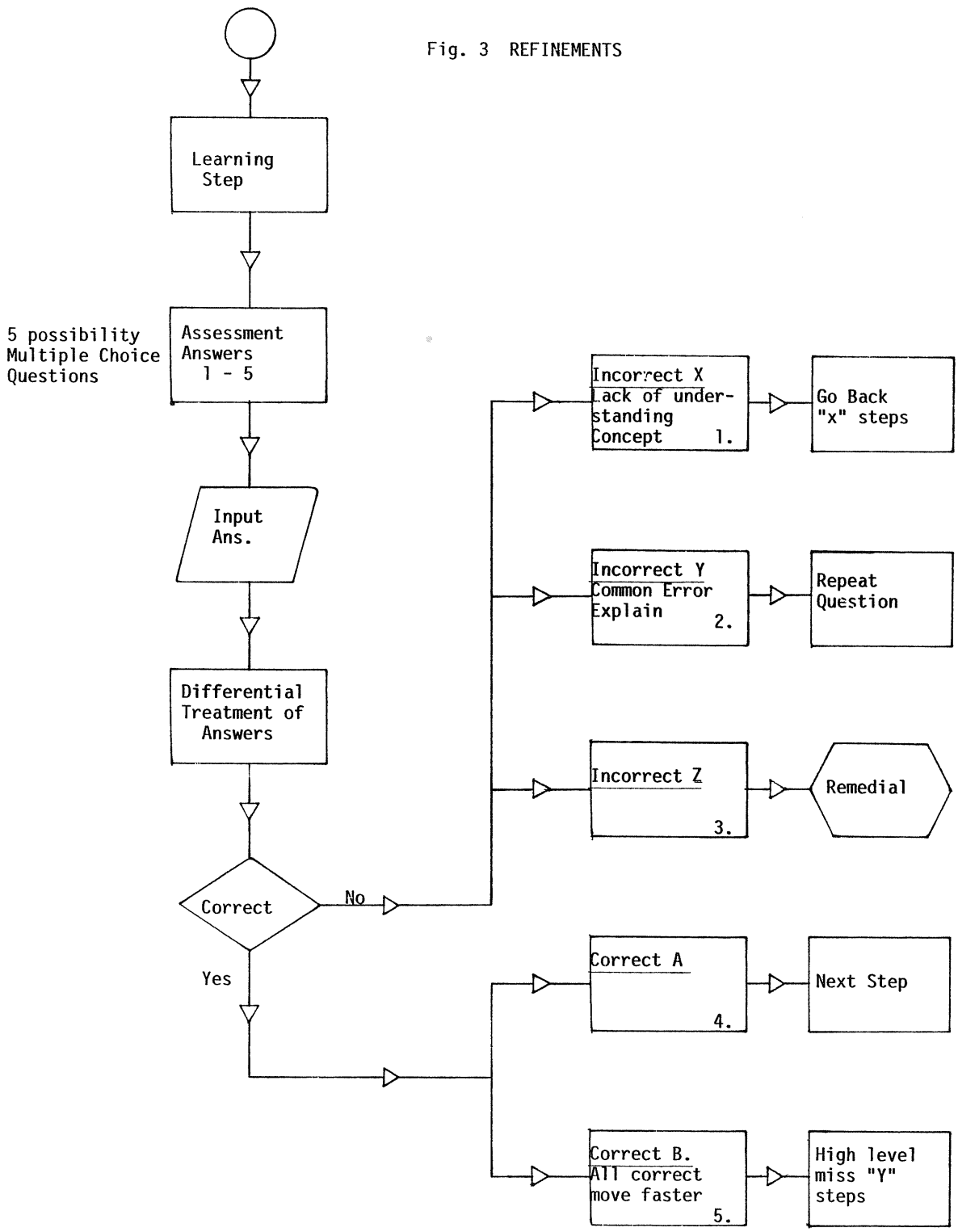
Further Refinements

The Block Diagram (Fig. 3) suggests further development.

After the learning frame an assessment question in the form of a 5 answer "multiple-choice" is presented.

Each answer is treated differently. The learner who gets this correct and has all steps correct so far (or whatever you define) can jump a number of steps to the next concept. Correct A may be given more practice to consolidate previous learning.

Fig. 3 REFINEMENTS



Of the incorrect, three treatments are applied. Demonstrating a serious misconception (Incorrect Z) sends the learner to a remedial sub-routine. (After that, it's up to your programming!) Incorrect X has missed the main idea or concept and goes back a few steps, while Incorrect Y made a "silly mistake" and repeats the question.

This procedure should 'trigger' a large number of possibilities - it will be important to "keep track" of where all possibilities lead (especially "infinite loops") and keeping in mind the instructional purpose of the program.

Content of Steps

In this presentation I am concentrating on simple CAI (Computer Assisted Instruction). Note that these are NOT GAMES OR SIMULATIONS (more on these aspects in a later article!)

Each step or frame referred to above can contain various types of information. It is probably best to concentrate on teaching knowledge or facts in your first attempt at programming CAI. However do not present long pages of material in which one only pushes a button for each page (books can do the same thing!)

Each step then can encourage

1. Knowledge of facts
2. Knowledge of ideas
3. Recall of facts, ideas (remembered from previous materials)
4. Recognition - visual prompt to encourage response
5. Discriminations - being able to tell difference between A & B, or A B C etc.
6. Knowledge of concepts and principles
7. Methods of solving problems.

The procedures outlined in this article are probably more suited to aspects 1-5 above. Areas 6 and 7 require a different approach - and another article!

- 0000000000 -

***** MODEL III GETS THE RUN-AROUND AS A STATUS SYMBOL by S.H. Liggins *****

In August 1981, the Tasmanian Little Athletics Association asked me if I would supply a computer system for the Australian Championships to be held in Hobart on Easter Sunday in 1982.

Their requirements are:

1. Up-to-date State Points to be available for every event.
2. Production of very well-presented results to be copied by Gestetner and bound into 300 booklets within one hour of the running of the last event.
3. Random allocation of lanes (to save a lot of mucking around with marbles).
4. Anything else I can think of.

It sounds a nice change from accounting so I agree, subject to certain conditions:-

1. Someone else does the operating - I fumble on a key-board under pressure.
2. A backup machine is provided (Model III, 48k, and printer).
3. A good team of people is available to do it all manually if I, the computer, the programmes or the power fail.

I find the operator myself. Louise, who works with me, is ideal. She is a Little Athletic Parent, an experienced computer operator, can do 100 things at once with 8 people talking to her, and is available for lunch-time design discussions, practice sessions, and preliminary data entry.

We are to have two 'dry runs' - the two days of the Tasmanian Championships, held on successive weekends in March. Needless to say I am still modifying programmes the night before the first trial.

Day One goes fairly well. Some of the listings are rather peculiar. (As most of the development work is done between 10pm and 2am while the rest of the household sleeps, and my printer sounds like a lawn-mower, I haven't done as much testing of the printing as I should have). The State

Points are completely wrong. I do a quick fix, and improve them, but don't solve the whole problem.

I spend the week polishing, and Trial Day Two arrives. Louise can't come, but a kind, calm friend who owns a Model III offers to do the data entry. The listings are much better. The point-checkers watch us like hawks and we have three discrepancies. Two of these I prove to be THEIR fault, and I leave them to prove that the third is mine. They can't. The Officials play all sorts of dirty tricks on us - combining two events, altering the order of others, upholding protests. We cope with it all, and triumph by preventing the presentation of a medal to the wrong child. Times (or distances) are entered as well as placings, and the programme checks that they correspond.

Now we have three weeks for the final tidying up, and of course enhancements. By this time the organising committee members have realised some of the benefits of the computer and ring me daily with some new request, particularly for selective lists and summaries of events and competitors that can be used by the Marshals, Starters etc. Louise spends a day entering the event and competitor details. She then prints a list of events for herself and turns it into a chart which she will use to tick off events as results are entered, points allocated etc. She also marks tentatively where there should be time for backups, cups of coffee and visits to the toilet.

When our data entry lists have been checked by the officials, we print an 'Entry Sheet' for every competitor, showing the starting time for each event in which they are entered. These will be distributed via the Team Managers as they arrive in Tasmania, and should assist in checking for 'time clashes' as well as making the visiting officials aware of the computer's role in the proceedings.

The Day dawns, and I arrive at the track later than I had planned. I leave my computer in the 'control tower' and go to collect the borrowed one. I don't dare to have them both in the car at once.

All set up by 9:15, and the first event will start at 10.00, but before that we have to deal with a batch of 'substitutions' (a competitor entered in an event is replaced by another from the same team). This was a contingency we did not have in the trials. There is no substitution function in my programme, so we do it by withdrawing the original competitor, and entering the replacement - two separate operations. (The Mark II version will do it in one). Two Australian records have been broken during the last three weeks, so we amend them in the event descriptions. (The programme will highlight the breaking or equalling of records when the results are entered).
[10:15] The first results come in and away we go.

[11:00] The State Point Auditing ladies say we are one point wrong already. I remember a 'did-not-finish' in one event result which the computer coped with. They go away to check their tally again, and don't come back!

[11:15] The Tasmanian State Premier, who has been Officially-Opening-and-Medal-Presenting, comes to inspect us. He is pleased when he learns that Louise and I work at the T.A.B. "Keep up the good work there - the State needs the revenue".

[12:00] Louise does the first backup. Now the spare computer gets down to producing the first page of results to be "Gestetnered" for the results booklet. It looks so good that I get a request to produce a title page. I write a very simple programme on the spare computer, while Louise labours on.

[12:30] The announcer says a record has been broken. Louise notices that the computer does not highlight it. There is some frantic checking until everyone realises that it was one of the events that we had altered earlier. The announcer explains to the crowd that it wasn't a record after all, and apologises to the competitor.

And so the day proceeds. I learn later that the points-auditors disagreed with our totals several times, but always found it was their error. During the running of the last event, at 4:00, we are presented with a nasty challenge - an earlier event has been officially 'confirmed' as correct, and the state points allocated. Now it looks as though a re-count may alter the placings. For safety reasons the programme does not allow simple 'de-confirming' and 'un-allocated' but it can be done. I give a quote of 20 minutes, but hope it will take about five. I draw up a plan of action while the Officials consult the rule books about protests. They rule that the placings are to stand unaltered, and I breathe a sigh of relief and resolve that Mark II will deal with such situations smoothly.

We give the final points to the announcer, do a backup, and print the last two pages of results for the Gestetner. Then comes the fairly long job of printing a 'Performance Sheet' for each competitor, again to be distributed via the Team Managers. This shows the placings, times (or distances) and state points earned in each event that the child had entered.

Indefatigable Louise now performs for an ABC TV film crew which is making a series of films about Computers In Life and Leisure. They filmed several events during the afternoon, and now she re-enacts the entering of results (using one of the backup discs from earlier in the day).

As part of the film, the Little Athletics Technical requirements manager is interviewed:-

"Why did you decide to use a computer?"

"Well, Tasmainia lacks some facilities for holding a competition of this kind when compared with other States. We decided to try using a computer because no other State had done so. After the project was underway of course, we realised the benefits to be gained and it certainly helped us today. But originally I suppose it was really a sort of Status Symbol!"

- 0000000000 -

***** USERS' GROUPS *****

The following is a list of Users' groups of which we are aware. Many are interested in a wide variety of computers. For further information about any of the groups, contact the person indicated. If you have a group not mentioned here, please let us know so that it can be included in this section.

** AUSTRALIA **

AUSTRALIAN CAPITAL TERRITORY

CANBERRA GROUP

Cont: Bill Cushing,
10 Urambi Village, Kambah, 2902.
Meet: 3rd Thurs. monthly, 7.30pm.
Urambi Village Comm. Cent, Crozier Circuit, Kambah.

NEW SOUTH WALES

BLUE MOUNTAINS OF N.S.W.

Cont: Greg Baulman
Tel: Home (047) 51 3221
Meet: 1st Fri. monthly, 7.30pm,
Springwood Civic Cent.

COMPUTERTOWN CAMDEN

Cont: Keith Stewart
P.O. Box 47, Camden 2570.

TRS-80 SYDNEY EASTERN SUBURBS USERS GROUP
MAPPER CP/M USERS GROUP

Cont: Dan Lawrence
G.P.O. Box 2551, Sydney 2001.

WOLLONGONG GROUP

Cont: Paul Janson
P.O. Box 397, Dapto, 2530.

VICTORIA

BALLARAT COMPUTER USERS GROUP

Cont: John Preston, Tel (053) 31 4363
Meet: 2nd Tue. monthly, Chisholm College, Frankston

EASTERN SUBURBS USERS GROUP

Cont: John Fletcher
Tel: Home (03)737 9544
Bus (03)89 0677 (9-4)
Meet: 4th Wed monthly, 7.pm,
Kingswood College, 355 Station St.
Box Hill

GEELONG COMPUTER CLUB

Cont: P.O. Box 6, Geelong, 3220.
Meet: 2nd Thurs monthly, Tybar Eng.,
Hampton St. Newton

MICROCOMPUTER CLUB OF MELBOURNE

Cont: MICOM, P.O. BOX 60, Canterbury 3126
Meet: 3rd Sat. monthly 2p.m. Burwood
State College, Burwood Hwy.

NORTHERN & WESTERN SUBURBS COMPUTER USERS GROUP

Cont: David Coupe, Tel: (03)370 9590
Meet: CPM Data Systems, 284 Union Rd,
Moonee Ponds - Alt. Thurs. 7pm.

PENINSULA COMPUTER GROUP

Cont: George Thompson, 3 Patterson St.
Bonbeach, 3196. Tel. 772 2674.

Meet: 2nd Tues monthly, Chisholm College,
Frankston.

QUEENSLAND

BRISBANE GROUP

Cont: Lance Lawes,
Tel: Home (07)396 2998
Bus. (07)268 1191 Ext. 15
Meet: 1st Sun. monthly, 21 Rodney St.
Lindum

COMPUTER OWNERS GROUP

Cont: Betty Adcock, Tel: (07)263 4268

TOWNSVILLE GROUP

Cont: Townsville Amateur Radio Club
Meet: 2nd Tues. monthly, State Energy.
Serv. HQ, Green St. West End.

SOUTH AUSTRALIA

ADELAIDE MICRO USER GROUP

Cont: Rod Stevenson, 36 Sturt St.
ADELAIDE, SA 5000
Tel: 515241 between 9-4

NORTHERN TERRITORY

DARWIN GROUP

Cont: Tony Domigan
PO Box 39086, WINNELLIE 5789.

WESTERN AUSTRALIA

PERTH '80 USERS GROUP

Cont: C. Powell (09) 457 6849
Meet: 1st Tues monthly, 7.30pm, Comm. Rec. Hall, MacDonald St. Yokine.

** UNITED KINGDOM **

COMPUTERTOWN NORTH-EAST

Cont: C/- 2 Claremont Pl. Gateshead, Co. Tyne & Wear NE8 1TL.
Tel: 0632-770036/643417/679119/ 559167

COMPUTERTOWN UNITED KINDOM

Cont: Dave Tebbutt, C/- 14 Rathbone Pl. London W1P 1DE

INTERNATIONAL TRS-80 LEVEL 1 USER GROUP

Cont: Mr. N. Rushton, 123 Roughwood Dr. Northwood, Kirkley, Merseyside L33 9U9.

NATIONAL TRS-80 USERS GROUP

Cont: Brian Pain, 40A High St. Stoney Stratford, Milton Keynes.

NEWCASTLE PERSONAL COMPUTING SOCIETY

Cont: John Stephen Bone - 0632 770036

NORTH-EAST TRS-80 USERS GROUP

Cont: Barry Dunn, 8 Ethick Tce. North Craighead, Stanley, Co. Durham DH9 6BE. Tel: 0207 30184.

NORTH WEST TRS-80 USERS GROUP

Cont: **The Secretary, 40 Cowlees, West Houghton Bolton BL5 3EG**

TANDY OWNERS PROGRAM & INFORMATION CO-OP

Cont: Derek Higbee, 12 Shelley Close, Ashley Heath, Ringwood. Tel: Ringwood 6720

TRS-80 EDUCATIONAL USERS GROUP

Cont: Dave Futcher - Head Teacher, Beaconsfield First & Middle School, Beaconsfield Rd, Southall, Middlesex

WEST HERTS 80 USERS GROUP

Cont: Terry Bradbury, 20 Spruce Way, St. Albans, Herts. Tel:Park St. 73663.

** NEW ZEALAND **

AUCKLAND

Cont: Ron Feasy. Bus. 799366 Hme: 469455
Meet: 1st Tues monthly, 7.30pm NZ Solenoid Co Ltd, 28 Kalmia St. Ellerslie, Auckland.

- 0000000000 -

***** SOFTWARE REVIEW - DUNJONQUEST - TEMPLE OF APSHAI by Terry Bradbury *****

The game comes in a solid well-made cassette accompanied by a lavishly illustrated 56 page 'BOOK OF LORE'. The contents of the book comprise an entertainment in their own right and include a complete listing of all the 200 or so rooms which make up the four levels. Also included are descriptions of all the treasures to be found and their values. Monsters, weapons, strategy and rules are all covered in detail. A final touch is a short story 'THE LEGEND OF THE RUINS OF APSHAI' just to put you in the mood.

Although the game is written in BASIC a special instruction sheet is included and a precise Memory Protect figure is needed in order to run. Enter the wrong number and you will have to start again. My recording seemed to be at a very low level and it took three attempts before I got a good load. This first of three modules takes three minutes to Cload.

On typing RUN a graphics heading is followed by your introduction to 'THE INNKEEPER'. At this point you may enter your own character 'Attributes' or the Innkeeper will provide you with a Character set plus a quantity of silver. You may then haggle with the Innkeeper over the purchase of arms and supplies. Eventually having been kitted out, you choose the level that you wish to enter the dungeons (1-4) and the monster speed (S,M or F). There follows a period of some five minutes during which 36 data files are loaded in. Then a further 3.5 minutes of loading the third and final module (Dunjonmaster) which overwrites the Innkeeper module.

After RUN is typed in, one or more rooms are graphically depicted together with a space ship like symbol which is you. Single letter commands are the order of the day and these cover movement, attack and various special functions like 'S' (search for traps) and 'G' (grab treasure). However, some items which are shown as treasure (single pixel graphics block) may be found on referring to the 'MASTER TREASURE KEY' to be 'NOTHING OF VALUE', all that glitters is not gold.

Monsters appear at random and there are more than thirty different types with varying degrees of strength and ambivalence. Fortunately, when you enter a room which already contains a monster, the monster (indicated by a graphics cross) appears before the room is fully formed, so you are usually able to line yourself up and start firing arrows at long range. You may swing your blade, thrust or parry, or even run if you are badly wounded. Most monsters will not follow you from the room. Advertising blurb claims that there are more than 16 million combinations of character 'Attributes', thus you will be doing well if you design your own and it turns out to be a level four master.

To assist you in your venture some of the treasure items are in fact magic weapons such as wondrous swords enabling you to overcome monsters with minimal effort. Magic arrows, cloaks and potions are also to be found.

Novices of lowly stature are well advised not to enter levels 2, 3 or 4 until they show a significant increase in their 'Attribute' levels, otherwise they will almost certainly be rapidly slain by the first 'MONSTER' they run into!

This game, often referred to as a 'ROLE PLAYING GAME' (R.P.G.), represents really good value for money and for those who would like a change from the conventional adventure type games, it represents a good investment. I estimate it will take several months before I can hold my own, even in level two.

The one annoying feature of the program is apparent when the player survives and returns to the Inn. If he does not wish to re-enter the dungeon but chooses instead to upgrade his character, then the complete loading sequence of all three modules must be carried out again, although all variables are saved. This seems to be the price paid for shrinking the very big Original program to fit into a mere 16K.

- 0000000000 -

***** SOFTWARE REVIEW - FS 1 FLIGHT SIMULATOR by Brian J. Fillery *****

I have always been fascinated by flying, not in your big commercial aircraft - those I don't like, but the small planes I really enjoy. I haven't enough money to learn to fly - I spend too much on this computer, amongst other things, so I was really interested to see an advert in an American magazine advertising a Flight Simulator. Well, I grabbed some money and sent off a bank cheque for $US28 (about $A25) and sat back impatiently.

Now, I have been conned by adverts, as so many of us have, and I was wondering what I was going to get. The ad said there were real 3D graphics and the illusion of flying was very good, etc. etc. When the packet arrived, I got quite a surprise. In addition to the tape there was a very comprehensive 35 page booklet, one of the best instruction books I have ever seen with a computer game. At least a great deal of thought had gone into the book. Firstly, it took 22 pages before it got to the 'how to load the tape' part and, believe me, you will need it for this one. I would say that it would be impossible to learn to fly without reading the instructions.

The plane is none other than the Sopwith FS 1 Camel, Snoopy's favourite, so that should pin your ears back in the breeze of an open cockpit. There are specifications for the performance of the plane and comparisons with other similar aircraft. However, the specs are a little more advanced than normal as you will see.

When you eventually get to the loading bit, and don't rush it if you want the best out of this game, the special care taken is very evident. You will need 16K minimum but you can load the tape into either Level 1 or Level 2. There is a special loader on the beginning that will do the job for you and, incidentally, makes the job of pirating a copy by normal TRCopy methods impossible. You have to load Level 2 by System but after the loader has gone in, you get a further prompt. You answer this by "/ENTER" and the program proper starts to load. More care can be seen here. You notice flashing graphic blocks instead of the normal asterisks and a series of numbers and symbols. If you have loading troubles, these help you in knowing if you have got further than you did before.

When you get it all in, you have a screen view as if you are looking out of the cockpit of an aircraft. In front of you stretches the runway. Down the right side of the screen are indicators for throttle, elevators, and roll. Along the bottom are airspeed and altitude. Down the left side are oil temp., oil pressure, fuel, tachometer, climb rate, heading (compass), and turn rate. In addition, for when you go to war you have a score, bomb and ammo count for the machine guns.

What, go to war? Yes, you can tackle the Red Baron on his home territory. Well, actually there is no mention of the Red Baron, but why not let your imagination fly? The actual flying of the aircraft (it's no longer a computer) is with the keys. F, G & H control rudder left, right and centre. Elevators are controlled with T and V/B, throttle with the left and right arrow keys, and the landing gear up and down with the U and D keys. Then there is the forwards or downwards view controlled by the up and down arrows.

What is this - landing gear and a form of radar on a Sopwith Camel? Yes, that's right, and it makes it a lot easier to fly. The landing gear gives you better simulated take-offs and landings, and the radar is very handy if you get lost. As a matter of fact, you will probably find it a lot easier to do your initial runs with radar.

In addition to the indicators on the screen, you can get messages flashed on the screen telling you if you are out of ammo, in range of the enemy, at war, in danger, in a stall or on radar mode. If you want to drop bombs, you press the W key to declare war, then X for bombs and the space bar to fire your machine guns, and when you eventually get back on the ground and roll to a stop, you apply the brakes with the . key (what else?) They do tell you that you might need a lot of flying practice to get the plane to perform right...believe it!

You have a world in which you fly the aircraft. It is 6 miles x 6 miles and contains an enemy fuel dump, his airfield and a commercial airfield as well as your airstrip and fuelling point. Along one side of the 'world' is a range of mountains which you can fly over or into if you wish. Although the world is 6 x 6 miles, you can fly over the edge without any ill effects, except that you can get lost very easily.

To get the feel of the plane you can spend a bit of time taxiing up and down the runway, trying to keep it on the strip, although running on the grass doesn't hurt it, just gets the air-intake clogged. The view from the screen is not so hot in this attitude and you will find that radar helps here. Once you are in the air you are on your own, and this is where the trouble can start. You have to learn to fly with the controls. Straight flight is easy but it's when you start to turn that the fun begins. There is one great consolation though. If you really get into bad trouble you can press the panic button (0) and you will find yourself back at the start of the runway with your plane once more refuelled.

I won't spoil your fun by telling you what happens when you get into a spin. Or what will happen when you decide you are good enough to declare war on the enemy. All I will say is that you had better have had a lot of practice or you'll get wiped out pretty quickly.

Well, were the ads right or did I do my money? If I was asked to pick a games program that was as well thought out, had a lot of research and a lot of work go into it and was backed by a really first-class instruction book, I would have a problem to pick one to beat FS 1 Flight Simulator.

And now the program is available here, so you don't have to worry about getting bank cheques in US dollars and all that hassle....get one, and happy flying!

- 0000000000 -

***** HARDWARE REVIEW - MICRO-80 LOWER CASE MODIFICATION FOR THE SYSTEM 80 - by G. Lawrence *****

Whilst for many applications the ability to print lower case characters is merely a luxury, there are many others for which it is absolutely essential. These applications include almost all forms of text handling and word processing. That there is a need for such applications is without dispute, and the Micro-80 modification more than makes them possible.

To provide lower case capability for the System 80 it is necessary to add more characters to the character generator (A device which contains the dot matrix pattern for each character, usually it is simply a ROM). It is also necessary to add an extra bit to the words of the video memory, as the standard System 80 is missing bit 6. It is highly unlikely that the provision of this missing bit will affect existing programs.

The Micro-80 kit addresses the first problem by the simple and efficient expediency of completely replacing the character generator ROM with a 2716 EPROM, which has been programmed with the patterns for both the new and the old set of characters. Note that this means that any custom character sets can be easily implemented by simply reprogramming, or exchanging the EPROM. Micro-80 have thoughtfully socketed the 2716, so this process should be very straightforward.

So that the 2716 can pretend to be the normal character generator it is mounted on a small converter board which effectively modifies its pinout configuration. There is also an additional "flying" lead which carries the extra address bit to indicate when the lower case character set is required.

The problem of providing an extra bit for the video memory is solved by the pragmatic approach of "piggy backing" an additional 2102 (1024x1 bit static RAM IC) onto one of the existing video memory ICs. This simply involves mounting one IC over the other, with the legs touching, and thus making electrical contact. This sounds slightly precarious but in practice the idea works well. One might even argue that connecting in this way results in better signal transfer as line capacitance is effectively minimized!

INSTALLATION:

Due to the easy to follow and comprehensive instructions I believe that almost anyone could fit this modification. All that is required is the appropriate tools (electronics soldering iron and screwdriver) plus about an hour of installation time.

The first step is removing the case, if this is the first time you have done this then you will probably find it the hardest part of the operation. That's how easy the rest is!

Removing the keyboard is the next step, eight easily accessible screws facilitate this. Once you have done that the instructions then direct you to remove the right hand logic board so that a track beneath can be cut. In fact, I found that this slightly tedious procedure is unnecessary as the track leads out to the upper surface and can effectively be cut here, just before it leads to pin 4 of IC Z3.

Next the character generator IC is removed and replaced with the modified 2716. There is no need to unsolder the character generator since it resides in an IC socket, nor is there any need to assemble the 2716 and converter board combination as it is thusly provided. The only soldering at this point is to connect a flying lead to the appropriate point in the video circuitry. The old character generator is no longer required, and becomes a potentially useful item in the spare parts box. For those who are not electronically minded the old character generator might find use attaching notices to bulletin boards etc!

The next step is to connect up the 2102 piggy back fashion to one of its existing counterparts. In this procedure I slightly disagree with the instructions, which advocate soldering the ICs together. In my opinion this is quite unnecessary and potentially dangerous to the ICs, especially if the solderer is inexperienced. Furthermore, on my unit the memory ICs are socketed in such a way that soldering would be fairly messy. With these sockets I found that it was quite possible for two ICs to share the same socket quite happily. Although to ensure mechanical stability I recommend a drop of epoxy resin between the casings of the ICs.

At this point two legs (data in/out) of the 2102 have been left unconnected and must be now connected by flying leads to the existing circuit. Having done this the kit is now installed and the System 80 can be re-assembled.

THE RESULT?

Well worth the effort! The new lower case characters are easily readable, even some of the old characters are considerably improved. For instance, it is now impossible to get "D" confused with "O" or "B" mixed up with "8". The punctuation marks are more visible as well. One notable difference is that the cursor is no longer an underline but a solid white block, about half the height of an upper case character. To add icing to the cake there are a number of additional characters such as braces, arrows and playing card symbols. One thing that appeals to me is that the " ↑ " symbol is now shown instead of the square bracket. This makes the System 80 even more compatible with the TRS 80 software.

SOFTWARE PATCH:

For those machine language programs that access the screen memory directly a software patch is unnecessary. However, if the ROM routines are employed, or if you are using BASIC and don't want to always POKE your lower case characters onto the screen then the patch will be required.

Essentially the patch does two things, it intercepts the video driver and prints both upper and lower case. It also intercepts the keyboard driver to allow the entry of lower case characters. This is achieved by providing what might be best termed as a "CAPS UNLOCK" function which when activated causes letters to be interpreted as lower case unless the shift key is also pressed. This function is switched on and off by pressing the SHIFT and "O" at the same time (SHIFT and SPACE also works).

With the kit there are two versions of the software patch provided. The first is a "Bare Bones" version providing the essentials described above. The second is a luxury version with key debounce and flashing cursor. The value of a flashing cursor is in my opinion a little dubious, but the key debounce feature is highly desirable, especially when the keyboard is starting to show its age. Indeed, for the minimal amount of overhead required I believe that key debounce should have been provided as standard.

The software patches are provided on tape in the standard SYSTEM format. An interesting feature is that each patch is initially loaded into the video RAM and a small routine then relocates it to just below the top of BASIC memory. Then it moves the top of memory pointer to just below the patch. This method is made possible only by the fact that the System 80 video memory now has its full 8 bits.

Unfortunately there is a somewhat disconcerting design feature (polite term for Bug) evident in the luxury patch. This prevents the flashing cursor from being erased after NEWLINE has been pressed. Luckily I found that the following procedure put things right.

Load the patch as normal then type "/" and NEWLINE. Having returned to BASIC issue the following commands -

```
POKE nnnnn-114,0
POKE nnnnn-124,0
```

Where nnnnn is the end of BASIC memory, i.e. the number that you answered to READY? or if you simply typed NEWLINE then it is the last memory location plus 2. (e.g. for 16K machine it is 32769)

I trust that Micro-80 will look into this and ensure that the bug is removed from future copies of this routine. Another minor beef is that the SHIFT-@ print pauser is disabled whenever "CAPS UNLOCK" is selected.

Despite the above minor complaints I hasten to say that on the whole I am extremely pleased with the lower case modification. It features ease of installation, convenience of use, versatility and compatibility with existing software. At a price of around $50 I believe that it is indeed good value for money.

(The minor bugs in the driving software have been reported by several other readers and will be corrected in the near future. The source code listings for these programs were published in MICRO-80 Issue 18 (May 1981) page 29. The changes necessary to overcome these bugs will be published in MICROBUGS in the near future. -Ed.)

- 0000000000 -

***** SM ELECTRONICS CONVERSION OF THE OLYMPIA ES100 ELECTRONIC DAISYWHEEL TYPEWRITER by David Hill *****

The most important comment to make about this conversion of the electronic typewriter into a RS 232C computer printer is .."It Works!" This conversion, whilst relatively inexpensive when lined up against dedicated letter quality printers, does miss out on some of the refinements one would desire in an ideal printer. Some compromises are to be expected when it is considered that the Olympia was originally primarily designed as a typewriter.

There is no front panel indication of whether the terminal is on- or off-line: a small indicator lamp or the mounting of the switch in a visible position would have helped. Proportional spacing, super- or sub-scripts, overstriking and (with Scripsit) underlining are refinements you will have to do without.

If the margins were reset for typewriter use and then not readjusted before going on line, one can be in for some nasty surprises. This is especially so when using Scripsit which does not allow "break" once the print command has been issued. This inability to interrupt printing, the loss of keyboard control of paper feed whilst on line, and the relatively large buffer can make things interesting if careis not exercised before 'print'.

The unit is relatively quiet, the image is definitely letter quality. Speed, however, is not startling with an actual 12 cps average delivered whilst printing a page. One unexpected bonus: the < > symbols do not appear on the Olympia keyboard. However, they are on the daisywheel and do print under the driver routine. Copy is clear and legible and could, I believe, be used profitably to LLIST programs for magazine reproduction.

The conversion of my typewriter was completed in less than a day by ASP Microcomputing of Melbourne (the Stringy Floppy people) who also supplied the driver routine and cassette port adaptor of their own design to allow my non-expansion-interface TRS-80 to drive this RS 232 printer. I was most impressed with the quality of their service.

- 0000000000 -

***** INPUT/OUTPUT *****

From: R.J. Jennings - Seaford, Victoria.

SCRIPSIT - CONVERSION FOR THE SYSTEM 80.

The SCRIPSIT word processor was reviewed in MICRO-80 for April 1981. This program will load and run on a System 80 but the PRINT command will not produce the desired output. In fact, the computer will hang up as though the printer is not connected.

The problem is the difference in line printer address between the System 80 and the TRS-80. The Tandy line printer is at memory location 37E8H and the System 80 addresses port FDH. The BASIC LPRINT and LLIST commands in each case are directed to the appropriate location but a program in assembly language specifies input or output to the actual address. The I/O routines in Scripsit must therefore be changed to the correct location for the System 80.

Disassembly of the SCRIPSIT program indicates two instructions that address the location 37E8:

```
3AE837    LD  A,(37E8H)    - - - - - Input from printer address
32E837    LD  (37E8H),A    - - - - - Output to printer address
```

In each case exchange is between the printer address and the Accumulator. The Hex representation of memory is shown on the left where 3A or 32 is the appropriate Z80 code for the LOAD instruction and the remainder is the address. Note that the address is stored backwards in memory.

To address the port FD for the System 80 case the instructions IN and OUT may be used as follows:

```
DBFD00      IN   A,(FDH)
D3FD00      OUT  (FDH),A
```

These instructions use one byte less than the original so a NOP (no operation) code is added as ØØ in each case. Modification is simply done by changing the appropriate memory locations. A monitor program such as BMON is required. The method is as follows:

1. Load BMON under SYSTEM command and type "/NEWLINE" as normal to leave the '80 in the BASIC mode.
2. Re-enter SYSTEM mode and load SCRIPSIT.
3. When SCRIPSIT is loaded press BREAK to return to BASIC.
4. Activate BMON and edit memory as follows:

ADDRESS	EXISTING	MODIFIED	ADDRESS	EXISTING	MODIFIED
4344	32 E8 37	D3 FD 00	5636	32 E8 37	D3 FD 00
4F18	3A E8 37	DB FD 00	56EC	32 E8 37	D3 FD 00
5617	3A E8 37	DB FD 00	69B8	32 E8 37	D3 FD 00
5628	3A E8 37	DB FD 00	69BF	32 E8 37	D3 FD 00

   NOTE: Three successive bytes are shown after each address.

5. Now exit the EDIT Mode and use BMON to prepare a new SYSTEM tape of the modified SCRIPSIT program. With my program the following parameters were used:

   START : 4300 END : 69C5 ENTRY : 4300

The above modification has proved successful with a 16K System 80 providing the desired output of text on the normal PRINT command.

(This type of software remedy will generally enable M/L programs used on the System-80 to drive the printer. One drawback, though, is that all such programs need to be modified and the exact locations will need to be found, as Mr. Jennings has done here. The other approach is to make a hardware modification, outlined in the Dick Smith Technical Manual, so that the printer is addressed at the same location as in the TRS-80 - 37E8H. - Ed.).

From: Mr. W.G. Neumann, Yeerongpilly, Qld.

Please find listed some comments regarding the program "FILES" as published in Vol. 3 No.2 January 1982. These comments arise from my own experience in getting the program operational on my System 80/48K RAM/2 disks. They may assist others who are having problems.

1. Data format - it should be emphasised that the operator must enter the number of headings as a decimal number followed by a comma and the headings in the first line of data.
2. Line one of the data must NOT be edited.
3. To enable use of this program on a disk based system with DISK BASIC the following alterations are necessary -

   ```
   Line 6  - - - - -:CLEAR5Ø:DEFUSR1=&HFE7E:X=USR1(Ø)
   Line 29Ø POKE164Ø5,Ø:DEFUSR2=&HFEØØ:X=USR2(Ø)
   ```

4. A new menu command may be made by inserting the following program lines -

   ```
   58 PRINT"          6 = = = SAVE DATA"
   EDIT 11Ø
   11Ø - - - :ON B GOTO 12Ø,31Ø,54Ø,56Ø,67Ø,9ØØ
   9ØØ CLS:PRINT" DATA TO BE SAVED"
   9Ø2 PRINT" PRESS (S) TO SAVE"
   9Ø4 PRINT" PRESS (R) TO RETURN TO MAIN MENU"
   91Ø A$=INKEY$:IF A$="" THEN 91Ø ELSE IF A$="R" THEN 4Ø
                          ELSE IF A$ <>"S" THEN 91Ø
   92Ø CSAVE"F":END: REM For cassette users
   or
   92Ø SAVE"FILENAME":END: REM For disk users
   ```

5. Keybounce can cause problems when entering new data causing incorrect DATA lines to be inserted. these should be edited from the program before saving.
6. Various references are made in the text to ' DOWN ARROW/ESC ' These references should be to ' DOWN ARROW/CTRL '.

7. On some TRS-80 and System 80's the instruction PRINT CHR$(92) generates a backward slash " \ " which is not the same as DOWN ARROW/CTRL. I have changed all references to CHR$(92) to "CTRL".

8. If anyone alters the line numbering, they must ensure that line 3ØØ is still at line 3ØØ. This is necessary because the author has used a trick to re-enter the program after making the DATA line from the input.

Thank you for your enjoyable magazine and I hope these comments are of some value.

(Thank you for this contribution, Mr. Neumann. The version of FILES supplied to Disk subscribers had somewhat similar modificatons to those you describe in lines 6 & 290 and works well. The Save Data function should be a useful addition to the program Ed).

- 0000000000 -

***** NEWDOS 2.1 - JKL FEATURE V. EPSON MX 80. THE FINAL ANSWER? by Ken B. Smith - Oman *****

I have been a firm fan of the boys from Denver ever since the appearance of NEWDOS 2.1 some four centuries ago (in microcomputer terms). However, when they wrote the modifications and additions to TRSDOS 2.3 that form this particular DOS they did not include the facility to JKL with a lower case modification or to print graphics. The second is excusable, but the first is rather a letdown. Well, I suffered with this handicap for about 2 hours after fitting my L/C mod. and, being an avid user of JKL, I was soon put to doing something about it. This was two years ago and, more recently, I converted to LDOS and the problem has disappeared. However, our club machine runs on NEWDOS 2.1, has a L/C mod. and an EPSON MX 80 printer. The problem reared its very ugly head - yet again...

The complaints were coming in thick and fast - as if it was my fault. I mooted that the solution was reasonably easy and that it was not beyond the skill of a couple of our members to find out the answer for themselves. I was wrong. In fact, the solution is indeed very easy but rather obscure. The following is the result of my giving in to the pleas of others and a couple of hours reading my old notes and 'mucking through the disk'.

The first trick is to find the enemy. With the JKL printing routine, the NEWDOS documentation, (who said "What documentation?" - leave the room) leaves a little to be desired, in fact, it is downright misleading. The answer is to search for the call to the printer routine (003BH) around the SYS0/SYS area in memory or disk. Then you find the JKL processing code. In memory this is in the 43BDH to 43 DCH area and on disk it lives on track 0, sector 06, relative byte 0A to 88H.

"Terrific" - you say. So what! Well, it actually constitutes a rather neat piece of coding. Let's take a closer look to save you getting the TBUG/MON3 out of mothballs.

JKL processing - NEWDOS - Memory locations 43BDH to 43DCH

```
43BD    D5          PUSH DE                 : SAVE DE REGISTER TO STACK
43BE    21003C      LD  HL,3C00H            : START OF VIDEO TO HL
43C1    7D          LD  A,L                 : LOAD ACC WITH LSB OF VIDEO
43C2    E63F        AND 3FH                 : AND THE ACC WITH 63 (CHRS/LINE)
43C4    3E0D        LD  A,0DH               : LOAD 13 INTO ACC (LINE FEED)
43C6    CC3B00      CALL Z, 003BH           : SEND ACC TO PRINTER IF THE AND 3FH CHECK ON
                                              L SETS ZERO FLAG
43C9    7E          LD  A,(HL)              : LOAD ACC WITH 'PEEK' HL
43CA    CB74        BIT  6,H                : NICE TOUCH. VIDEO ENDS AT 3FFFH 00111111.
                                              SO WHEN BIT 6 GOES HIGH WE ARE FINISHED
                                              WITH JKL
43CC    23          INC   HL                : NEXT VIDEO LOCATION
43CD    200B        JR  NZ,43DAH            : IF BIT 6 HIGH - FINISH
43CF    FE80        CP  80H                 : COMPARE ACC WITH 128 DEC.
43D1    3802        JR  C,43D5H             : GOTO PRINT IF ACC LESS THAN 128
43D3    3E2E        LD  A,2EH               : PUT A FULL STOP (.) IN THE ACC
43D5    CD3B00      CALL 003BH              : LPRINT CONTENTS OF ACC
43D8    18E7        JR  43C1H               : GO BACK TO CHECK FOR LINE FEED
43DA    D1          POP  DE                 : GET DE BACK FROM THE STACK
43DB    AF          XOR  A                  : CLEAR ALL THE FLAGS FROM AF
43DC    C9          RET                     : BACK TO THE CALLING CODE
```

A nice piece of code for brevity's sake, but rather awkward for a patch. There is no space or 'NOPs' around this area to patch into so we must use the available space. First let's examine the aim.

With the lower case mod. installed the PEEKs to the screen (from BASIC or from Machine Code) will produce a false character for the upper case letters. Usually these are from ASCII 65 to 91 (A - Z) but once the lower case is installed, with or without the driver, these upper case letters appear as 1 to 26 when PEEKed. Essentially we must add 64 to all PEEK results below 32 to get the correct character output to the printer and avoid all that wasted paper when a line feed or page feed is generated instead of the letters M or L.

MICRO-80 PRODUCTS - CATALOGUE

HIGH QUALITY PRODUCTS FOR YOUR COMPUTER AT UNBEATABLE PRICES.

ABOUT MICRO-80 PRODUCTS

MICRO-80 PRODUCTS was started at the request of MICRO-80 readers who wanted to obtain good quality peripherals and software for their computers at reasonable prices. In the past 2 1/2 years literally thousands of satisfied customers can attest to the fact that MICRO-80 PRODUCTS has achieved this objective. We have removed much of the mystique which surrounds the interfacing of such useful peripherals as disk drives and printers and have become the major Australian source of supply for such software products as NEWDOS and DOSPLUS which have increased the power and speed of TRS-80 micro-computers enormously. More recently, we have saved Hitachi owners considerable sums by interfacing MPI disk drives to the Hitachi Peach. We were the first in Australia (in the world?) to successfully interface the range of Olivetti electronic typewriters to be printers and have designed and produced a number of useful modifications for the TRS-80/System 80.

As the interest of micro-computer users broaden, so do our own. We now actively sell and support the TRS-80 Model 3, the Osborne 1, the Hitachi Peach, the Olivetti M20 microcomputer, the North Star Advantage and the Altos multi-user system. We would be happy to assist you in upgrading your present computer with new peripherals or even exchanging it for a more modern machine.

MAIL ORDER POLICY

Much of our business is carried out by Mail Order and our customers find it a simple and efficient way to do business. You may place your order by telephone or by mailing in the order form from any issue of MICRO-80 magazine. Generally, it takes about 1 week from receipt of order until dispatch. You should allow 2-3 days for your letter to reach us and 7-10 days for the parcel to reach you making a total turnaround time of about 3 weeks. If we are temporarily out of stock of an item, we will send you a notification of back order giving our best estimate of when it will be back in stock. Payment, which should accompany the order, may be by Cheque, Money Order, Bankcard or Access. If we are unable to supply an order immediately, we apply the following rules:

-If payment is by cheque and none of the order is in stock, the cheque is not presented until the order can be fulfilled.

-If payment is by cheque and some items are in stock, the cheque is presented and the items back ordered are shown on the invoice which accompanies the goods.

-If payment is by Bankcard or Access, only these items which can be supplied are charged. Back ordered items are not charged until available.

If you wish to speed up delivery, you may pay a special delviery fee to have the item sent by road freight or even air express. Ring for prices.

WARRANTY AND SERVICE

All hardware products carry a 90 day parts and labour warranty either from the manufacturer/distributor or from MICRO-80 PTY. LTD. In many cases, warranty servicing can be arranged in your own city, otherwise goods should be returned to MICRO-80 PTY. LTD. the cost of freight to MICRO-80 is at customer's expense. Return freight on goods which require repair or adjustment, either by road or post at MICRO-80's discretion, will be paid for by MICRO-80 PTY. LTD. Customers should obtain a return authorisation from MICRO-80 before despatching goods for warranty repair, post warranty servicing can also be carried out at very reasonable rates.

-0000000000-

TRADE-INs, EASY PAYMENT TERMS

MICRO-80 BRINGS COMMONSENSE TO COMPUTER BUYING

If you wish to buy a new car, you are able to trade-in your existing vehicle and arrange finance for your new purchase, all under the one roof. Not so with microcomputers. If you want to dispose of an existing machine, you are on your own and, in most cases, you must make your own arrangements about finance, too. Here at MICRO-80 we think this is ridiculous, so we have done something about it. We are now able to accept TRADE-INs on new COMPUTERS and PERIPHERALS and to arrange CONSUMER MORTGAGE terms to approved customers. This offer applies to our customers ALL OVER AUSTRALIA, not just in South Australia.

Here is what you do.
If you are interested in trading-in existing equipment:-

1) Write to us or phone us, describing the equipment you wish to trade-in. Make sure you tell us its age and any distinguishing features. Eg.: TRS-80 Model 1, early style keyboard with "square' monitor, L2/16K, 3 years old, good condition.

2) Tell us too, what computer you wish to purchase from our range of Hitachi, TRS-80 Model 3, Osborne, Olivetti and North Star.

3) We will write, offerring you a trade-in valuation and quoting the price of the equipment requested. Our trade-in offer will be subject to inspection of the equipment at our premises. Our letter will also include instructions for sending the equipment to us in the most cost effective manner.

4) If you are satisfied with our offer and quotation, together with payment for the balance (or if you wish to purchase on terms, see 6 below) and we will send your new computer to you.

If you would like to take advantage of consumer mortgage or leasing finance, with or without a trade-in:-

5) Write or 'phone telling us the equipment you wish to purchase.

6) We will send you a written quotation, an order form and an application form for the appropriate finance.

7) Complete the application and order forms and return them, together with a cheque made out to the finance company indicated, for the first monthly payment. The order is conditional upon you obtaining finance of the required amount at the quoted rate. If this is not available at the time the order is received, we will contact you for further instructions. South Australia has some of the most stringent regulations in Australia affecting consumer financing and you may rest assured that your interests will be well protected.

8) When authorised to do so by the finance company, (generally 3-7 days) we will despatch the new equipment to you.

EASY PAYMENT TERMS ALSO AVAILABLE ON PERIPHERALS

The same consumer finance is also available on hardware peripherals selling for more than $250. For example, if you require a disk drive costing $499, you could purchase it on 10% deposit and payments of only $4.17 over a period of 36 months.

Even software can be included in the overall purchase to a limited extent. Eg.; If you purchase a new computer system then you could also finance a Disk Operating System and application programs up to about 10% of the total value of the purchase.

WE HAVE CUSTOMERS WAITING FOR USED COMPUTER SYSTEMS

In high demand are TRS-80 Model 1 systems with one or more disk drives. If you have such a system, why not trade it in on a new computer?

Finance and leasing facilities to approved clients is available through "ESANDA" Adelaide.

-0000000000-

***** BOOKS *****

THE CUSTOM TRS-80 & OTHER MYSTERIES
$32.50 + $1.20 p.&p.

The complete guide to interfacing your TRS-80 to the outside world, covering both hardware and software.

TRS-80 DISK & OTHER MYSTERIES
$27.00 + $1.20 p.&p.

A must for the serious disk user. Disk file structures revealed. DOS's compared and explained, how to recover lost files, how to rebuild crashed directories.

LEARNING LEVEL 2 NOW ONLY $7.95 +$1.20 p.&p.

Written by David Lien, the author of the TRS-80 Level 1 Handbook, this book teaches you, step-by-step, how to get the most from your Level 2 machine. Invaluable supplement to either the TRS-80 Level 2 manual or the System 80 manuals.

Level 2 ROM ASSEMBLY LANGUAGE TOOLKIT
$29.95 + $1.20 p.&p.

The definitive work on using Level 2 ROM routines in your own programs. Covers TRS-80 Model 1 and 3 and System 80. Comes complete with DBUG, a machine language debugging monitor distributed on cassette. This package is a must for machine language programmers and BASIC programmers.

BASIC BETTER AND FASTER
$32.50 + $1.20 p.&p.

Fast becoming the "bible" on the TRS-80 for BASIC programmers , this book is packed full of useful routines and techniques all fully explained, which you can use in your own programs. If you are serious about learning to program, then this is a must.

LOOKING FOR A NEW COMPUTER?

MICRO-80 PRODUCTS HAS A COMPLETE RANGE OF
HIGH QUALITY MICRO-COMPUTERS FOR EVERY REQUIREMENT.

HITACHI PEACH - Add Some Colour To Your Life.
The Hitachi Peach is built to the highest quality standards we have seen in a microcomputer. Not surprising really when it comes from Japan's leading computer company. The Peach can be a versatile, cassette based system for home use or a fully expanded business system carrying out word processing, accounting, financial planning etc. etc. When you see a Peach, you will quickly be convinced, as we are, that it is the shape of things to come. Within the next few years high resolution colour graphics will become a standard feature on business computers. Fortunately, you do not have to wait. The Hitachi Peach is here now and so is high quality, applications software for business use.

Standard features of the "Peach" include 40/80 character display, Upper/Lower case with descenders, high density graphics (640 x 200), 8 colours, numeric keypad, ten programmable function keys, speaker with volume control, RS232 Port, Centronics parallel printer port, Red/Green/Blue video output, light pen port, 32K RAM, 24K ROM, six expansion slots, RAM expandable to 64K in cabinet, 6809 Microprocessor, and Microsoft basic.

		PRICE	*WEEKLY PAYMENTS
32K CASSETTE SYSTEM Consists of Hitachi Peach with 32K RAM, 24K ROM, Mono Cassette deck and High Resolution Monitor	Green Screen Colour	$2,100 $3,000	$17.57 $25.10
48K SINGLE DENSITY DISK SYSTEM Hitachi Peach with 48K RAM, 24K ROM, Disk Controller two 5 1/4 inch single-sided single density MPI disk drives (80 Kbyte capacity each) and disk operating system	Green Screen Colour	$3,300 $4,200	$27.79 $35.13
Two 5 1/4 inch double-sided single density disk drives, controller, and disk operating system in lieu of single-sided drives.	Add	$ 250	$2.09
48K DOUBLE DENSITY DISK SYSTEM Hitachi Peach with 48K RAM, 24K ROM disk controller two 5 1/4 inch double-sided, double desnity disk drives (280 Kbyte capacity each) and Disk operating system.	Green Screen Colour	$4,400 $5,300	$36.81 44.33
COMPLETE PROFESSIONAL WORD PROCESSOR IN COLOUR! 48K single density disk system with colour monitor ITOH 40 character per second daisywheel printer and HiWriter word processing program.		$6,450	$53.95
COLOUR ACCOUNTING SYSTEM 48K double-density disk system with colour monitor, Star dot matrix printer and any three modules from the HiFinance Accounting Software e.g. Debtors, Inventory and Invoicing. This System is capable of handling up to: - 1200 inventory items - 100 creditors - 100 general ledger accounts - 300 transactions per month	From	$6,900	$57.72
The capacity can be more than trebled by using DS/DD 8 inch drives.	Add	$ 700	$5.85

Finance to approved customers is available through "ESANDA" Adelaide. Weekly payment figures shown are approximate only and are calculated after deducting 10% deposit from the selling price, using an effective rate of interest of 25.98% pa over a period of 36 months.

-0000000000-

ANNOUNCING THE OLIVETTI M20

FULL 16 BIT PROCESSING POWER IN AN OFFICE MACHINE.

The Olivetti M20 represents a breakthrough in computer design - both technically and aesthetically. From its latest full 16 bit technology using a Z8000 microprocessor to its attractive styling, it is as modern as tommorow. Most important however, are the human interfaces. Olivetti has drawn on its years of expereince in office equipment design to produce a computer which is easy and comfortable to use. The keyboard is well laid out with a nice "feel", the disk drives are quiet and unobtrusive whilst the all important display is without equal. We have not seen such an attractive, easy to read display anywhere. The special anti-glare treatment makes text readable under the most adverse lighting conditions whilst characters are of high definition and clearly formed. The high resolution graphics display (512 x 256) is sharp and rock-steady. If you think we are laying it on a bit thick, then you haven't seen the M20!

The Olivetti M20 is primarily designed to be used in a commercial or technical office although its low price could also make it attractive to the serious hobbyist or programmer. The basic specification is: Full 16 bit Z8001 microprocessor, 128K RAM expendable in the cabinet to 512K, one 5 1/4 inch, double-sided, double density 35 track disk drive (320K unformatted capacity) RS232 serial selectable port, Centronics parallel port, software 80 x 24 or 64 x 16 display with high resolution characters - lower case with full descenders, fully programmable keyboard, PCOS Disk Operating System and MICROSOFT Version 5.2 BASIC interpreter on Disk and high resolution graphics (512 x 256).

Options include, second disk drive in cabinet, winchester hard disk drive, IBM 3870 terminal emulation, videotext, high resolution colour graphics.

The operating software and BASIC interpreter support graphics commands such as DRAW, BOX, PAINT, 16 INDEPENDENT WINDOWS, LABELS, WHILE AND WEND AND MANY OTHER POWERFUL STATEMENTS. The operating system and interpreter reside in RAM so no system disk is required after boot-up thus making a single disk system viable. Olivetti is fully supporting the M20 with a wide and ever increasing range of applications software. Available immediately is the OLIBIZ suite of accounting programs - written in Australia, MULTIPLAN financial planning, ISAM file handling utility and a wide range of scientific and engineering sub-routines. which have been thoroughly tested on earlier Olivetti systems. Releases scheduled over the remainder of 1982 include OLIWORD word processor, OLIENTRY data handler, OLISORT high speed sorting routine and an Editor/Assembler. If you require an up-to-date powerful super-fast and well supported computer for your business, then you should seriously consider the OLIVETTI M20.

SINGLE DISK DRIVE SYSTEM consists of Olivetti M20 with 128K RAM, one DS/DD disk drive (320K unformatted) PCOS disk operating system and Microsoft BASIC 5.2	$4,400 inc. ST. $37 per week
ACCOUNTING SYSTEM consists of Olivetti M20 with 128K RAM two DS/DD disk drives (320K unformatted) complete suite of OLIBIZ programs including Debtors, Creditors, Inventory, Invoicing and General Ledger, 80 Column dot matrix printer.	$7,500 incl. ST. $63 per week

Add $30 for road freight to anywhere in Australia.

Finance to approved customers is available through "ESANDA" Adelaide. weekly payment figures shown are approximate only and are calculated after deducting 10% deposit from the selling price, using an effective interest rate of 25.98% over a period of 36 months.

-0000000000-

SPECIAL OFFER - STOCK CLEARANCE

SAVE $600 ON A BRAND NEW OSBORNE 1
ONLY $1,995 INCL. SALES TAX

The new blue-case Osborne 1 is on the way so we are closing out our stock of current model brown-case Osborne 1's below cost. The new Osborne will differ only in the shape and colour of its case. It will be the same as the current model in every other way. Now is your chance to secure the computer bargain of a life time and we will accept trade-ins and arrange terms too! HURRY, this offer only applies while stocks last.

THE OSBORNE 1 BUSINESS SYSTEM
$2,595
THE BEST VALUE FOR MONEY IN COMPUTING

Without any shadow of a doubt, the Osborne 1 represents the best value for money available in a computer system anywhere. Not only does $2,595 buy you a 4MHZ, 64K CP/M computer with two disk drives but also a well thought out selection of applications and utility software. Osborne pioneered the complete package aproach in microcomputers and no one else has been able to equal it. The software supplied with the Osborne includes:

- WORSTAR word processing program. WORDSTAR is undoubtedly one of the leading word processors available on a microcomputer.	Retail Value	$ 495.
- MAILMERGE, used in conjunction with WORDSTAR, MAILMERGE enables you to carry out mass mailings of personalised letters.	Retail Value	$ 150.
- SUPERCALC, sophisticated electronic spread sheet for budgeting, cash flow predictions engineering calculations, prices lists etc.	Retail Value	$ 295.
- CBASIC compiler for developing your own application programs.	Retail Value	$ 150.
- MBASIC BASIC interpreter for writing interactive programs.	Retail Value	$ 390
- CP/M Disk Operating System	Retail Value	$ 150.
	TOTAL RETAIL VALUE OF SOFTWARE	$ 1630

As if all that is not enough, the Osborne 1 is portable. You can take your computing power with you, wherever you need it. Do your budgetting in the quiet of your own home, enter your sales results and orders in the motel room as you travel, write your reports while they are still fresh in your mind. Wherever you go your Osborne computer can go to increase your productivity and automate those dreary administrative tasks.

OSBORNE PORTABLE WORDPROCESSING SYSTEM — $3,590
CONSISTS OF OSBORNE 1, PRAXIS 35 TYPEWRTER/PRINTER — $33 per week
AND CABLE.

OSBORNE WORDPROCESSING STATION — $4,850
CONSISTS OF OSBORNE 1, ITOH 40CPS DAISYWHEEL PRINTER — $41 per week
AND CABLE 12 INCH GREEN MONITOR.

Add $20 for road freight to anywhere in Australia.

Finance to approved customers is available through "ESANDA" Adelaide. Weekly payment figures shown are approximate only and are calculated after deducting 10% deposit from the selling price, using an effective interest rate of 25.98% over a period of 36 months.

-0000000000-

BUY YOUR MODEL 3 FROM MICRO-80 AND SAVE $000's

MICRO-80 fits reliable MPI disk drives to the TRS-80 Model 3 to give system capacities and capabilities far in excess of those available elsewhere. All our conversions utilise low dissipation, switching-mode power supplies to avoid screen jitter and overheating. The disk controller boards used incorporate special compensation circuitry for 80 track disk drives and may also be used to run 8 inch disk drives with an appropriate cable and DOS.

MODEL 340
2 40 TRACK SINGLE-HEAD DISK DRIVES GIVING
350K FORMATTED STORAGE, 48K RAM **$3130**

MODEL 340+
2 40 TRACK DUAL-HEAD DRIVES GIVING
700K FORMATTED STORAGE, 48K RAM **$3350**

MODEL 380+
2 80 TRACK DUAL-HEAD DRIVES GIVING
1.4 MEGABYTE FORMATTED STORAGE, 48K RAM **$3800**

★ *NEW* ★ ★ *NEW* ★ ★ *NEW* ★

MODEL 500 — 5+ MEGABYTE MODEL 3
1 40 TRACK DUAL-HEAD DRIVE GIVING
350K OF FLOPPY DISK STORAGE FOR TRANSFERRING
PROGRAMS AND BACKUP, 48K RAM, EXTERNAL
5 MEGABYTE WINCHESTER SUB-SYSTEM,
CP/M (ORG 4200N) DISK OPERATING SYSTEM **$5895**

The MODEL 500 offers the high speed, mass storage capacity and reliability of a Winchester drive for thousands of dollars less than you would pay for any comparable system. Model 500 is a serious business computer able to tackle the most demanding tasks.

All prices are in Australian dollars, include Sales Tax and are subject to change without notice. Prices are FOB Adelaide. Add $20 road freight anywhere in Australia. All computers and systems carry MICRO-80's 90-day Warranty covering parts and labour.

PRINTERS GALORE AT UNBEATABLE PRICES

MICRO-80 has a range of printers to suit every requirement from dot-matrix to correspondence quality daisywheel. Chose from the table below:

BRAND	MODEL	TYPE	SPECIFICATIONS									
			COL	SPEED CPS	BI-DIR	LOWER CASE	PAPER FEED	GRAPHICS	INTER FACES	FREIGHT	PRICE	WEEKLY PAY-MENTS*
STAR	DP	DM	80	80	Y	ND	F/T	BLOCK	P	1	$ 575	$4.81
EPSON	MX-80	DM	80	80	Y	FULL	F	BLOCK	P	1	$ 899	$7.53
EPSON	MX-80II	DM	80	80	Y	FULL	F/T	HI-RES	P	1	$ 999	$8.35
EPSON	MX-100	DM	132	100	Y	FULL	F/T	HI-RES	P	1	$1500	$12.55
MICROLINE	83A	DM	132	120	Y	FULL	F/T	BLOCK	P/S	1	$1599	$13.37
MICROLINE	84	DM	132	200	Y	FULL	F/T	HI-RES	P	1	$2220	$18.57
MICROLINE	84	DM	132	200	Y	FULL	F/T	HI-RES	S	1	$2340	$19.57
C ITOH	8510	DM	80	112	Y	FULL	F/T	HI-RES	P	1	$1099	$9.19
C ITOH	M1550	DM	132	120	Y	FULL	F/T	HI-RES	P	1	$1499	$12.54
DATA SOUTH	DS-180	DM	132	180	Y	FULL	T	OPT.	P/S	1	$2590	$21.66
OLIVETTI	PRAXIS30	DW	100	6	N	FULL	F	NO	P	1	$ 895	$7.49
OLIVETTI	PRAXIS35	DW	100	6	N	FULL	F	NO	P	1	$ 995	$8.33
OLIVETTI	ET121	DW	132	12	N	FULL	F	NO	P	2	$1500	$12.55
OLIVETTI	ET221	DW	132	16	N	FULL	F	NO	P	2	$2650	$22.17
ITOH	F10-40P	DW	132	40	Y	FULL	F	NO	P	2	$1950	$16.31
ITOH	F10-40S	DW	132	40	Y	FULL	F	NO	S	2	$2190	$18.32

NOTES: The following symbols are used:

	TYPE+	DM = DOT MATRIX. DW = DAISYWHEEL
-	BI DIRECTIONAL	Y = YES; N = NO
-	LOWER CASE	FULL - means Lowercase descenders go below line ND - means Lowercase descenders do not go below line
-	PAPER FEED	F - means Friction Feed T means Tractor Feed F/T - means both Friction and Tractor feed included in the price
-	INTERFACES	P = PARALLEL (Centronics). S = SERIAL (RS232)
-	FREIGHT	1 - Add $10 for road freight anywhere in Australia 2 - Add $20 for road freight anywhere in Australia

Note Prices subject to change without notice. Prices quoted include Sales Tax at the 17.5% rate.

Call or write for more details.

NOTE!!! NEW LOW PRICES FOR PRAXIS 30 AND 35 PRINTER/TYPEWRITERS

Finance to approved customers is available through "ESANDA" Adelaide. Weekly payment figures shown are approximate only and are calculated after deducting 10% deposit from the selling price, using an effective interest rate of 25.98% over a period of 36 months.

-0000000000-

EDITOR ASSEMBLER PLUS NOW ON DISK!!!!!!

The disk version of Editor/Assembler Plus does everything that the cassette version does with the added speed and convenience of disk drives. the best news is the price, only $39.50 plus $1.20 p.&p.

-0000000000-

DISK OPERATING SYSTEMS FOR TRS-80/SYSTEM 80 COMPUTERS

You can increase your programming productivity, the execution speed and 'user friendliness" of your programs by using an enhanced Disk Operating System (DOS). MICRO-80 recommends DOSPLUS and NEWDOS 80 according to your requirements and experience.

USERS REQUIREMENTS	RECOMMENDED DOS	PRICE	ORDERING INFORMATION
Single-sided Disk Drives, Economy, First-Time User (requires TRSDOS & DISK BASIC MANUAL to supplement DOSPLUS MANUAL.	DOSPLUS 3.3	$ 99.95	Specify Model 1 or Model 3. If Model 1 whether single or double density.
Single or Double-sided Disk Drives, any track count 5 inch or 8 inch. First-time or experienced user wanting Fuss-Free, Bug-Free easy to understand, but very powerful DOS which support variable length records up to 255 Bytes long with stand alone manual. High degree of compatability with TRSDOS.	DOSPLUS 3.4 Highly Recommended	$149.95	Specify Model 1 or Model 3. If Model 1 whether single or double density
Single or Double-sided single or double density disk drives, any track count. 5 inch or 8 inch. Experienced user who has already used TRSDOS and understands the manual. Requires the most powerful DOS available and is prepared to learn the somewhat complicated Syntax. Requires flexible and powerful file handing in BASIC including variable length records up to 4096 Bytes long. Definitely not for the Beginner.	NEWDOS 80 Version 2.0	$169.00	Specify Model 1 or Model 3

NEWBASIC $99.95 PLUS $1.20 P.&P.

BASIC is the programming language used on most microcomputers. One of its main limitations is its unstructured nature which not only leads to untidy and complicated code but also allows very little portability of code from one program to another. NEWBASIC overcomes this limitation by adding PROCEDURE CALLS and enabling you to define BLOCKS thus localising parts of your program yet enabling you to pass parameters to the remainder of the program. With NEWBASIC loaded on top of your BASIC interpreter, you have the familiarity and interactive nature of BASIC with many of the advantages of PASCAL. NEWBASIC adds the following facilities to your interpreter.

COMMANDS & FUNCTIONS

BREAK	lets you program commands for breakpoints.	0.3K
CALL	now you have procedures and sub-programs in BASIC.	*
CONT	continue after a break by just pressing enter.	0.1K
DEF BLOCK	localise parts of your programs yet pass parameters.	*
DEF END	end of a BLOCK, FUNCTION, or PROCEDURE.	*
DEF FUNCTION	start of a multi-line function.	**
DEF PROCEDURE	start of a CALLed procedure.	*
FIELD @	point strings at any part of memory.	0.1K
&FIND	find strings very quickly, anywhere in memory.	0.5K
&FN	access to multi-line functions	**
MERGE	Very speedy loading of programs	***
MOVE	copy memory anywhere, fill it with anything, fast	0.3K
PLUG	chain + pack parts of your program, keep running.	***
RESERVE	reserve and release protected memory as you run.	0.3K
STRINGS	extend and reduce string space when you want to.	0.2K
TIME	measure the time taken by any lines in your program.	1.0K

NewBasic has a 2.9K mandatory root.
* 5.0K in total for blocks.
** 0.5K for functions in addition to blocks.
*** 1.0K for segmented overlaying.

NEWBASIC requires a single disk drive TRS-80 1 or 3 with at least 32K of RAM using TRSDOS, NEWDOS, or NEWDOS 80 Versions 1 or 2.

-0000000000-

We must find room to add this to the code. But inspection shows some redundant code for an EPSON or other graphic capable printer owners. The check for any character above 80H (128) and its replacement with a 2EH (Full Stop) is no longer needed. In fact, it stops us getting the graphics that our printer is capable of printing. So here is the place to patch our new code. It is really delightfully simple. Just four bytes changed alters the logic to our requirements.

NEWDOS PATCH FROM 4C3F to 43D4

```
43CF    FE20    CP   20H              : COMPARE ACC WITH 20H (32)
32D1    3002    JR   NC,43D5H         : GOTO PRINT IF MORE
43D3    C640    ADD  A,40H            : ADD 40H (64) TO ACC BEFORE LP
```

This is the changed portion of the code. The only trick remaining is to identify this area on the disk and make the modification.

MODIFYING YOUR NEWDOS DISK.

FIRST MAKE A BACKUP - IN CASE OF A "MUCKUP".....and if you can't manage that - you've read too far.

I won't go into the whys and wherefors of how to use your SUPERZAP - there are many better writers than I doing just that. What follows is the modification positions and replacements only.

1. Run SUPERZAP and press ENTER when the MENU appears.
2. Display Track 0, Sector 6 of your NEWDOS DOS DISK.
3. Type MOD7C.
4. Replace the 80 with 20 in relative byte 7C.
5. Replace the 38 with 30 in relative byte 7D.
6. Overtype the 02 with another 02 at relative byte 7E (it stays).
7. Replace the 3E with C6 in relative byte 7F.
8. Replace the 2E with 40 in relative byte 80.
9. Press ENTER and write the changes to Disk.
10. Pray a little and have a coffee - you dun good!!
11. Boot up with your modified NEWDOS Disk and try a JKL or two.

Well, that wasn't so hard was it? Incidentally, non-lower case users who have a graphics capable printer can do this mod to get the graphics without any problems and certain non-graphic printers do not mind being sent codes above 127 as they have internal rejection of graphic codes. Oh, and it goes without saying that your EPSON must be in the TRS-80 mode for the graphics to be correct. In the normal mode you must add 32 decimal to all TRS graphic codes to make it work correctly - now there's an interesting thought...

All this was written with the help of SCRIPSIT, SCRIPLUS and the invaluable aid of HEXSPELL. The Hardware was by TANDY, TANDON and EPSON. The Disk errors were produced by ROTHMANS (or the ash thereof), and the crazy syntax by BACARDI. The original idea for this article comes from the 'Moaners About NEWDOS Society' (MANS) and it just goes to prove that you only get what you ask for. If you have any queries or require help with any particular problem with your TRS-80 (or TRASH 50H, depending on the reliability) don't hesitate to drop me a line. The postal service to this part of the world is erratic but we get there eventually. Anyway, your problems are always someone else's - so why not come right out with it? (Ken Smith's address is Officers' Mess, Lansab, P.O. Box 897, Muscat, Sultanate of Oman.)

- 0000000000 -

**** SOFTWARE SECTION ****

***** AUSTRALIAN INCOME TAX CALCULATOR - PEACH and CC *****

Orginally published in the February '82 issue, this program has been revised to reflect the changes that seem to take place each year in the calculation of tax payable. The main additions to the program concerns the calculation of any tax rebates. Direct tax rebates are allowed for dependants and basic health insurance and these can be subtracted directly from the tax payable. In addition, the taxation department allows some rebate for

(i) Lump Sum Payments in respect of unused leave after August 1978.

and

(ii) Interest on Commonwealth Government Loans issued before 1 November, 1968.

Naturally, at least one of these needs to be completely confusing as to how the rebate can be calculated, otherwise, the exercise is pointless (!). The interest rebate is straight forward and amounts to 10% for each $1 of interest. The method of calculating the Lump Sum rebate is not quite so simple:

"This rebate applies only if you have shown an amount at Item 7A, Page 2, and your taxable income exceeds $17,894. The rebate is calculated as follows:-

Tax payable on your taxable income as calculated by you above ...
Less the total of: (a) Tax payable on an amount equal to your taxable income less the amount shown at Item 7A, Page 2 (calculated in the same manner as for your taxable income); and

(b) 32 cents for each $1 of $.....*
*Note: Insert here the lesser of - the amount shown Item 7A, Page 2, and
- the amount by which your taxable income exceeds $4195."

The program seems to work (but you can never tell with interpretation of tax laws) and separate listings for each of the colour computers appears at the back of the magazine.

- 0000000000 -

***** 3-D CUBE - PEACH and CC *****

This program presents a high resolution three dimensional diagram of a cube. Perceptually, such a drawing can be viewed in two ways as you will see when you enter the program. It shows both views of the cube and continually switches between them, giving the illusion of movement. The superior BASIC commands available for line drawing are used to improve the speed of drawing the lines and demonstrates the potential for fast animation of graphics.

The cube looks a little mis-shapen due to the way the eye and brain interact in perception. Edges at the back of the cube, although the same length as corresponding front edges, appear to be longer. The brain expects objects further away (or perceived as such) to be of smaller angular size - like the way railway tracks appear to converge at infinity. However, although this was taken into account, the fact that both views of a cube are presented means that some front edges become back edges and vice versa. An interesting point to note for those of you intending to add 3-D to your programs.

As an exercise see if you can add to the program to show the faces of a die. Perhaps remove the broken lines altogether for a better visual effect. In any case, separate listings for each of the colour computers are included at the back.

- 0000000000 -

***** SKYDIVER LII/16K by C. Cranstone *****

The year: 1942

The cold, FRENCH countryside is covered with snow.

Your mission: safely land behind enemy lines. As the DAKOTA transport plane flies over the countryside, you must jump out of the plane and steer your parachute towards your final destination. Use keys 'J' - jump, 'R' - ripcord.

Use keys '<' and '>' to steer to the left or right. When you successfully land without being hit by snipers, you will have won the game.

You must release your chute as close as possible to the ground or you risk being shot down.

You can commit suicide by pressing 'S' if you feel that you cannot continue the mission. If you do, your score = zilch and you lose miserably!

Sometimes, you may get a 'CANDLE' (chute won't open) and be flattened to a pulp when you make contact with the ground.

XxXx = Only known safe landing spot.
TtTt = Trees. Contact fatal!
LlLl = Lake. Sub-zero temperatures.
EeEe = Electricity lines. Danger 9,000 volts!

Your score is calculated on your jump and the time taken to successfully land alive!!!!! It may sound a daunting task! But remember, it's for your home country!!

Good Luck!! Be heroic!!

PLEASE NOTE:

If you are typing this program in from the magazine line 10 "MUST" be entered exactly as it appears in the magazine or, at the very least, the line "MUST" be FILLED with characters of some sort after the REM as this program will POKE a machine language routine into this line as soon as the program is RUN. You should save the program BEFORE you run it.

If you do not do the above, when the program is RUN it will POKE Line 20 instead and you will probably lose the program.

- 0000000000 -

***** JUMBLED PLAYERS LII/16K by Geoff Egel *****

This program is suitable for decoding jumbled names provided that the player's first initial, along with a full stop, is entered with the jumbled surname.

The team name should be added first before typing the team member's name; for example, you could type TORRENSM.TEHWET and the program would return with M.HEWETT-TORRENS

You can even enter it as NRTORSEM.TEHWET and the program will still find it.

The version here is suitable for the South Australian News Tangled Players and all that is necessary is to enter the jumbled letters in any order when requested to do so. If you are unsure of what a particular letter could be, then enter all of the possibilities. To update the data or amend it to suit your State/Country, you should enter data lines in the program using the following format:

TEAM NAME,PLAYER1,PLAYER2, ETC,...,Ø

The 0 (zero) should be typed after the last team member's name, so the above would appear as:

DATA NORWOOD,A.SMITH,B.SMITH,C.SMITH,0

The operation of the program is to check each letter in the data blocks until a team name is found. When a match occurs, the remaining letters are compared one by one to each of the team members until a match occurs. Then the team, along with a surname, is displayed with any remaining jumbled letters displayed to the right.

- 0000000000 -

***** MATRIX MANIPULATION L2/16K (C) Martin Downey *****

FEATURES:
* PRINT, MULT, ADD/SUB, INVERSE commands.
* Handles up to 1Øx1Ø and larger.
* Stand-alone or use in your own programs.

After the program has loaded type in RUN and press ENTER (or NEW LINE). The title will be displayed along with the SIX COMMANDS (MATRIX INSTRUCTIONS). Initially there will be Ø MATRICES ALREADY DEFINED. To use a particular INSTRUCTION simply type in the corresponding number (1-6) and press ENTER (NEW LINE). The SIX commands are described below.

PRINT: This command will print out all the matrices that have been DEFINEd or created using MULT, etc. The matrices will be printed one at a time. Pressing ENTER (NEW LINE) will display the next matrix. Very large matrices (more than 1Ø columns) may wrap around spoiling the display.

DEFINE: Allows you to input the matrices to be manipulated. You will first be asked to input the size of the matrix. type in the number of rows then a comma then the number of columns then press ENTER (NEW LINE). If you only input one number a double question mark appear (??) telling you to input the second number. This sort of error should be avoided as it will mess up the display. After entering the size you type in each element of the matrix followed by ENTER (NEW LINE). The matrix will be displayed in its normal format as the elements are entered. When finished control will return to the SIX COMMANDS.

MULT: Allows multiplication of TWO MATRICES or a MATRIX and a CONSTANT. Normal laws of Matrix multiplication apply and if the two matrices are not of correct dimensions then a 'NOT POSSIBLE' will be displayed. Enter the number of the first matrix. Then enter a 1 or a 2 depending on whether you want to multiply by a constant or another matrix. Finally enter the CONSTANT or MATRIX NUMBER accordingly. The resulting matrix will then be displayed (and is automatically saved with the other matrices).

ADD/SUB: After entering the number of the first matrix you will be asked to enter a 'MULTIPLE OF ADDITION'. Entering a 1 will give straight addition, entering a -1 will give straight subtraction, but you can enter any other number depending on the result you want. Finally enter the number of the SECOND MATRIX and the result will be displayed (and saved). Again normal matrix rules apply.

INVERSE: Enter the matrix number (matrix must be square of course) and after a short delay the inverse will be printed (and saved). If there is no inverse then 'SINGULAR' will be displayed. In either case THE ORIGINAL WILL BE DELETED so you might want to save another copy (use MULT times a constant 1).

DELETE: This command allows you to get rid of matrices that you may not want any more. This may become necessary if you get a message 'INSUFFICIENT MEMORY. TRY DELETE'. (Note: All matrices are stored in the single dimension array A which is dimensioned according to your MEMORY size (line 30). you will have to keep this in mind if you wish to use parts of the program in your own program (see below).)

When you DELETE a matrix all higher-numbered matrices will be moved down one. So if you DELETE matrix number 3, say, then matrix 4 will become 3, matrix 5 will become 4 and so on. You can check on the new matrix numbers using the PRINT command.

All the MATRIX COMMANDS are divided into subroutines so you can use them in your own programs. This will however require some modification so the following may be of help.

80-130	PRINT command.
140-180	DEFINE command.
190-300	MULTI command. Also uses part of PRINT and SUB 660.
310-380	ADD/SUB command. Also uses part of PRINT and SUB 660.
390-650	INVERSE command. Also uses part of PRINT and DELETE and SUB 660.
660-680	SUB used by MULT, ADD/SUB, INVERSE and DELETE.
690-710	DELETE command. Also uses SUB 660.
720	Displays 'INSUFFICIENT MEMORY' flag. Used by MULT and ADD/SUB.

All matrices are stored in the single dimensioned array A. Matrices are stored sequentially from MATRIX 1 up. The first number is the number of ROWS then comes the number of COLUMNS followed by the matrix ELEMENTS. (NOTE: The number of elements equals ROWS * COLUMNS).

SOME MATRIX LAWS:

Take Matrices A and B of size $(r_A \times c_A)$ and $(r_B \times c_B)$ respectively.

Then AxB is only valid if $c_A = r_B$. Also the resultant matrix C=AxB will have size $(r_A \times c_B)$.

For addition, A+B is only valid if $r_A = r_B$ and $c_A = c_B$. The resultant C=A+B has the same size as A and B.

If a matrix is square and NON-SINGULAR then there exists an inverse matrix such that:

$$A \times (A)^{-1} = I = (A)^{-1} \times A$$ where I is the IDENTITY matrix.

NOTE: In general AxB ≠ BxA.

IDENTITY: I = (variable size)

```
1 0 0 0 ... 0
0 1 0 0 ... 0
0 0 1 0 ... 0
.............
0 0 0 0 ... 1
```

ZERO MATRIX: Z = (variable size)

```
0 0 0 0 ... 0
0 0 0 0 ... 0
0 0 0 0 ... 0
.............
0 0 0 0 ... 0
```

A+Z = A = Z+A AxZ = Z = ZxA

- 0C00000000 -

***** RESET Disk 32K or 48K by Dennis Bareis *****

This is a program that will reset your computer. It allows you to reboot by holding the space-bar and the clear keys down together.

To run the program from DOS type "RESET/CMD". I use "AUTO RESET/CMD" to load it on reset or power up.

I decided to use two keys to reset the computer so as to minimize the chances of an accidental reboot, and not to use the BREAK key to decrease the size of the code required. (You would need code to bypass the ROM test for the BREAK key).

When I write a program I try to make it compatible with as many types of systems as possible. For this reason the program looks at the top of memory pointer and relocates below this point, and before patching in my program I get the addresses of any existing drivers and patch them into my code. This means that you can still use lower case drivers etc. (as long as you load them before this program).

The program patches into both the keyboard scanning routine and the interrupt driven clock. This means that it will reboot the computer from almost anywhere.

I have noticed that some programs patch into the keyboard by placing a zero at 4015H (H for Hex) and then putting the address of their own program at 4034H as well as a jump instruction at 4033H, while others just load the address of their program into 4016H. The initialization at lines 340-360 tests for each of the above situations, and then loads the driver (from 4016H or 4033H) and patches this address into my "CALL DUMMY" instruction in line 690 (so it becomes "CALL DRIVER").

From line 410 to 450 the program looks at the top of memory pointer, and moves it down in RAM so that it can fit in between the new pointer position and the old one. The start of this program is to be at the top of memory pointer plus one, so I add one to the HL register pair in line 460.

Since there are two ways of intercepting the keyboard, I make sure that 4015H contains a "one" (lines 470-480). I then patch in my program at line 490.

As different DOS's have their clock routine at different places, and to allow for any previous patches to the clock, the program gets the address of the clock from 4013H and patches this into itself (lines 510-520).

The program now relocates to high memory (lines 530-550).

It must now find the position of INTER relative to the start of the program so that it can work out the new location of INTER (lines 560-580), and is then patched into the clock (line 590). It then returns to DOS.

Since the program adjusts the location of the clock, I have bracketed the initialization program with DI and EI (Disable/Enable Interrupts).

The main program (lines 630-650 and 730-750) tests to see if the space-bar and clear keys are both down. If they are, it will jump to RES (for reset).

I noticed that whenever I tried to use this program while the disk drive was idling (before turning off) the computer would crash. For this reason I include lines 660-670.

The program will now reset almost every time even during disk I/O. (To be safe, wait for the disk drive to stop).

I used an ORIGIN of 9000H so the program (before relocation) will sit above any low memory and below any high memory routines.

HOW TO LOAD.

From any DOS type RESET and press ENTER/NEWLINE

The program is now active and will REBOOT the system if the CLEAR key and SPACE BAR are pressed together. As the program reboots the system, it is a good idea to use:

AUTO RESET/CMD on your disk as this will restart the RESET program each time after a BOOT.

- 0000000000 -

***** MAGIC CUBE SOLVER LII/16K by R.J. Burling *****

This program uses a step-by-step method for solving your cube. When a choice is possible the computer simply asks for your choice and then proceeds. This is made possible by the use of INKEY$ and ON xxx GOTO functions, allowing the computer to proceed to the portion of the program of your choice. Here the next set of instructions, as relevant to the operator's choice, is printed up on the screen or else a new set of choices.

The program is based on a set colour pattern. The colour pairs assumed by the computer (that is, opposite sides as per the centre cube) are BLUE/WHITE, ORANGE/RED, YELLOW/GREEN. If the operator's cube has a different combination of colours you will need to change the relevant colour names within the program.

- 0000000000 -

VIEWS OF DISPLAY
FOR MAGIC CUBE SOLVER

A SINGLE FACE OF THE CUBE SHOWING NAMES.

CORNER	EDGE	CORNER
EDGE	CENTRE	EDGE
CORNER	EDGE	CORNER

HIT ANY KEY
TO RETURN.

THE CUBE POSITION NUMBERS FOR THE MOVEMENT SEQUENCES BELOW

1	2	3
4	5	6
7	8	9

1. 1 TO 3 : 3 TO 7 : 7 TO 1
2. 1 TO 7 : 7 TO 3 : 3 TO 1
3. 1 TO 3 : 3 TO 1
4. 3 TO 7 : 7 TO 3
5. TO PROCEED TO - EDGE CUBES

THE CUBE POSITION NUMBERS FOR THE MOVEMENT SEQUENCES BELOW

1	2	3
4	5	6
7	8	9

1. 2 TO 8 : 8 TO 4: 4 TO 2
2. 2 TO 4 : 4 TO 8 : 8 TO 2
3. 2 TO 8 AND 8 TO 2 : 4 TO 6 AND 6 TO 4
4. 2 TO 6 AND 6 TO 2 : 4 TO 8 AND 8 TO 4
5. TO POSITION COLOURS ONCE ALL THIS LAYER IS LOCATED.

****** AUST. TAX CALCULATOR ******
COLOR COMPUTER

```
1  REM * COPYRIGHT BY DAVID WEAV
ER
2 REM * 11 PINE GROVE SALE VICTO
RIA 3850  051 445019
3 REM * 03 APRIL 1981
10 REM * PROGRAM TO COMPUTE NET
INCOME AND TAX PAYABLE
20 DIM LL(4),TD(4),TC(4)
30 A$="$$########.##"
40 CLS
50 PRINT"PERSONAL INCOME AND TAX
 PAYABLE"
60 REM * COMPUTES NET INCOME
70 PRINT:PRINT:INPUT"WHAT IS YOU
R GROSS INCOME FROM  GROUP CERTI
FICATE  ";GC
80 PRINT:INPUT"ANY OTHER INCOME
  ";OI
90 PRINT:PRINT "YOUR TOTAL INCOM
E IS $";GC+OI
100 PRINT:INPUT"WHAT ARE YOUR TO
TAL DEDUCTIONS   ";DD
110 TI =(GC+OI)-DD
120 PRINT:PRINT"YOUR NET TAXABLE
 INCOME IS       $";TI
130 PRINT:INPUT"WHAT IS THE TAX
PAID FROM YOUR  GROUP CERTIFICAT
E  ";TP
140 REM * PROGRAM TO FIND TAX PA
YABLE ON NET INCOME
150 REM * LL STORES LOWER LIMIT
OF TAX SCALE * TD STORES TAX
      DOLLAR ON LOWER LIMIT
160 REM * TC STORES CENTS IN THE
 DOLLAR ON NET INCOME OVER THE
    LOWER LIMIT
170 REM * INPUTS TAX SCALES
180 REM * SEARCH FOR FIRST VALUE
 IN LIST LL WHICH EXCEEDS TI
190 FOR I=1 TO 4
200 READ LL(I),TD(I),TC(I)
210 IF LL(I) < TI THEN J=I
220 NEXT I
230 REM * PRINT OUT INCOME TAX P
AYABLE ON NET INCOME TI
240 IT = TD(J)+TC(J)*(TI-LL(J))
250 REM * CALCULATE ANY REBATES
260 PRINT:INPUT"TOTAL REBATE CLA
IMED AT ITEM 32  ";RB
270 IF TI < 17895 THEN 320
280 PRINT:INPUT"LUMP SUM PAYMENT
 AT ITEM 7A       ";LS:IF LS<1 TH
EN 320
290 TX=TI-LS:IF (TI-4195)<LS THE
N LS=TI-4195
300 IF TX<LL(J) THEN J=J-1:GOTO3
00
310 RB= INT((RB+IT-(TD(J)+TC(J)*
(TX-LL(J))+0.32*LS))*100+.5)/100
320 PRINT:INPUT"INTEREST ON COMM
. GOVT. LOAN AT ITEM 9 ";RX:RB=
INT((RB+.1*RX)*100+.5)/100
330 IT=IT-RB:IF IT<0 THEN IT=0
340 PRINT:PRINT"INCOME TAX PAYAB
LE $";IT
350 IF IT > TP THEN 390
360 REM * PRINTS OUT WHO OWES WH
O
370 PRINT:PRINT "THE TAX DEPARTM
ENT < OWES YOU >  $";INT(((TP-IT
)+.005)*100)/100
380 GOTO400
390 PRINT:PRINT"< YOU OWE > THE
TAX DEPARTMENT   $";INT(((IT-TP)
+.005)*100)/100
```

```
400 INPUT"DO YOU WANT TO CONTINU
E   ";Z$
410 IF LEFT$(Z$,1)="Y" THEN REST
ORE:GOTO40 ELSE CLS:END
420  REM * IF TAX SCALES CHANGE
ALTER VALUES IN DATA LINES
430 DATA 1,0,0
440 DATA 4195,0,0.32
450 DATA 17894,4383.68,0.46
460 DATA 35788,12614.92,0.6
```

***** 3-D CUBE *****
COLOR COMPUTER

```
10 REM * MICRO-80 DEMO
20 CLS:PRINT@270,"CUBE":PRINT@30
2,"====="
30 PRINT@480,"HIT ANY KEY..."
40 A$=INKEY$:IF A$="" THEN 40
50 CLS
60 PMODE 1,1
70 PCLS
80 X1=58:X2=98:X3=158:X4=198
90 Y1=26:Y2=66:Y3=126:Y4=166
100 DT=1000:REM TIME DELAY BETWE
EN CHANGES
110 SCREEN 1,1
120 REM THE CUBE
130 LINE (X1,Y1)-(X3,Y1),PSET
140 LINE (X1,Y1)-(X2,Y2),PSET
150 LINE (X1,Y1)-(X1,Y3),PSET
160 LINE (X2,Y4)-(X1,Y3),PSET
170 LINE (X2,Y4)-(X2,Y2),PSET
180 LINE (X2,Y4)-(X4,Y4),PSET
190 LINE (X4,Y2)-(X3,Y1),PSET
200 LINE (X4,Y2)-(X2,Y2),PSET
210 LINE (X4,Y2)-(X4,Y4),PSET
220 LINE (X2,Y2)-(X4,Y2),PSET
230 LINE (X2,Y2)-(X2,Y4),PSET
240 LINE (X2,Y2)-(X1,Y1),PSET
250 XA=X3:YA=Y3:XB=X1:GOSUB370
260 XA=X3:YA=Y3:YB=Y1:GOSUB430
270 XA=X3:YA=Y3:XB=X4:YB=Y4:GOSU
B490
280 FOR I=1 TO DT:NEXT I
290 XA=X2:YA=Y2:XB=X4:F$="R":GOS
UB370
300 LINE (X3,Y3)-(X1,Y3),PSET
310 XA=X2:YA=Y2:YB=Y4:GOSUB430
320 LINE (X3,Y3)-(X3,Y1),PSET
330 XA=X2:YA=Y2:XB=X1:YB=Y1:GOSU
B490
340 LINE (X3,Y3)-(X4,Y4),PSET
350 FOR I=1 TO DT:NEXT I
360 A$=INKEY$:IF A$="" THEN 220
ELSE END
370 IF XA>XB THEN DX=-5 ELSE DX=
5
380 IF F$="R" THEN 410
390 LINE (XA,YA)-(XA+DX,YA),PSET
:XA=XA+3*DX
400 IF ABS(XA-XB)>5 THEN 390 ELS
E RETURN
410 LINE (XA+2*DX,YA)-(XA+3*DX,Y
A),PRESET:XA=XA+4*DX:PSET(XB,YA)
420 IF ABS(XA-XB)>5 THEN 410 ELS
E RETURN
430 IF YA>YB THEN DY=-5 ELSE DY=
5
440 IF F$="R" THEN 470
450 LINE(XA,YA)-(XA,YA+DY),PSET:
YA=YA+3*DY
460 IF ABS(YA-YB)>5 THEN 450 ELS
E RETURN
470 LINE (XA,YA+DY)-(XA,YA+2*DY)
,PRESET:YA=YA+3*DY:PSET(XA,YB)
480 IF ABS(YA-YB)>5 THEN 470 ELS
E RETURN
490 IF XA>XB THEN DX=-5 ELSE DX=
5
500 IF YA>YB THEN DY=-5 ELSE DY=
5
510 IF F$="R" THEN 540
520 LINE (XA,YA)-(XA+DX,YA+DY),P
SET:XA=XA+2*DX:YA=YA+2*DY
530 IF ABS(XA-XB)>5 THEN 520 ELS
E RETURN
540 LINE (XA+2*DX,YA+2*DY)-(XA+3
*DX,YA+3*DY),PRESET:XA=XA+3*DX:Y
A=YA+3*DY:PSET(XB,YB)
550 IF ABS(XA-XB)>5 THEN 540 ELS
E RETURN
```

**** AUST. TAX CALCULATOR ****
HITACHI PEACH

```
1 REM * COPYRIGHT BY DAVID WEAVER
2 REM * 11 PINE GROVE SALE VICTORIA 3850
  051 445019
3 REM * 03 APRIL 1981
10 REM * PROGRAM TO COMPUTE NET INCOME A
ND TAX PAYABLE
20 DIM LL(4),TD(4),TC(4)
30 A$="$$#########.##
40 CLS
50 PRINT TAB(25)"PERSONAL INCOME AND TAX
 PAYABLE";
60 REM * COMPUTES NET INCOME
70 PRINT:PRINT:INPUT"WHAT IS YOUR GROSS
INCOME FROM GROUP CERTIFICATE  ";GC
80 PRINT:INPUT"ANY OTHER INCOME ?
                                ";OI
90 PRINT:PRINT "YOUR TOTAL INCOME IS
                                 ";GC+OI
100 PRINT:INPUT"WHAT ARE YOUR TOTAL DEDU
CTIONS                          ";DD
110 TI =(GC+OI)-DD
120 PRINT:PRINT"YOUR NET TAXABLE INCOME
IS                               ";TI
130 PRINT:INPUT"WHAT IS THE TAX PAID FRO
M YOUR GROUP CERTIFICATE  ";TP
140 REM * PROGRAM TO FIND TAX PAYABLE ON
 NET INCOME
150 REM * LL STORES LOWER LIMIT OF TAX S
CALE * TD STORES TAX           DOLLAR ON
LOWER LIMIT
160 REM * TC STORES CENTS IN THE DOLLAR
ON NET INCOME OVER THE       LOWER LIMIT
170 REM * INPUTS TAX SCALES
180 REM * SEARCH FOR FIRST VALUE IN LIST
 LL WHICH EXCEEDS TI
190 FOR I=1 TO 4
200 READ LL(I),TD(I),TC(I)
210 IF LL(I) < TI THEN J=I
220 NEXT I
230 REM * PRINT OUT INCOME TAX PAYABLE O
N NET INCOME TI
240 IT = TD(J)+TC(J)*(TI-LL(J))
250 REM * CALCULATE ANY REBATES
260 PRINT:INPUT"TOTAL REBATE CLAIMED AT
ITEM 32                         ";RB
270 IF TI < 17895 THEN 320
280 PRINT:INPUT"LUMP SUM.PAYMENT AT ITEM
 7A                             ";LS:IF LS<1 T
HEN 320
290 TX=TI-LS:IF (TI-4195)<LS THEN LS=TI-
4195
300 IF TX<LL(J) THEN J=J-1:GOTO300
310 RB=INT((RB+IT-(TD(J)+TC(J)*(TX-LL(J)
)+0.32*LS))*100+.5)/100
320 PRINT:INPUT"INTEREST ON COM. GOVT. L
OAN AT ITEM 9                   ";RX:RB=INT((R
B+.1*RX)*100+.5)/100
330 IT=IT-RB:IF IT<0 THEN IT=0
340 PRINT:PRINT"INCOME TAX PAYABLE
                                 ";IT
```

```
350 IF IT > TP THEN 390
360 REM * PRINTS OUT WHO OWES WHO
370 PRINT:PRINT"THE TAX DEPARTMENT < * O
WES YOU * >                    ";INT(((TP-IT
)+.005)*100)/100
380 GOTO400
390 PRINT:PRINT"< * YOU OWE * > THE TAX
DEPARTMENT                     ";INT(((IT-TP
)+.005)*100)/100
400 INPUT"DO YOU WANT TO CONTINUE    ";Z$
410 IF LEFT$(Z$,1)="Y" THEN RESTORE:GOTO
40 ELSECLS:END
420 REM * IF TAX SCALES CHANGE ALTER VAL
UES IN DATA LINES
430 DATA 1,0,0
440 DATA 4195,0,0.32
450 DATA 17894,4383.68,0.46
460 DATA 35788,12614.92,0.6
```

****** 3-D CUBE ******
HITACHI PEACH

```
10 REM * MICRO-80 DEMO
20 CLS:LOCATE37,11:PRINT"CUBE":LOCATE37,
12:PRINT"===="
30 LOCATE7,18:PRINT"HIT ANY KEY..."
40 A$=INKEY$:IF A$="" THEN 40
50 CLS
60 X1=200:X2=240:X3=400:X4=440
70 Y1=26:Y2=66:Y3=126:Y4=166
80 DT=1000:REM TIME DELAY BETWEEN CHANGE
S
90 REM THE CUBE
100 LINE (X1,Y1)-(X3,Y1),PSET
110 LINE (X1,Y1)-(X2,Y2),PSET
120 LINE (X1,Y1)-(X1,Y3),PSET
130 LINE (X2,Y4)-(X1,Y3),PSET
140 LINE (X2,Y4)-(X2,Y2),PSET
150 LINE (X2,Y4)-(X4,Y4),PSET
160 LINE (X4,Y2)-(X3,Y1),PSET
170 LINE (X4,Y2)-(X2,Y2),PSET
180 LINE (X4,Y2)-(X4,Y4),PSET
190 LINE (X2,Y2)-(X4,Y2),PSET
200 LINE (X2,Y2)-(X2,Y4),PSET
210 LINE (X2,Y2)-(X1,Y1),PSET
220 XA=X3:YA=Y3:XB=X1:GOSUB340
230 XA=X3:YA=Y3:YB=Y1:GOSUB400
240 XA=X3:YA=Y3:XB=X4:YB=Y4:GOSUB460
250 FOR I=1 TO DT:NEXT I
260 XA=X2:YA=Y2:XB=X4:F$="R":GOSUB340
270 LINE (X3,Y3)-(X1,Y3),PSET
280 XA=X2:YA=Y2:YB=Y4:GOSUB400
290 LINE (X3,Y3)-(X3,Y1),PSET
300 XA=X2:YA=Y2:XB=X1:YB=Y1:GOSUB460
310 LINE (X3,Y3)-(X4,Y4),PSET
320 FOR I=1 TO DT:NEXT I
330 A$=INKEY$:IF A$="" THEN 190 ELSE END
340 IF XA>XB THEN DX=-5 ELSE DX=5
350 IF F$="R" THEN 380
360 LINE (XA,YA)-(XA+DX,YA),PSET:XA=XA+3
*DX
370 IF ABS(XA-XB)>5 THEN 360 ELSE RETURN
380 LINE (XA+2*DX,YA)-(XA+3*DX,YA),PRESE
T:XA=XA+4*DX:PSET(XB,YA)
390 IF ABS(XA-XB)>5 THEN 380 ELSE RETURN
400 IF YA>YB THEN DY=-5 ELSE DY=5
410 IF F$="R" THEN 440
420 LINE(XA,YA)-(XA,YA+DY),PSET:YA=YA+3*
DY
430 IF ABS(YA-YB)>5 THEN 420 ELSE RETURN
440 LINE (XA,YA+DY)-(XA,YA+2*DY),PRESET:
YA=YA+3*DY:PSET(XA,YB)
450 IF ABS(YA-YB)>5 THEN 440 ELSE RETURN
460 IF XA>XB THEN DX=-5 ELSE DX=5
470 IF YA>YB THEN DY=-5 ELSE DY=5
480 IF F$="R" THEN 510
490 LINE (XA,YA)-(XA+DX,YA+DY),PSET:XA=X
A+2*DX:YA=YA+2*DY
500 IF ABS(XA-XB)>5 THEN 490 ELSE RETURN
510 LINE (XA+2*DX,YA+2*DY)-(XA+3*DX,YA+3
*DY),PRESET:XA=XA+3*DX:YA=YA+3*DY:PSET(X
B,YB)
520 IF ABS(XA-XB)>5 THEN 510 ELSE RETURN
```

```
10 GOTO20:REM ** Skydiver/bas ** Written by C. Cranstone (C) **
              ** December 1981: compatible with Trs-80 both  **
              ** Models and System-80 computers.              **
              ** Works with any Dos,Disk basic
20 CLS:TT=PEEK(16549)*256+PEEK(16548):TA=TT+59:M=TT+10
30 M1=INT(M/256):M2=INT(M-M1*256):FORL=MTOM+47:READZ:POKEL,Z:NEX
T:IFTT<>17129ANDTT<>17385THENDEFUSR0=M:CMD"T"ELSEPOKE16526,M2:POK
E16527,M1
40 DATA205,127,10,229,221,225,221,78,0,121,183,200,221,70,1,62,5
,211,255,16,254,221,70,1,62,6,211,255,16,254,13,32,235,221,35,221
,35,1,255,255,33,48,0,9,56,253,24,214
50 TA=PEEK(16549)*256+PEEK(16548)+59
60 POKETA,100:POKETA+1,100:POKETA+2,0:C=USR(TA)
70 CLS:CLEAR2000:GOSUB790:GOSUB1000
```

```
80 O=0:P=0:CLS:IN=0:F$="":GOSUB1090:SC=0:C=0:TA=PEEK(16549)*256+
PEEK(16548)+59
90 GOTO130
100 POKETA,56:POKETA+1,56:POKETA+2,43:POKETA+3,43:POKETA+4,0:C=U
SR(TA)
110 FORT%=P%-2TOP%+2:PRINT@T%-63,CHR$(RND(63)+129);:PRINT@T%-1,C
HR$(RND(63)+129):NEXTT%:PRINT@T%-128-(T%-P%),"S p l a t !";:PRINT
@0,CHR$(143);STRING$(62,141);CHR$(143);:PRINT@128,"You were flatt
ened to a pulp!";:PRINT@192,"Play again?";:GOTO770
120 FORJ=1TO200:NEXTJ:CLS:PRINTCHR$(143);STRING$(62,133);CHR$(14
3);:PRINT"You have lost!":PRINT"The plane ran out of fuel!!":PRIN
T:PRINT"Play again?";:GOTO770
130 FORT%=1TO54:PRINT@T%," ";P1$;:FORJ=1TO12:NEXTJ:POKETA,120:PO
KETA+1,1:POKETA+2,0:C=USR(TA)
140 IFINKEY$="J"THEN150ELSEPRINT@T%," ";P2$;:FORJ=1TO12:NEXTJ:NE
XTT%:GOTO120
150 FORY%=T%TO50STEP2:PRINT@Y%,"  ";P1$;:PRINT@Y%,"  ";P2$;:NEXT
160 FORP%=T%TOT%+832STEP64:PRINT@P%,M$(1);:ZX=1:GOSUB170:PRINT@P
%,M$(2);:ZX=2::GOSUB170:PRINT@P%,M$(3);:ZX=3:GOSUB170:PRINT@P%,M$
(4);:ZX=4:GOSUB170:PRINT@P%,M$(5);:ZX=5:GOSUB170:PRINT@P%,M$(6);:
ZX=6:GOSUB170:NEXTP%:GOTO100
170 IFINKEY$="R"THENAS=P%:GOTO200ELSEIFINKEY$="S"THEN480
180 IFPEEK(Y%+15424)>=129ANDPEEK(Y%+15424)<=191THEN100
190 PRINT@0,CHR$(31);:C=USR(TA):RETURN
200 RANDOM:GOSUB500:IFRND(10)=2THEN250ELSE210
210 CLS:PRINT@P%-1,C1$;:FORH=1TO90:NEXTH:CLS:PRINT@P%-1,C2$;:FOR
H=1TO90:NEXTH:FORR%=P%-1TO832STEP64:ZX=1:GOSUB270:PRINT@R%+IN,P$(
1);:GOSUB270:PRINT@R%+IN,P$(2);:ZX=2:GOSUB270:PRINT@R%+IN,P$(3);:
ZX=3:GOSUB270:NEXTR%:PRINT@0,"Safe landing!!";
220 POKETA,89:POKETA+1,89:POKETA+2,56:POKETA+3,56:POKETA+4,0:C=U
SR(TA):GOTO230
230 R%=R%+128:IFR%+IN+64>=1023THENR%=R%-128ELSEIFX>=LEN(GR$)-62T
HENX=1
240 PRINT@65,CHR$(31);:PRINT@960,MID$(GR$,X,62);:PRINT@R%+IN+64,
C4$;:FORK=1TO600:NEXTK:GOTO730
250 FORT%=TATOTA+9:POKET%,O+1:O=O+5:NEXT:POKETA+10,0:C=USR(TA):I
FX=0THENX=1ELSEIFX>=LEN(GR$)-62THENX=1
260 PRINT@0,"Oh no! A candle!!!";:PRINT@P%-1,C1$;:PRINT@P%-1,CHR
$(31);:FORR%=P%-1TO832STEP64:PRINT@R%,CC$;:FORK=1TO90:NEXTK:PRINT
@0,CHR$(31);:PRINT@960,MID$(GR$,X,62);:NEXTR%:P%=R%:GOTO100
270 IFPEEK(14368)=16THENIN=IN-1:GOSUB440:SC=SC+RND(40):FORT%=64T
0896STEP64:PRINT@T%,CHR$(30);:NEXT:PRINT@R%+IN,P$(ZX);
280 IFPEEK(14368)=64THENIN=IN+1:GOSUB450:SC=SC+RND(40):FORT%=64T
0896STEP64:PRINT@T%,CHR$(30);:NEXT:PRINT@R%+IN,P$(ZX);
290 X=X+5:IFX>=LEN(GR$)-62THENX=0ELSEPRINT@961,MID$(GR$,X,62);
300 IFR%+15360+322+IN>=16383THENR%=R%-64ELSEKO=PEEK(R%+15360+322
+IN)
310 IFKO=ASC("E")THEN650ELSEIFKO=ASC("T")THEN630
320 IFKO=ASC("L")THEN670
330 IFKO=ASC("X")THENPRINT@0,"Safe landing!";:FORJ=1TO150:NEXTJ:
GOTO230
340 IFR%+322+IN>=1023THEN710
350 IFINKEY$="S"THEN480
360 G1=RND(959)+15360:IFPEEK(G1)<>32THEN400ELSEPOKEG1-1,ASC("*")
:POKEG1,ASC("."):POKEG1+1,ASC("*")
370 POKEG1-1,32:POKEG1,32:POKEG1+1,32
380 WO=R%+IN-65:IFWO<=-1THEN690ELSEPRINT@R%+IN-65,E2$;
390 RETURNELSERETURN
400 POKEG1-1,ASC("!"):POKEG1,ASC("."):POKEG1+1,ASC("!")
410 POKETA,67:POKETA+1,67:POKETA+2,23:POKETA+3,23:POKETA+4,0:C=U
SR(TA)
420 PRINT@0,CHR$(30);:FORJ=1TO100:NEXTJ:FORY%=(R%+IN)TO832STEP64
:PRINT@Y%,P$(1);:PRINT@Y%-1,E2$;:PRINT@960,MID$(GR$,X,62);:NEXT:P
RINT@Y%+64,C4$;:FORK=1TO300:NEXTK:CLS:PRINTCHR$(143);STRING$(62,1
39);CHR$(143);:PRINT:PRINT:PRINT"You were shot down!":PRINT
430 PRINT"You were captured by enemy soldiers and sentenced to":
PRINT"slow torture!":PRINT:PRINT"Play again";:GOTO770
440 FORT%=64TO896STEP64:IFPEEK(T%+15360)<>32THENIN=IN+2:RETURNEL
SENEXT:RETURN
450 FORT%=127TO959STEP64:IFPEEK(T%+15360)<>32THENIN=IN-2:RETURNE
LSENEXT:RETURN
460 CLS:PRINT@960,MID$(GR$,X,62);:PRINT@R%+64+IN,P$(1);:FORJ=1TO
70:NEXTJ:CLS:PRINT@960,MID$(GR$,X,62);:PRINT@R%+256+IN,C4$;:FORJ=
1TO80:NEXTJ
470 RETURNELSERETURN
480 FORT%=TATOTA+55STEP2:POKET%,15:POKET%+1,255-II:II=II+2:NEXTT
%:POKETA+56,0:C=USR(TA)
490 PRINT@0,"You've taken your suicide pill!";:FORJ=1TO300:NEXTJ
:CLS:PRINTCHR$(143);STRING$(62,137);CHR$(143);:PRINT@128,"You hav
e committed suicide!";:PRINT@320,"You died of cyanide poisoning!"
;:PRINT@448,"Play again";:GOTO770
500 POKETA,12:POKETA+1,12:POKETA+2,0:C=USR(TA):ONINT(AS/64)GOTO5
10,520,530,540,550,560,570,580,590,600,610,620
510 SC=1000:RETURN
520 SC=2000:RETURN
530 SC=3000:RETURN
540 SC=4000:RETURN
550 SC=5000:RETURN
560 SC=6000:RETURN
570 SC=7000:RETURN
580 SC=8000:RETURN
590 SC=9000:RETURN
600 SC=10000:RETURN
610 SC=11000:RETURN
620 SC=12000:RETURN
630 POKETA,43:POKETA+1,43:POKETA+2,21:POKETA+3,21:POKETA+4,0:C=U
SR(TA):GOSUB460
640 PRINT@0,"Crump!!!!";:FORJ=1TO350:NEXTJ:CLS:PRINTCHR$(143);ST
RING$(62,170);CHR$(143);:PRINT@128,"You crashed into trees and br
oke your neck!":PRINT:PRINT"Can't you do anything right??":PRINT:
PRINT"Play again";:GOTO770
650 POKETA,70:POKETA+1,70:POKETA+2,25:POKETA+3,25:POKETA+4,0:C=U
SR(TA):GOSUB460
660 PRINT@0,"Zap!!!!";:FORJ=1TO350:NEXTJ:CLS:PRINTCHR$(143);STRI
NG$(62,139);CHR$(143);:PRINT@128,"You got entangled in power line
s!":PRINT:PRINT"The way you play this game is shocking!":PRINT:PR
INT"Play again";:GOTO770
670 POKETA,40:POKETA+1,40:POKETA+2,60:POKETA+3,40:POKETA+4,0:C=U
SR(TA):GOSUB460
```

```
680 PRINT@0,"Blub blub blub!";:FORJ=1TO350:NEXT:CLS:PRINTCHR$(14
3);STRING$(62,179);CHR$(143);:PRINT@128,"You landed in a lake and
 froze into a solid icicle!":PRINT:PRINT"You drowned in the lake
first though!":PRINT:PRINT"Try again";:GOTO770
690 FORT%=TATOTA+9STEP2:POKET%,15:POKET%+1,RND(100):NEXT:POKETA+
10,0:C=USR(TA):GOSUB460
700 PRINT@0,"Rip!";:FORJ=1TO100:NEXTJ:CLS:PRINTCHR$(143);STRING$
(62,RND(63)+129);CHR$(143);:PRINT@128,"Your chute got tangled up
with the plane!":PRINT:PRINT"You pulled the ripcord too early!":P
RINT:PRINT"Try again";:GOTO770
710 GOSUB460
720 PRINT@0,CHR$(143);STRING$(62,134);CHR$(143);:PRINT@128,"Crum
p! Hard ground!":PRINT"Play again";:GOTO770
730 POKETA,39:POKETA+1,39:POKETA+2,48:POKETA+3,48:POKETA+4,0:FOR
T%=1TO10:C=USR(TA):NEXT
740 IFSC>HSTHENHS=SC
750 CLS:PRINTCHR$(143);STRING$(62,134);CHR$(143);:PRINT@128,"You
 have won!!!!!!":PRINT"You will be remembered for your gallant co
nduct!!":PRINT
760 PRINT"Your final score was ";SC:PRINT:PRINT"The highest scor
e to date is ";HS:PRINT:PRINT"Play again?";:GOTO770
770 INPUTQ$:IFLEFT$(Q$,1)="Y"THEN80ELSECLS:PRINTCHR$(143);STRING
$(62,135);CHR$(143);:PRINT"** SKYDIVING (C) C. Cranstone. Dec. '8
1  All rights reserved **":PRINTCHR$(143);STRING$(62,139);CHR$(14
3):PRINT:PRINT:FORT%=TATOTA+99STEP2
780 POKET%,25:POKET%+1,P:P=P+2:NEXT:POKETA+100,0:C=USR(TA):END
790 RESTORE:FORT%=1TO48:READA:NEXT:FORT%=15438TO15856:READA%:POK
ET%,A%:NEXT:PRINT@540,CHR$(131);CHR$(131);STRING$(3,191);CHR$(131
);CHR$(131);:PRINT@603,CHR$(176);CHR$(191);CHR$(131);CHR$(131);"
";CHR$(131);CHR$(131);CHR$(191);CHR$(176);
800 FORT%=15963TO15972:L1$=L1$+CHR$(PEEK(T%)):NEXT:L2$=CHR$(32)+
CHR$(136)+CHR$(191)+CHR$(147)+" "+CHR$(163)+CHR$(191)+CHR$(132)+"
 "+CHR$(26)+STRING$(9,24)+"  "+CHR$(130)+CHR$(143)+" "+CHR$(143)+
CHR$(129)+"  "
810 PRINT@832,CHR$(30);:PRINT@640,CHR$(30);:PRINT@603,L1$;:PRINT
@768+10,"Press 'I' for instructions, 'ENTER' for game!";:IFPEEK(1
5359)<>0THENK=PEEK(15359):GOTO830ELSEFORJ=1TO80:NEXTJ
820 PRINT@603,L2$;:PRINT@832+28,"Geronimo!";:IFPEEK(15359)<>0THE
NK=PEEK(15359):GOTO830ELSEFORJ=1TO80:NEXTJ:GOTO810
830 IFK=20RK=3THEN850ELSEIFK=1THENRETURN
840 GOTO810
850 CLS:PRINT@0,STRING$(64,134);:PRINT@896,STRING$(64,164);:FORT
%=1TO14:PRINT@T%*64-1,CHR$(191);CHR$(191);:NEXTT%:PRINT@959,CHR$(
191);:PRINT@0,CHR$(191);
860 PRINT@66,"The year:  1942.";:PRINT@130,"The cold, FRENCH cou
ntry side is covered with snow.";:PRINT@194,"Your mission: safely
 land behind enemy lines.";
870 PRINT@386,"As the DAKOTA transport plane flys over the count
ryside, you";:PRINT@450,"must jump out of the plane and steer you
r parachute towards";:PRINT@514,"your final destination.  Use key
s 'J' - jump.";
880 PRINT"'R' =ripcord";
890 PRINT@578,"Use keys '<' & '>' to steer to the left or right.
";:PRINT@642,"When you successfully land without being hit by sni
pers,";:PRINT@706,"You will have won the game.";
900 PRINT@770,"You must release your chute as close as possible
to the ";:PRINT@834,"ground or you risk being shot down.";
910 PRINT@921,CHR$(188);:PRINT@929,CHR$(188);:IFPEEK(15359)<>0TH
EN920ELSEPRINT@922,STRING$(7,32);:FORJ=1TO20:NEXTJ:PRINT@922,"<En
ter>";:FORJ=1TO20:NEXTJ:GOTO910
920 FORT%=66TO834STEP64:PRINT@T%,STRING$(60,32);:NEXT
930 PRINT@66,"You can commit suicide by pressing 'S' if you feel
 that you";:PRINT@130,"cannot continue the mission. If you do, yo
ur score = zilch";:PRINT@194,"and you lose miserably!";:PRINT@258
,"Sometimes, you may get a 'CANDLE' (chute won't open) and";
940 PRINT" be";:PRINT@322,"flattened to a pulp when you make con
tact with the ground";
950 PRINT@379,".";
960 PRINT@450,"XxXx = Only known safe landing spot.";:PRINT@450+
64,"TtTt = Trees. Contact fatal!";:PRINT@450+128,"L1L1 = Lake. Su
b-zero temperatures";:PRINT@450+192,"EeEe = Electricity lines. Da
nger 9,000 volts!";
970 PRINT@706,"Your score is calculated on your jump and the tim
e taken";:PRINT@770,"to successfully land alive!!!!!!  It may so
und a";:PRINT@834,"daunting task! But remember, it's for your hom
e country!!";
980 PRINT@921,CHR$(188);:PRINT@929,CHR$(188);:IFPEEK(15359)<>0TH
EN990ELSEPRINT@922,STRING$(7,32);:FORJ=1TO20:NEXTJ:PRINT@922,"<En
ter>";:FORJ=1TO20:NEXTJ:GOTO980
990 CLS:PRINT:PRINT:PRINTCHR$(23)"      G o o d  l u c k ! !":PR
INT:PRINT:PRINT"      B e  h e r o i c !";:FORJ=1TO1000:NEXTJ:RET
URN
1000 CLS:PRINT@448,CHR$(23);"       ** Initialising **";
1010 RESTORE:FORT%=0TO48:READA:NEXT:FORT%=15438TO15856:READA:NEX
T:FORT%=1TO59:READA:P$(1)=P$(1)+CHR$(A):NEXT:FORT%=1TO59:READA:P$
(2)=P$(2)+CHR$(A):NEXT:FORT%=1TO76:READA:P$(3)=P$(3)+CHR$(A):NEXT
1020 FORT%=1TO10:READA:M$(1)=M$(1)+CHR$(A):NEXT:FORT%=1TO10:READ
A:M$(2)=M$(2)+CHR$(A):NEXT:FORT%=1TO10:READA:M$(3)=M$(3)+CHR$(A):
NEXT:FORT%=1TO10:READA:M$(4)=M$(4)+CHR$(A):NEXT:FORT%=1TO17:READA
:M$(5)=M$(5)+CHR$(A):NEXT
1030 FORT%=1TO17:READA:M$(6)=M$(6)+CHR$(A):NEXT
1040 C1$=CHR$(168)+CHR$(189)+" "+CHR$(26)+STRING$(3,24)+CHR$(154
)+CHR$(143)+CHR$(144):C2$=" "+CHR$(184)+CHR$(191)+CHR$(189)+CHR$(
144)+CHR$(26)+STRING$(5,24)+CHR$(138)+CHR$(175)+CHR$(143)+CHR$(17
5)+CHR$(143)
1050 C4$=" "+CHR$(184)+STRING$(4,191)+CHR$(189)+CHR$(144)+CHR$(2
6)+STRING$(8,24)+CHR$(130)+STRING$(3,131)
1060 CC$=C1$+CHR$(26)+STRING$(3,24)+CHR$(137)+CHR$(157)+CHR$(129
)+CHR$(26)+STRING$(3,24)+CHR$(142)+CHR$(138)+CHR$(132)
1070 P1$=CHR$(143)+CHR$(140)+CHR$(140)+"!!"+CHR$(140)+CHR$(140)+
CHR$(191):P2$=LEFT$(P1$,7)+CHR$(140)
1080 E2$=STRING$(10," ")+CHR$(26)+STRING$(10,24)+STRING$(10," ")
+CHR$(26)+STRING$(10,24)+STRING$(10," ")+CHR$(26)+STRING$(10,24)+
STRING$(10," ")+CHR$(26)+STRING$(10,24)+STRING$(10," ")+CHR$(26)+
STRING$(10,24)+STRING$(10,32)
1090 GR$="":GR$=STRING$(64,RND(63)+129)+STRING$(64,RND(63)+129)
1100 GR$=LEFT$(GR$,10)+"tTtT"+MID$(GR$,11,12)+"EeEe"+MID$(GR$,17
,4)+"1L1L"+MID$(GR$,20,4)+"XxXxXxXxXx"+MID$(GR$,25,5)+"eEeE"+MID$
(GR$,30,6)+"L1L1":GR$=GR$+GR$
1110 RETURNELSERETURN
```

```
1120 DATA191,191,191,32,32,32,32,32,32,32,32,32,32,32,32,32,32,191,
191,189,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,
32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,32,32,191,191,32,32,191,144,158,129,191,32,191,32
1130 DATA32,32,32,32,191,32,191,32,131,191,131,32,191,32,191,32,
191,131,131,32,191,131,191,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,191,191,32,19
1,183,32,32,131,191,131,32,140,140,140,32,191,32
1140 DATA191,32,32,191,32,32,170,176,176,149,32,191,140,140,32,191,1
88,143,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,3
2,32,32,32,32,32,32,32,32,32,32,191,191,191,32,191,130,173,144,32,19
1,32,32,32,32,32,32,191,191,159,32,176
1150 DATA191,176,32,32,191,32,32,191,176,176,32,191,131,191,32,3
2,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,
32,32,32,32,32,32,131,137,140,176,144,32,131,140,176,32,32,137,14
4,32,32,32,149,32,170,32,32,32,160,134,32,32,176
1160 DATA140,131,32,160,176,140,134,131,32,32,32,32,32,32,32,32,
32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32
,32,32,32,130,131,140,164,176,131,172,176,181,176,176,140,129,32,
130,140,176,176,186,176,156,131,176,152,140
1170 DATA131,129,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,
,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,32,3
2,32,32,32,32,131,129,32,32,175,181,136,183,191,187,132,186,159,3
2,32,130,131,32,32,32,32,32,32,32,32,32
1180 DATA32,184,191,191,191,191,189,144,26,24
1190 DATA24,24,24,24,24,24,24,130,131,167
1200 DATA183,167,183,135,131,26,24,24,24,24
1210 DATA24,24,24,24,32,32,32,137,157,129
1220 DATA32,32,26,24,24,24,24,24,24,24
1230 DATA24,32,32,32,138,138,32,32,32
1240 DATA32,160,188,188,188,188,180,32,26,24
1250 DATA24,24,24,24,24,24,24,136,143,159
1260 DATA159,159,159,159,141,26,24,24,24,24
1270 DATA24,24,24,24,32,32,130,167,182,135
1280 DATA32,32,26,24,24,24,24,24,24,24
1290 DATA24,32,32,32,168,169,32,32,32
1300 DATA32,32,176,176,176,176,144,32,26,24
1310 DATA24,24,24,24,24,24,24,160,190,191
1320 DATA191,191,191,191,180,26,24,24,24,24
1330 DATA24,24,24,24,32,32,137,157,153,157
1340 DATA129,32,26,24,24,24,24,24,24,24
1350 DATA24,32,32,32,162,167,32,32,32,26
1360 DATA24,24,24,24,24,24,24,24,32,32
1370 DATA32,130,130,32,32,32,32
1380 DATA130,174,134,26,24,24,24,32,133,133
1390 DATA136,174,140,26,24,24,24,136,129,137
1400 DATA136,184,152,26,24,24,24,32,150,148
1410 DATA160,184,176,26,24,24,24,160,134,164
1420 DATA160,160,160,26,24,24,24,32,155,145
1430 DATA26,24,24,24,32,129,129
1440 DATA32,160,32,26,24,24,24,130,155,147
1450 DATA26,24,24,24,130,32,130
```

***** MAGIC CUBE SOLVER *****

```
10 CLS:PRINT@335,CHR$(23)"MAGIC CUBE":PRINT@596,"SOLVER":T=8:GOS
UB920:CLS:PRINTCHR$(28)
20 A$="EDGE":B$="CORNER":C$="CENTRE"
30 D$="FRONT FACE ":E$="RIGHT FACE ":F$="LEFT FACE ":G$="TOP FAC
E ":H$="BOTTOM FACE ":I$="REAR FACE "
40 J$="A QUARTER TURN CLOCKWISE":K$="A QUARTER TURN ANTICLOCKWIS
E":L$="A HALF TURN":M$="MOVE TO NEXT STAGE":N$="REPEAT LAST STAGE
"
50 A=1:B=2:C=3:D=4:E=5:F=6:G=7:H=8:I=9:J=10:K=11:L=12:M=13
60 O$="STAGE 1 -- PREPARING THE BLUE FACE":P$="STAGE 2 -- THE MI
DDLE LAYER":Q$="STAGE 3 -- THE LAST LAYER":R$="FIRST LAYER EDGE C
UBES":S$="FIRST LAYER CORNER CUBES":T$="MIDDLE LAYER":U$=" - CORN
ER CUBES":V$=" - EDGE CUBES"
70 W$=" - EDGE FLIPS AND CORNER TWIRLS":Y$="PRESS ANY KEY TO CON
TINUE."
80 PRINT:PRINT"THIS PROGRAMME IS A STEP BY STEP METHOD FOR SOLVI
NG THE PUZZLE":PRINT:PRINT"OF THE MAGIC CUBE. IT IS A 'REPEATABLE
 STEP' METHOD THAT WILL":PRINT
90 PRINT"ALLOW THE USER TO REPEAT THE STEPS IN EACH STAGE UNTIL
THEY":PRINT:PRINT"HAVE THAT PARTICULAR STAGE SOLVED. THEY CAN THE
N MOVE ON TO":PRINT
100 PRINT"THE NEXT STAGE UNTIL, FINALLY, THE CUBE IS SOLVED.":PR
INT:PRINT"FIRST WE WILL NEED TO KNOW THE NAMES OF THE SMALL CUBES
.":PRINT:PRINTY$:GOSUB900
110 GOSUB850
120 CLS:PRINT@64,O$
130 PRINT:PRINT"LOCATE THE BLUE ";C$;" SQUARE.":PRINT:PRINT"TURN
 THE CUBE SO THAT THIS SQUARE IS ON THE ";G$:PRINT:PRINTY$:GOSUB9
00:X$="RED"
140 CLS:PRINTO$:PRINTR$:V=V+1
150 PRINTA;" IF EDGE CUBE WITH BLUE AND ";X$;" COLOUR IS ON ":PR
INTH$;" THEN MOVE TO 3 ELSE CONTINUE":PRINTB;" BRING THIS CUBE TO
 BOTTOM BY TURNING APPROPRIATE SIDE FACE"
160 PRINTC;" IF BLUE SIDE OF THIS CUBE IS ON THE BOTTOM THEN ROT
ATE":PRINT"BOTTOM FACE UNTIL ";X$;" COLOUR IS ON ";X$;" FACE THEN
 ":PRINT"ROTATE ";X$;" FACE UNTIL BLUE IS ON THE BLUE FACE":PRINT
D;" IF ";X$;" SIDE OF THIS CUBE IS ON BOTTOM THEN ROTATE"
170 PRINTH$;" THROUGH A QUARTER TURN TO BRING BLUE":PRINT"COLOUR
 OF THIS SMALL CUBE TO A FACE ADJACENT TO ";X$;" FACE":PRINT"THEN
 SIMPLY ROTATE THIS FACE UNTIL ";X$;" COLOUR OF THE SMALL"
180 PRINT"CUBE IS ON ";X$;" FACE THEN TURN ";X$;" FACE UNTIL THE
 BLUE":PRINT"COLOUR COMES TO THE BLUE FACE":PRINTY$:GOSUB900
190 ONVGOTO200,210,220,230
200 X$="ORANGE":GOTO140
210 X$="YELLOW":GOTO140
220 X$="GREEN":GOTO140
230 CLS:PRINT@64,"WE WILL NEED TO EXPLAIN TWO TERMS BEFORE PROCE
EDING WITH THE":PRINT"REMAINING STEPS.":PRINT
```

```
240 PRINT"CLOCKWISE TURN - THIS IS A TURN IN THE DIRECTION OF TH
E CLOCK":PRINT"OF A PARTICULAR FACE WHEN IT IS THE FACE NEAREST T
O YOU.":PRINT"BE SURE TO REMEMBER THAT THIS DIRECTION WILL 'APPEA
R' DIFFERENT"
250 PRINT"ON THE OTHER FACES AND IN FACT WILL APPEAR TO BE ANTIC
LOCKWISE":PRINT"ON THE REAR FACE.":PRINT:PRINT"ANTICLOCKWISE TURN
 - THIS WILL BE THE OPPOSITE TO THE ABOVE.":PRINT:PRINT:PRINT"WE
WILL REVIEW THE FACE NAMES":PRINTY$:GOSUB900:GOSUB850
260 CLS:PRINTO$:PRINTS$
270 PRINTA;" IF THERE ARE NO CORNER CUBES IN THE ";H$:PRINT"PRES
S THE 1 KEY ELSE PRESS 2":GOSUB900:ONZGOTO350,280
280 CLS:PRINTA;" LOOK AT THE THREE COLOURS ON THIS SMALL CUBE AN
D DETERMINE":PRINT"THE POSITION YOU REQUIRE IT TO BE MOVED TO. KE
EPING THE":PRINT"BLUE FACE TO THE TOP ROTATE THE CUBE SO THAT THE
 NEW"
290 PRINT"POSITION IS AT THE FRONT UPPER RIGHT HAND CORNER":PRIN
T"***YOU MUST REMEMBER THE LOCATION OF EACH FACE COLOUR ***":PRIN
TB;"TURN THE BOTTOM FACE UNTIL THE REQUIRED SMALL CUBE IS AT THE"
300 PRINT"FRONT BOTTOM RIGHT HAND CORNER (DIRECTLY UNDER THE REQ
UIRED":PRINT"LOCATION OF THIS SMALL CUBE.":PRINTY$:GOSUB900:CLS
310 PRINTO$:PRINTR$:PRINT"PRESS 1 IF THE BLUE SQUARE IS ON THE "
;D$:PRINT"PRESS 2 IF THE BLUE SQUARE IS ON THE ";E$:PRINT"PRESS 3
 IF THE BLUE SQUARE IS ON THE ";H$:GOSUB900:CLS:PRINTO$:PRINTR$:O
NZGOTO320,330,340
320 PRINT"BLUE SQUARE ON THE ";D$:PRINTA;D$;J$:PRINTB;H$;J$:PRIN
TC;D$;K$:PRINTY$:GOSUB900:GOTO260
330 PRINT"BLUE SQUARE ON THE ";E$:PRINTA;E$;K$:PRINTB;H$;K$:PRIN
TC;E$;J$:PRINTY$:GOSUB900:GOTO260
340 PRINT"BLUE SQUARE ON THE ";H$:PRINTA;E$;K$:PRINTB;H$;J$:PRIN
TC;E$;J$:PRINTD;H$;L$:GOTO330
350 CLS:PRINTO$:PRINTR$:PRINT" EITHER YOU HAVE FINISHED AND ALL
THE CUBES IN THE TOP LAYER":PRINT"ARE IN THEIR CORRECT POSITION (
IN WHICH CASE YOU PRESS 1)":PRINT"OTHERWISE YOU HAVE A SMALL CUBE
 THAT HAS TO BE REMOVED FROM"
360 PRINT"THE TOP LAYER AND REPLACED IN ITS CORRECT POSITION.":P
RINTA;" TURN THE WHOLE CUBE SO THAT THE SMALL CUBE TO BE MOVED IS
":PRINT"AT THE FRONT UPPER RIGHT HAND CORNER"
370 PRINTB;E$;K$:PRINTC;H$;J$:PRINTC;E$;J$:PRINT"PRESS 1 TO MOVE
 TO NEXT STAGE OR 2 TO CONTINUE THIS STAGE":GOSUB900:CLS:ONZGOTO3
80,280
380 CLS:PRINTP$:PRINT:PRINT"TURN THE CUBE UPSIDE DOWN SO THAT TH
E BLUE FACE IS ON THE":PRINTH$;": FROM THIS POINT ONWARDS WE WILL
 NOT REFER TO":PRINT"SPECIFIC COLOURS BUT TO SPECIFIC FACES AND L
OCATIONS."
390 PRINT"THUS WE WILL REVIEW THE NAMES OF THE INDIVIDUAL SQUARE
S THAT":PRINT"GO TO MAKE UP A SINGLE FACE. THEN WE CAN CONTINUE."
:PRINT:PRINT:PRINTY$:GOSUB900:GOSUB850:CLS
400 CLS:PRINTP$:PRINT"FIRST LOCATE AN EDGE CUBE IN THE ";G$;"REQ
UIRED IN THE":PRINTT$:PRINT"IF THERE ARE NONE PRESS 1 OTHERWISE P
RESS 2":GOSUB900:CLS:ONZGOTO450,410
410 PRINTP$:PRINTA;" ROTATE THE TOP FACE UNTIL THE COLOURS ON TH
E SMALL EDGE":PRINT" CUBE ARE THE SAME AS THE FRONT FACE AND THE
LEFT FACE. LOOK ":PRINT"AT THE TOP SQUARE OF THIS EDGE CUBE YOU W
ANT TO MOVE."
420 PRINTB;" PRESS 1 IF IT MATCHES THE CENTRE SQUARE OF THE ";F$
:PRINTC;" PRESS 2 IF IT MATCHES THE CENTRE SQUARE OF THE ";D$:GOS
UB900:CLS:ONZGOTO430,440
430 PRINTP$:PRINT"MATCHING CENTRE SQUARE OF ";F$:PRINT:PRINTA;D$
;J$:PRINTB;G$;L$:PRINTC;E$;J$:PRINTD;G$;J$:PRINTE;E$;K$:PRINTF;G$
;L$:PRINTG;D$;K$:PRINT:PRINTY$:GOSUB900:GOTO400
440 PRINTP$:PRINT"MATCHING CENTRE SQUARE OF ";D$:PRINT:PRINTA;D$
;J$:PRINTB;G$;L$:PRINTC;E$;J$:PRINTD;G$;K$:PRINTE;E$;K$:PRINTF;G$
;L$:PRINTG;D$;K$:PRINT:PRINTY$:GOSUB900:GOTO400
450 CLS:PRINTP$:PRINT:PRINT"IF ALL ";T$;" CUBES ARE IN THEIR COR
RECT POSITION PRESS 1":PRINT:PRINT"IF NOT THEN PRESS 2":GOSUB900:
ONZGOTO490,460
460 CLS:PRINTP$:PRINT:PRINT"THIS SEQUENCE WILL REMOVE AN UNWANTE
D EDGE CUBE FROM THE":PRINTT$;" AND PLACE IT IN A USABLE POSITION
.":PRINT:PRINT"TURN THE WHOLE CUBE (KEEPING THE BLUE FACE ON THE
BOTTOM)"
470 PRINT"AROUND UNTIL THE SMALL CUBE YOU WISH TO SHIFT IS AT TH
E LEFT":PRINT"HAND FRONT EDGE - THAT IS FACING YOU.":PRINT:PRINTY
$:GOSUB900
480 CLS:PRINTP$:PRINT:PRINTA;D$;J$:PRINTB;G$;L$:PRINTC;E$;J$:PRI
NTD;G$;J$:PRINTE;E$;K$:PRINTF;G$;L$:PRINTG;D$;K$:PRINT:PRINTY$:GO
SUB900:GOTO400
490 CLS:PRINTQ$:PRINT:PRINT"THERE IS NOW ONLY ONE LAYER TO BE CO
MPLETED. YOU WILL NEED TO":PRINT"CHECK WHICH OF THE SMALL CUBES Y
OU NEED TO MOVE BEFORE YOU":PRINT"PROCEED. SOME CUBES WILL BE IN
THE RIGHT PLACE BUT THE WRONG"
500 PRINT"WAY AROUND WHILST OTHERS WILL BE IN THE WRONG POSITION
. YOU":PRINT"WILL NEED TO CHECK WHICH SEQUENCE OF MOVEMENTS YOU R
EQUIRE.":PRINT"TO DO THIS WE WILL HAVE A LOOK AT A SERIES OF POSS
IBILITIES"
510 PRINT"IN RELATION TO THE SINGLE FACE OF THE CUBE. WHEN YOU F
EEL THAT":PRINT"YOU HAVE THE CORRECT SHIFTING POSSIBILITY JUST EN
TER THE":PRINT"NUMBER AT THE FRONT OF THE SEQUENCE AND THE MOVEME
NTS WILL"
520 PRINT"THEN BE GIVEN. AFTER YOU HAVE EXECUTED THOSE MOVES YOU
 WILL":PRINT"RETURN TO THE DIAGRAM OF POSSIBLE MOVES. IF YOU HAVE
 ALL THE":PRINT"CORNER CUBES IN PLACE JUST PRESS 5.":PRINTY$:GOSU
B900
530 CLS:PRINTQ$:PRINTU$:PRINT:PRINT"ROTATE THE TOP FACE UNTIL AT
 LEAST ONE CORNER CUBE IS IN THE":PRINT"CORRECT CORNER. DON'T WOR
RY ABOUT POSITION, JUST THE CORNER":PRINT"NOW STUDY THE DIAGRAM T
HAT FOLLOWS AND DETERMINE THE REQUIRED"
540 PRINT"MOVEMENT. THE SEQUENCE WILL BE CORNER TO CORNER. CHOOS
E YOUR":PRINT"SITUATION AND ENTER THE APPROPRIATE NUMBER":PRINT"D
ON'T WORRY IF THE CUBE SEEMS CHAOTIC DURING A SEQUENCE"
550 PRINT"THIS WILL SORT ITSELF OUT BY THE END OF THE SEQUENCE."
:PRINT:PRINTY$:GOSUB900
560 GOSUB810:PRINT@576,"1.  1 TO 3 : 3 TO 7 : 7 TO 1":PRINT@640,
"2.  1 TO 7 : 7 TO 3 : 3 TO 1":PRINT@704,"3.  1 TO 3 : 3 TO 1":PR
INT@768,"4.  3 TO 7 : 7 TO 3":PRINT@832,"5.  TO PROCEED TO ";V$
570 GOSUB900:ONZGOTO580,590,600,610,620
580 CLS:PRINTQ$;U$:PRINT:PRINTA;F$;K$:PRINTB;G$;J$:PRINTC;E$;J$:
PRINTD;G$;K$:PRINTE;F$;J$:PRINTF;G$;J$:PRINTG;E$;K$:PRINTH;G$;K$:
PRINT:PRINTY$:GOSUB900:GOTO560
```

```
590 CLS:PRINT@$;U$:PRINT:PRINTA;G$;J$:PRINTB;E$;J$:PRINTC;G$;K$:
PRINTD;F$;K$:PRINTE;G$;J$:PRINTF;E$;K$:PRINTG;G$;K$:PRINTH;F$;J$:
PRINT:PRINTY$:GOSUB900:GOTO560
600 CLS:PRINT@$;U$:PRINT:PRINTA;D$;J$:PRINTB;G$;K$:PRINTC;I$;K$:
PRINTD;G$;J$:PRINTE;D$;K$:PRINTF;G$;K$:PRINTG;I$;J$:PRINTH;G$;L$:
PRINT:PRINTY$:GOSUB900:GOTO560
610 CLS:PRINT@$;U$:PRINT:PRINTA;G$;J$:PRINTB;D$;J$:PRINTC;G$;J$:
PRINTD;E$;J$:PRINTE;G$;K$:PRINTF;E$;K$:PRINTG;D$;K$:PRINT:PRINTY$
:GOSUB900:GOTO560
620 CLS:GOSUB810:PRINT@576,"1.  2 TO 8 : 8 TO 4: 4 TO 2":PRINT@6
40,"2.  2 TO 4 : 4 TO 8 : 8 TO 2":PRINT@704,"3.  2 TO 8 AND 8 TO
2 : 4 TO 6 AND 6 TO 4":PRINT@768,"4.  2 TO 6 AND 6 TO 2 : 4 TO 8
AND 8 TO 4"
630 PRINT@832,"5.  TO POSITION COLOURS ONCE ALL THIS LAYER IS LO
CATED.":GOSUB900:ONZGOTO640,650,660,670,680
640 CLS:PRINT@$;V$:PRINT:PRINTA;F$;L$:PRINTB;G$;J$:PRINTC;D$;K$:
PRINTD;I$;J$:PRINTE;F$;L$:PRINTF;D$;J$:PRINTG;I$;K$:PRINTH;G$;J$:
PRINTI;F$;L$:PRINT:PRINTY$:GOSUB900:GOTO620
650 CLS:PRINT@$;V$:PRINT:PRINTA;F$;L$:PRINTB;G$;K$:PRINTC;D$;K$:
PRINTD;I$;J$:PRINTE;F$;L$:PRINTF;D$;J$:PRINTG;I$;K$:PRINTH;G$;K$:
PRINTI;F$;L$:PRINT:PRINTY$:GOSUB900:GOTO620
660 CLS:PRINT@$;V$:PRINT:PRINTA;E$;L$:PRINTB;F$;L$:PRINTC;H$;J$:
PRINTD;E$;L$:PRINTE;F$;L$:PRINTF;G$;L$:PRINTG;E$;L$:PRINTH;F$;L$:
PRINTI;H$;J$:PRINTJ;E$;L$:PRINTK;F$;L$:PRINT:PRINTY$:GOSUB900:GOT
0620
670 CLS:PRINT@$;V$:PRINT:PRINTA;E$;J$:PRINTB;I$;J$:PRINTC;G$;J$:
PRINTD;I$;K$:PRINTE;G$;K$:PRINTF;E$;L$:PRINTG;D$;K$:PRINTH;G$;K$:
PRINTI;D$;J$:PRINTJ;G$;J$:PRINTK;E$;J$:PRINT:PRINTY$:GOSUB900:GOT
0620
680 CLS:PRINT@$;W$:PRINT:PRINT"NOW ALL THAT IS NEEDED IS FOR YOU
 TO POSITION THE COLOURS":PRINT" ON EACH OF THE CUBES IN THIS FIN
AL LAYER.":PRINT:PRINT"LOOK AT THE CUBE VERY CAREFULLY.":PRINT:PR
INT:PRINTY$:GOSUB900
690 CLS:PRINT@$;W$:PRINT:PRINT"A TWIRL IS DETERMINED AS FOLLOWS:
":PRINTA;" HOLD THE WHOLE CUBE SO THAT THE SMALL CUBE TO BE TWIRL
ED":PRINT"IS AT THE UPPER RIGHT POSITION."
700 PRINTB;" IF THE COLOUR OF THE SQUARE ON THE FRONT FACE IS TH
E SAME":PRINT"AS THE COLOUR OF THE CENTRE SQUARE OF THE TOP FACE
THEN THIS":PRINT"SMALL CUBE WILL NEED TO BE TWIRLED * CLOCKWISE *
"
710 PRINTC;" IF THE COLOUR OF THE SQUARE ON THE RIGHT HAND FACE
IS THE":PRINT"SAME AS THE COLOUR ON THE TOP FACE THEN THIS SMALL
CUBE WILL":PRINT"WILL NEED TO BE TWIRLED * ANTICLOCKWISE *":PRINT
:PRINTY$:GOSUB900
720 CLS:PRINT@$;W$:PRINT:PRINT"CORNER TWIRLS":PRINT:PRINT"PRESS
1 FOR TWIRLS (REMEMBER THAT IF THERE ARE THREE CORNERS":PRINT"TO
BE TWIRLED THIS SEQUENCE WILL NEED TO BE REPEATED UNTIL":PRINT"AL
L CUBES ARE TWIRLED."
730 PRINT"PRESS 2 TO REVIEW THE DEFINITIONS.":PRINT"PRESS 3 TO F
LIP THE EDGES.":GOSUB900:ONZGOTO740,690,760
740 CLS:PRINT@$;W$:PRINT:PRINT"CLOCKWISE CORNER TWIRL":PRINT:PRI
NTA;E$;K$:PRINTB;H$;J$:PRINTC;E$;J$:PRINTD;D$;J$:PRINTE;H$;J$:PRI
NTF;D$;K$:PRINTG;" ROTATE TOP FACE UNTIL THE SMALL CUBE TO BE ROT
ATED"
750 PRINT"ANTICLOCKWISE IS AT THE TOP RIGHT HAND CORNER.":PRINT:
PRINTY$:GOSUB900:CLS:PRINT@$;W$:PRINT:PRINTH;D$;J$:PRINTI;H$;K$:P
RINTJ;D$;K$:PRINTK;E$;K$:PRINTL;H$;K$:PRINTM;E$;J$:PRINT:PRINTY$:
GOSUB900:GOTO720
760 CLS:PRINT@$;W$:PRINT:PRINT"EDGE FLIPS - TO FLIP ONE PAIR":PR
INT:PRINTA;" START WITH ONE OF THE PAIR IN THE TOP FRONT POSITION
":PRINTB;D$;J$:PRINTC;G$;J$:PRINTD;H$;K$:PRINTE;F$;L$:PRINTF;G$;L
$:PRINTG;H$;L$:PRINTH;E$;J$:PRINT:PRINTY$
770 GOSUB900:CLS:PRINT"EDGE FLIPS - PART 2":PRINT:PRINTA;" DO NO
T CHANGE THE ORIENTATION OF THE CUBE.":PRINTB;" ROTATE TOP FACE U
NTIL OTHER CUBE OF PAIR BEING FLIPPED IS":PRINT"IN THE TOP RIGHT
POSITION."
780 PRINTC;E$;K$:PRINTD;H$;L$:PRINTE;G$;L$:PRINTF;F$;L$:PRINTG;H
$;J$:PRINTH;G$,K$:PRINTI;D$;K$:PRINT:PRINT"PRESS 1 TO REPEAT OR 2
 IF COMPLETED.":GOSUB900:ONZGOTO760,790
790 CLS:PRINT@386,CHR$(23)"CONGRATULATIONS."
800 GOTO800
810 CLS:PRINT@0,"THE CUBE POSITION NUMBERS FOR THE MOVEMENT SEQU
ENCES BELOW"
820 PRINT@153,A:PRINT@158,B:PRINT@163,C:PRINT@281,D:PRINT@286,E:
PRINT@291,F:PRINT@409,G:PRINT@414,H:PRINT@419,I
830 FORX=45TO75:SET(X,4):SET(X,10):SET(X,16):SET(X,22):NEXTX:FOR
Y=4TO22:SET(45,Y):SET(55,Y):SET(65,Y):SET(75,Y):NEXTY
840 RETURN
850 CLS
860 PRINT@10,"A SINGLE FACE OF THE CUBE SHOWING NAMES."
870 PRINT@202,B$:PRINT@214,A$:PRINT@226,B$:PRINT@458,A$:PRINT@47
0,C$:PRINT@482,A$:PRINT@562,"HIT ANY KEY":PRINT@626,"TO RETURN.":
PRINT@714,B$:PRINT@726,A$:PRINT@738,B$
880 FORX=16TO88:SET(X,4):SET(X,16):SET(X,28):SET(X,40):NEXTX:FOR
Y=4TO40:SET(16,Y):SET(40,Y):SET(64,Y):SET(88,Y):NEXTY
890 GOSUB900:RETURN
900 Z$=INKEY$:IFZ$=""THEN900
910 Z=VAL(Z$):RETURN
920 FOR N=1TO(T*250):NEXTN:RETURN
```

```
10 REM  #### NEWS JUMBLED PLAYERS  FINDER ####
 #### COPYRIGHT GEOFFREY EGEL 18 STURT ST LOXTON AUG 1981 #####
20 CLEAR 5000:CLS
30 DATA GLENELG,J.PAYNTER,R.WALTER,M.LUNNIS,S.COPPING,S.KERNAHAN
,R.SEWER,D.MARSHALL,D.RADY,C.MCDERMOTT,D.HOLST,P.WESTON,P.JOHNSTO
N,J.LITHOU,K.KUHLMANN,M.FARQUHAR,C.MACVICAR,G.CORNES,P.MCINERNEY,
C.VEIDE,L.TWELVFTREE,J.SEEBOHM
40 DATA P.CAREY,P.ROGERS,N.CALDWELL,A.SYMONDS,P.BROWN,D.ARMFIELD
,M.GREENSLADE,J.SEEBOHM,M.HARRISON
50 DATA 0
60 DATA TORRENS,M.HEWETT,R.GALT,T.HANK,K.HILL,D.INGLIS,T.HILL,A.
GREEN,K.MCSPORRAN,R.MANOUGE,P.PHILLIPOU,S.GRAMBEAU,S.WHITE,R.ENRI
GHT,L.ROBSON,P.ROBERTS,T.RENDELL,I.HANAJA,M.FOOTE,N.ROBERTS,J.CUN
NINGHAM,G.TRIPP,I.HEWETT
```

```
70 DATA I.HEWETT,G.FLANAGAN,B.MILLER,  B.HANK,M.GEORGE,S.BUCKLEY,
I.HEWETT,P.JOHANSON,B.MILLER,R.DIORIO,D.SYMONDS,P.DROGEMULLER,I.H
ANNA
80 DATA 0
90 DATA WESTADELIADE,S.WELLS,R.LUDERS,G.MORRIS,T.THORN,D.KENNEDY
,J.FALTING,G.FIELKE,L.WATSON,R.DONALDSON,D.DREW,R.HAYES,A.BURGESS
,R.WINTER,M.DAVOREN,J.KANTILAFTAS,M.GREGG,I.BORCHARD,S.RECHNER,S.
BECHNER,R.BENNETT,P.MEURET
100 DATA G.MORRIS,T.BURGESS,  B.LINDNER,R.HAYES,P.KENNEDY,R.HAMIL
TON,M.BEECH,T.BURGESS,D.DEJONG
110 DATA 0
120 DATA SOUTHADELIADE,W.SLATTERY,G.LINKE,P.COLLINS,G.AITKEN,J.S
CHNEEBICHLER,J.DAWKINS,R.WHITE,P.CRATE,P.HARRADINE,G.MOUSLEY,G.BR
OOKSBY,J.ALLEN,S.BULTER,P.BROOKSBY,S.PALMER,R.HAWKINS,G.BAYNES,M.
NALEY,S.KUSS,M.SCHULZ
130 DATA G.BROOKSBY,R.HATELY,R.DAWKINS,M.KEAM,R.DAWKINS,S.EMERY,
T.ALLEN,B.HONOR,S.PALMER,G.FARQUHAR,D.FAIRCLOUGH,  P.WOULD,A.MCINT
YRE
140 DATA 0
150 DATA STURT,T.SIMS,P.HOLLIS,P.HEINRICH,M.GRAHAM,G.WISEMAN,E.F
RY,C.FIELKE,P.FRIEDRICHS,R.WILSON,C.CASEY,R.MAGAREY,K.KLOMP,G.LEO
NARD,R.HILL,N.WARK,F.SPIEL,N.CRAIG,J.DERRINGTON,P.HARGREAVES,G.RE
ED,K.SMITH
160 DATA  P.HOLLIS,B.HOWARD,G.WHITTLESEA,P.SIMOUNDS,M.HIENRICH,G
.ZURBINICH,K.WHYATT,G.BRAND,K.SMITH,J.HUNT,D.SMITH
170 DATA 0
180 DATA PORTADELIADE,S.WILLIAMS,T.EVANS,G.BOYD,D.CAHILL,R.JOHNS
TON,R.AGIUS,K.KINNEAR,S.CLIFFORD,P.BELTON,P.HOFNER,G.PHILIPS,A.GI
LES,I.BRADMORE,T.SORRELL,I.ECKERMAN,C.NATT,R.EBERT,B.CUNNINGHAM,M
.HAMIL,A.PORPLYCIA
190 DATA C.EBERT,G.NATT,D.GRANGER,A.MARTYN,L.WARREN,B.ABERNETHY,
D.HUGHES,R.AGIUS,B.COLE,M.ERREY
200 DATA 0
210 DATA CENTRALDISTRICT,C.MCLAY,M.GEORGE,I.MCKAY,S.TRIGG,P.BEYT
HIEN,P.GRAHAM,P.VIVAN,S.GILLIES,S.LIVESEY,L.STONE,C.EARNSHAW,D.CO
WIE,G.MUTZE,M.OBRIEN,T.MOORE,R.VANDOMMELE,M.NORSWORTHY,K.LIENERT,
J.LOKAN
220 DATA G.EDWARDS,D.HART,G.GILLIES,P.MOBBS,M.PRIOR,D.CLOHESSY,P
.BUBNER,S.BOEHM,W.COCHRANE,J.PLATTEN,W.WILSON,B.GRAHAM,M.WRIGHT
230 DATA 0
240 DATA WOODVILLE,R.MANSFIELD,G.HEWITT,R.BEECROFF,P.WELSH,W.CAR
TY,N.MCCORMACK,D.WATSON,B.CLOSE,A.MATHESON,C.SCHEER,M.PARKER,L.HE
AVEN,S.REARDON,W.DAVIS,C.LUKOSIUS,T.MCLEOD,A.LENNON,R.HUPPATZ,B.T
YRRELL,I.DETTMANN
250 DATA K.MANSFIELD,T.PIERSON,W.CARTY,C.DOWSETT,J.SIMPSON,R.KER
STEN,B.BEECROFT,P.BOLTON,L.RICHARDS,G.SHEER,D.BOLTON,B.TONKIN,K.C
RANE,W.DAVIS,B.TONKIN,M.LOCKWOOD
260 DATA 0
270 DATA NORTHADELIADE,P.MAKSIMOVIC,G.SCHMIDT,G.MCADAM,C.STANBRI
DGE,P.ARNOLD,D.ALGATE,B.PHILIPS,J.RILEY,R.GEHLING,W.STRINGER,T.CL
ISBY,S.FORTH,D.TILLER,S.KERLEY,R.ROBRAN,M.REDDEN,G.SCHLEIN,N.NUNA
N,P.CLOKE,D.CONDER
280 DATA T.ROACH,B.STRINGER,K.GREEN,C.PELLE,W.LOKAN,A.STRINGER,M
.ELLIS,D.WILDY,T.GRENFELL,S.MOLLOY
290 DATA 0
300 DATA NORWOOD,G.DALWOOD,N.BUTTON,G.MENZEL,D.FOSDIKE,B.HARTFIE
LD,R.WOODCOCK,D.JENKINS,P.GALLAGHER,K.THOMAS,A.AIGH,J.THIEL,F.STE
MPER,G.NICHOLSON,G.DUNSTAN,T.BELL,U.COLASANTE,P.ADLER,G.TURBIL,D.
SEROTZKI,S.HOFFMANN
310 DATA N.BALME,P.IULIANO,A.AISH,M.NOBLE,J.SAMBELL,M.AISH,D.FOS
DIKE,G.HUGHES,B.WINTER,K.THOMAS,G.MENZEL,T.HOILE,N.PEARSON,J.GALL
AGHER
320 DATA 0,END
330 PRINT "JUMBLED PLAYERS":PRINT STRING$(63,140):PRINT "INSTRUC
TIONS Y/N"
340 A$=INKEY$:IF A$="" THEN 340ELSE IF A$="Y" THEN 750
350 CLS:PRINT"JUMBLED NAMES DECODER":PRINT STRING$(63,140):L$="/
":C$=".-":W$="":PRINT@768,STRING$(63,140):PRINT"          A  PROGR
AM  BY GEOFF  EGEL  ":PRINT STRING$(63,140)
360 PRINT@128,"ENTER THE NEWS JUMBLED PLAYERS ":INPUT" LETTERS";
W$:IF W$="" THEN 360ELSE W$=W$+C$:J$=W$+L$
370 PRINT@64,
380 T=T+1: ON T GOTO 390,400,410,420,430,440,450,460,470,480,720
390 A$="GLENELG": GOTO 490
400 A$="TORRENS":GOTO 490
410 A$="WESTADELIADE":GOTO 490
420 A$="SOUTHADELIADE":GOTO 490
430 A$="STURT":GOTO 490
440 A$="PORTADELIADE":GOTO 490
450 A$="CENTRALDISTRICT":GOTO 490
460 A$="WOODVILLE":GOTO 490
470 A$="NORTHADELIADE":GOTO 490
480 A$="NORWOOD"
490 T$=J$
500 X=LEN(A$):U=LEN(T$)
510 FOR F=1 TO X:Z=0
520 P$=MID$(A$,F,1):IF P$="/" THEN 570
530 Z=Z+1:IF Z>U THEN F=X : Z=0:GOTO  380
540 F$=MID$(T$,Z,1)
550 IF F$=P$ THEN H$=MID$(T$,1,Z-1):Z$=MID$(T$,Z+1):T$=H$+Z$:GOT
O 570
560 GOTO 530
570 NEXT F: PRINT A$: AA=1
580 E=E+1:READ N$:IF N$<>A$ THEN 580
590 READ N$
600 IF N$="O" THEN 380
610 J$=W$:I$="-"
620 N$=N$+I$+A$+L$:R=LEN(N$):Q=LEN(J$)
630 FOR K=1 TO R:S=0
640 P$=MID$(N$,K,1):IF P$="/"  THEN 690
650 S=S+1 : IF S>Q THEN K=R:S=0:GOTO 710
660 F$=MID$(J$,S,1)
670 IF F$=P$ THEN H$=MID$(J$,1,S-1):Z$=MID$(J$,S+1):J$=H$+Z$:Q=Q
+1: GOTO 690
680 GOTO 650
690 NEXT K
700 FF=1:PRINT N$,J$
710 J$=W$:K=R: GOTO 590
```

```
720 IF AA=0 THEN PRINT"THIS CLUB DOES NOT EXIST IN SOUTH AUST ":
GOTO 730ELSE IF FF=0 THEN PRINT"THIS SURNAME NOT IN DATA FILE"
730 PRINT "SEARCH COMPLETED"
740 END
750 CLS:PRINT"JUMBLED PLAYERS DECODER":PRINT STRING$(63,140):PRI
NT "THIS PROGRAM IS SUITABLE FOR DECODING JUMBLED NAMES PROVIDING
   THE PLAYERS FIRST INITIAL ALONG WITH A FULLSTOP AND SURNAME IS
  ADDED INTO THE DATA LINES
760 PRINT"THE TEAM NAME SHOULD BE ADDED FIRST BEFORE PLAYERS NAM
ES IF YOU ARE  MAKING AN INTERSTATE VERSION OF THIS PROGRAM AND W
HEN A    COMPLETED TEAM IS ENTERED  (0)  SHOULD ALSO BE ADDED TO
THE DATALINE"
770 PRINT "THE VERSION HERE IS SUITABLE FOR THE NEWS TANGLED PLA
YERS AND   ALL THAT IS NEEDED IS TO ENTER THE LETTERS WHEN REQUES
TED YOU   SHOULD ALSO CHECK THE VITAL STATISTICS FOR NAMES THAT
 ARE NOT IN THE DATA FILE AND ADD THEM"
780 PRINT" PRESS ANY KEY TO CONTINUE"
790 A$=INKEY$:IF A$="" THEN 790ELSE 350
```

```
10 '   *************************
         MATRIX MANIPULATION
       (C) 1980 BY MARTIN DOWNEY
       *************************

20 Y=0:I=1:CLS:PRINTTAB(12);"MATRIX MANIPULATION (BY M.DOWNEY)":
PRINT
30 ME=MEM/6:DIMA(ME)
40 PRINT:PRINTY;"MATRICES ALREADY DEFINED":PRINT"MATRIX INSTRUCT
IONS:"
50 PRINT"PRINT=1 : DEFINE=2 : MULT=3 : ADD/SUB=4 : INVERSE=5 : D
ELETE=6"
60 INPUTZ:CLS:ONZGOSUB80,140,190,310,390,690
70 GOTO40
80 IFY=0RETURN
90 W=0:FORX=1TOY:PRINT"MATRIX";X
100 W=W+2:M=A(W-1):N=A(W)
110 FORR=1TOM:FORC=1TON:W=W+1:PRINTTAB((C-1)*6)A(W);
120 NEXTC:PRINT:NEXTR:INPUT"PRESS (ENTER) TO CONTINUE":A$:IFZ<>1
RETURN
130 NEXTX:RETURN
140 CLS:PRINT"MATRIX";Y+1
150 INPUT"NUMBER OF ROWS & COLUMNS: R,C";M,N:A(I)=M:A(I+1)=N
160 Y=Y+1:I=I+2:PRINT"ENTER ONE ELEMENT AFTER EACH  '?'"
170 FORR=1TOM:PRINT"ROW";R;:FORC=1TON:PRINT@(130+R*64+C*6),"";:I
NPUTA(I)
180 I=I+1:NEXTC:NEXTR:RETURN
190 INPUT"FIRST MATRIX'S NUMBER";A:IF(A<1)+(A>Y)THENRETURN
200 INPUT"MULTIPLIED BY: CONSTANT=1  OR  MATRIX=2";V:IFV=1THEN28
0
210 INPUT"SECOND MATRIX'S NUMBER";B:IF(B<1)+(B>Y)RETURN
220 GOSUB660:A(I-1)=P:IFN<>OTHENPRINT"NOT POSSIBLE":I=I-2:RETURN
230 IF(I+M*P+10)>METHEN720
240 FORJ=0TO(M-1):FORH=0TO(P-1):A(I)=0:FORK=0TO(N-1)
250 A(I)=A(I)+A(L+J*N+K)*A(Q+K*P+H):NEXTK
260 I=I+1:NEXTH:NEXTJ:Y=Y+1
270 PRINT"MATRIX";A;"TIMES MATRIX";B;"GIVES MATRIX";Y:GOTO100
280 INPUT"CONSTANT OF MULTIPLICATION";K:GOSUB660
290 FORJ=1TO(M*N):A(I)=A(L+J-1)*K:I=I+1:NEXTJ
300 Y=Y+1:PRINT"MATRIX";A;"TIMES";K;"GIVES MATRIX";Y:GOTO100
310 B$="PLUS (":INPUT"FIRST MATRIX'S NUMBER";A:IF(A<1)+(A>Y)RETU
RN
320 INPUT"MULTIPLE OF ADDITION. (-1 GIVES MINUS , +1 GIVES PLUS)
";V
330 IFV<0B$="MINUS ("
340 INPUT"SECOND MATRIX'S NUMBER";B:IF(B<1)+(B>Y)RETURN
350 GOSUB660:IF(M<>O)OR(N<>P)THENPRINT"NOT POSSIBLE":I=I-2:RETUR
N
360 IF(I+M*N+10)>METHEN720
370 FORJ=0TO(M*N-1):A(I)=A(L+J)+V*A(Q+J):I=I+1:NEXTJ:Y=Y+1
380 PRINT"MATRIX";A;B$;ABS(V);"TIMES MATRIX";B;") GIVES MATRIX";
Y:GOTO100
390 INPUT"MATRIX NUMBER";A:IF(A<1)+(A>Y)THENRETURN
400 GOSUB660:L=L-1:I=I-1:IFN<>MTHENPRINT"NOT SQUARE":I=I-1:RETUR
N
410 FORJ=1TON:FORK=1TON:IFK=JTHENA((J-1)*N+K+I)=1:GOTO430
420 A((J-1)*N+K+I)=0
430 NEXTK:NEXTJ
440 FORQ=1TO(N-1):FORJ=QTO(N-1):U=1:GOTO620
450 IFA((Q-1)*N+Q+L)=0THEN570
460 R=A(J*N+Q+L)/A((Q-1)*N+Q+L):FORK=1TON
470 A(J*N+K+L)=A(J*N+K+L)-R*A((Q-1)*N+K+L)
480 A(J*N+K+I)=A(J*N+K+I)-R*A((Q-1)*N+K+I):NEXTK:NEXTJ:NEXTQ
490 FORQ=1TO(N-1):FORJ=(Q+1)TON:U=2:GOTO620
500 R=A((Q-1)*N+J+L)/A((J-1)*N+J+L):FORK=1TON
510 A((Q-1)*N+K+L)=A((Q-1)*N+K+L)-R*A((J-1)*N+K+L)
520 A((Q-1)*N+K+I)=A((Q-1)*N+K+I)-R*A((J-1)*N+K+I)
530 NEXTK:NEXTJ:NEXTQ
540 FORJ=0TO(N-1):FORK=1TON:A(J*N+K+I)=A(J*N+K+I)/A(J*N+J+1+L):N
EXTK:NEXTJ
550 I=I+N*N+1:Y=Y+1:GOSUB700:W=I-N*N-3
560 PRINT"MATRIX";A;"HAS BEEN DELETED. IT'S INVERSE IS MATRIX";Y
:GOTO100
570 FORO=JTO(N-1):IFA(O*N+Q+L)=0THEN600
580 FORP=QTON:A((Q-1)*N+P+L)=A((Q-1)*N+P+L)+A(O*N+P+L)
590 A((Q-1)*N+P+I)=A((Q-1)*N+P+I)+A(O*N+P+I):NEXTP:GOTO460
600 NEXTO
610 PRINT"MATRIX";A;"DELETED. SINGULAR.":GOSUB700:I=I-1:RETURN
620 FORP=1TON:T=0:S=0:FORO=1TON:IFA((P-1)*N+O+L)=0THENT=T+1
630 IFA((O-1)*N+P+L)=0THENS=S+1
640 NEXTO:IF(S=N)OR(T=N)THEN610
650 NEXTP:ONUGOTO450,500
660 W=1:FORJ=1TOY:IFJ=ATHENM=A(W):N=A(W+1):L=W+2
670 IFJ=BTHENO=A(W):P=A(W+1):Q=W+2
680 W=W+2+A(W)*A(W+1):NEXTJ:A(I)=M:A(I+1)=N:W=I-1:I=I+2:RETURN
690 INPUT"MATRIX NUMBER";A:IF(A<1)+(A>Y)RETURN
700 GOSUB660:Y=Y-1:I=I-M*N-4
710 FORJ=(L-2)TO(I-1):A(J)=A(J+M*N+2):NEXTJ:RETURN
720 PRINT"INSUFFICIENT MEMORY. TRY DELETE.":RETURN
```

```
                00100 ;         RESET/ED   DISK (32 OR 48 K)
                00110
9000            00120           ORG     9000H
                00130
                00140 ;         DENNIS BAREIS (C)
                00150 ;         286 LENNOX ST
                00160 ;         MARYBOROUGH, 4650
                00170
                00180 ;         THIS PROGRAM WILL RESET THE COMPUTER
                00190 ;         WHEN YOU HOLD THE SPC-BAR AND THE CLEAR
                00200 ;         KEY DOWN TOGETHER.
                00210 ;         THE PROGRAM RELOCATES ITSELF TO HIGH
                00220 ;         MEMORY AND WILL PROTECT ITSELF, IT WILL
                00230 ;         ALSO PATCH IN ANY EXISTING KEYBOARD DRIVER.
                00240 ;         YOU SHOULD WAIT FOR THE DISK DRIVES TO
                00250 ;         TURN OFF BEFORE RESETING THE COMPUTER.
                00260
                00270
402D            00280 DOS       EQU     402DH           ;RE-ENTRY POINT FOR DOS.
0000            00290 DUMMY     EQU     0000H           ;ANY VALUE, GETS PATCHED.
4049            00300 HIMEM     EQU     4049H           ;TOP OF MEMORY POINTER.
0000            00310 RESET     EQU     0000H           ;JUMP TO ZERO TO RESET.
                00320
9000 F3         00330 INIT      DI
9001 3A1540     00340           LD      A,(4015H)
9004 FE00       00350           CP      0               ;TEST IF DRIVER AT 4016H,
9006 2805       00360           JR      Z,DVR34         ;OR AT 4034H.
9008 2A1640     00370 DVR16     LD      HL,(4016H)      ;DRIVER AT 4016H.
900B 1803       00380           JR      SAVEDR
900D 2A3440     00390 DVR34     LD      HL,(4034H)      ;DRIVER AT 4034H.
9010 225490     00400 SAVEDR    LD      (DRIVER+1),HL   ;PATCH IN EXISTING DRIVER
9013 2A4940     00410           LD      HL,(HIMEM)      ;GET MEM. SIZE.
9016 B7         00420           OR      A               ;RESET CARRY FLAG.
9017 111E00     00430           LD      DE,FINISH-START+1  ;SIZE OF PROGRAM.
901A ED52       00440           SBC     HL,DE
901C 224940     00450           LD      (HIMEM),HL      ;NEW TOP OF MEMORY SIZE.
901F 23         00460           INC     HL              ;POINT TO START OF PROG.
9020 3E01       00470           LD      A,01H           ;MAKE SURE (4015H)=1.
9022 321540     00480           LD      (4015H),A
9025 221640     00490           LD      (4016H),HL      ;PATCH IN MY PROGRAM.
9028 EB         00500           EX      DE,HL           ;DE=DESTINATION FOR LDIR.
9029 2A1340     00510           LD      HL,(4013H)      ;FIND LOCATION OF CLOCK.
902C 226190     00520           LD      (CLOCK+1),HL    ;PATCH IT INTO MY PROGRAM
902F 214590     00530           LD      HL,START        ;START PROG(BEFORE RELOC)
9032 011E00     00540           LD      BC,FINISH-START+1  ;LENGTH OF PROGRAM.
9035 EDB0       00550           LDIR                    ;RELOCATE PROGRAM.
9037 011200     00560           LD      BC,INTER-START  ;FIND RELATIVE POS, INTER
903A 2A1640     00570           LD      HL,(4016H)      ;POSITION OF START.
903D 09         00580           ADD     HL,BC
903E 221340     00590           LD      (4013H),HL      ;PATCH IN INTER.
9041 FB         00600           EI
9042 C32D40     00610           JP      DOS             ;RETURN TO DOS.
                00620
9045 3A4038     00630 START     LD      A,(3840H)       ;PEEK KEY-BOARD.
9048 FE82       00640           CP      82H             ;SPC-BAR & CLEAR DOWN ?
904A 2007       00650           JR      NZ,DRIVER
904C AF         00660 RES       XOR     A               ;YES, ZERO THE A REGISTER
904D 32EC37     00670           LD      (37ECH),A       ;OUTPUT 0 AS COMMAND.
9050 C30000     00680           JP      RESET
9053 CD0000     00690 DRIVER    CALL    DUMMY           ;GETS PATCHED HERE WITH
9056 C9         00700           RET                     ;THE PREVIOUS DRIVER.
                00710
9057 F5         00720 INTER     PUSH    AF
9058 3A4038     00730           LD      A,(3840H)       ;PEEK KEY-BOARD.
905B FE82       00740           CP      82H             ;SPC-BAR & CLEAR DOWN ?
905D 28ED       00750           JR      Z,RES           ;IF SO, RESET COMPUTER.
905F F1         00760           POP     AF
9060 C30000     00770 CLOCK     JP      DUMMY           ;JMP TO THE CLOCK ROUTINE
                00780
9062            00790 FINISH    EQU     CLOCK+2         ;MARKS THE END OF PROGRAM
                00800
9000            00810           END     INIT
00000 TOTAL ERRORS
```

***** NEXT MONTH'S ISSUE *****

Next month's issue will contain at least the following programs plus the usual features and articles. An (80) after a program title indicates that the program will be for TRS-80 Model 1/3 or System 80/Video Genie computers. (Colour) indicates that the program will be for the TRS-80 Colour Computer and the Hitachi Peach.

** OTHELLO LII/4-16K (80) **

This popular game can now be played on your computer. The game of OTHELLO is played on an 8 x 8 grid like a chess board. The object of the game is to occupy more squares than your opponent, in this case the computer.

** PASSWORD LII/16K (80) **

Password is a machine language program that allows the user to put the computer into a dead loop until he enters a four letter code specified by the user. When a valid code has been entered, the program returns you to BASIC. Just the thing to keep those sticky fingers in your home from doing any harm in your absence.

** CONVERSIONS (COLOUR) **

This program allows you to convert Metric units to Imperial and vice versa for the quantities temperature, length, distance and weight. Handy for those brought up on the old system.

** MICRO GRAND PRIX 16K/ml (80) **

Any similarity between the way that Micro Grand Prix works and the way a well known "RACING" game works (sold by you know who!) is purely INTENTIONAL! Any similarity in the code that achieves it is purely COINCIDENTAL!

** LOAN CALCULATION PACKAGE LII/16K (80) **

With housing loans going up the way they are, this program should interest just about everybody. It provides five separate functions, Repayment Calculation, Remaining term calculation, Remaining balance calculation, Dissection of repayments and Repayment factor calculations.

** NORMAL DISTRIBUTION (COLOUR) **

Many statistical programs assume that data is distributed normally. This program allows you to enter up to 100 data points, to edit them if necessary and then displays these statistical measures:- mean, variance, standard deviation, standard error and range.

APPLICATION FOR PUBLICATION OF A PROGRAM IN MICRO-80

Date .....

To **MICRO-80** SOFTWARE DEPT. PO BOX 145 MORPHETTVALE SA 5162

Please consider the enclosed program for . . .

Tick where appropriate

(i) Publication in MICRO-80 .....

(ii) Publication on disk or cassette only .....

(iii) Both .....

Name .....

Address .....

.....

.....Postcode .....

*** * * CHECK LIST * * ***

Please ensure that the cassette or disk is clearly marked with your name and address, program name(s), Memory size, Level I, II, System 1 or 2, Edtasm, System, etc. The use of REM statements with your name and address is suggested, in case the program becomes separated from the accompanying literature.

Ensure that you supply adequate instructions, notes on what the program does and how it does it, etc.

For system tapes, the start, end, and entry points, etc.

The changes or improvements that you think may improve it.

Please package securely — padabags are suggested — and enclose stamps or postage if you want your cassette or disk returned.

***** CASSETTE/DISK EDITION INDEX *****

The Cassette and Disk editions are presently available for '80 computers only.
The cassette edition of MICRO-80 contains all the software listed each month, on cassette. The cassette also contains the source code for machine language programs which may not have been printed due to space restrictions. All programs are recorded twice. Level I programs can only be loaded into a Level I TRS 80. To load these programs into a Level II TRS 80 or a System 80/Video Genie, you must first load the Level I in Level II program from the MICRO-80 Software Library. System 80/Video Genie computers have had different tape-counters fitted at different times. The approximate start positions shown are correct for the very early System 80 without the volume control or level meter. They are probably incorrect for later machines. The rates for a cassette subscription are printed on the inside front cover of each issue of the magazine.

The disk edition contains all those programs which can be executed from disk, including LevelI programs. Level I disk programs are saved in the NEWDOS format. Users require the Level I/CMD utility supplied with NEWDOS + or NEWDOS 80 version 1.0 to run them. Distribution disks are recorded in single density, Model I format. Model 3 users will need to Convert them.

SIDE 1	TYPE	I.D.	DISK FILESPEC	CTR-41	CTR-80	SYSTEM-80
JUMBLED PLAYERS	L2/16K	P	PLAYERS/BAS	18	10	10
"	"	"	"	82	46	48
CUBE SOLVER	L2/16K	S	SOLVER/BAS	141	79	83
"	"	"	"	236	133	140
MATRIX MANIPULATIONS	L2/16K	M	MATRIX/BAS	323	182	190
"	"	"	"	348	196	206
RESET	SYSTEM	RESET	RESET/CMD	375	211	220
"	"	"	"	379	214	225
SIDE 2						
RESET	EDTASM	RESET	RESET/EDT	18	10	10
"	"	"	"	48	27	28
SKY DIVER	L2/16K	D	SKYDIVE/BAS	78	44	46
"	"	"	"	188	106	111
SPACE GALAXY	L1/4K	-	Not on disk	284	160	168
"	"	-	"	337	190	200

Please note that there was an error in Space Galaxy on the April tape. The progam has therefore been included on this month's tape. The error did not involve the disk edition so the program is not included on this month's disk.

TO:
MICRO-80, P.O. BOX 213, GOODWOOD, SOUTH AUSTRALIA. 5034.

Please RUSH to me the items shown below:

$ enclosed Date

.....12 month subscription to MICRO-80

.....12 month subscription to MICRO-80, plus the cassette edition

.....The latest issue of MICRO-80 (see inside front cover for prices)

The MICRO-80 PRODUCTS listed below:

DESCRIPTION	PRICE

TOTAL ENCLOSED WITH ORDER

☐ Cheque ☐ Bankcard ☐ Money Order

Bankcard Account Number

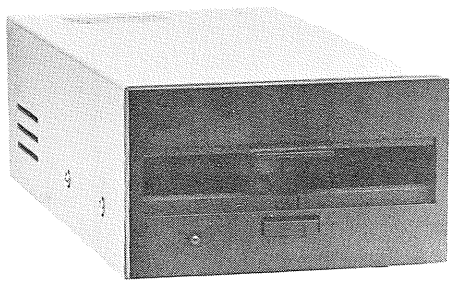
Signature ..... Exp. End .....

NAME .....

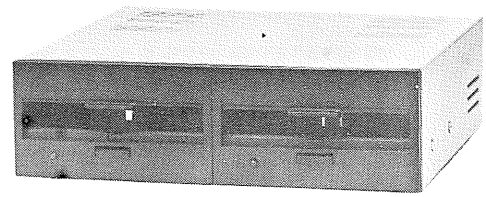
ADDRESS .....

Postcode .....

SAVE A PACKET ON MICRO-80's DISK DRIVE PACKAGES FOR TRS-80 MODEL 1 AND SYSTEM 80 MICROCOMPUTERS



SINGLE DRIVE PACKAGE *from* ... $499



DUAL DRIVE PACKAGE *from* ... $874

Bigger volume means lower cost price, which we are passing on to you. Avoid the annoying bundle of cables, wires and separate boxes. MICRO-80 is now offering our well-proven MPI disk drives in attractive, self-contained single or dual-drive cabinets complete with internal power supply. Our drive Ø and dual-drive packages also include the appropriate version of DOSPLUS and dual-drive cable.

The best news of all is the specially reduced package prices ...
SAVE $23 — $107 over our already low prices!

Choose the appropriate system from the table below:

DRIVE TYPE	No. of Tracks	No. of Heads	Capacity	Dosplus Version	Price	* Saving
DRIVE Ø						
1 x MPI B51	40	1	100K	3.3	$499	$77.95
1 x MPI B52	40	2	200K	3.4	$639	$97.95
1 x MPI B92	80	2	400K	3.4	$799	$107.95
DRIVE 1						
1 x MPI B51	40	1	100K	—	$415	$23.00
1 x MPI B52	40	2	200K	—	$525	$23.00
1 x MPI B92	80	2	400K	—	$695	$23.00

*Represents the saving compared with buying all the items included in the package separately

•Drive Ø package includes one bare disk drive, self-contained single-drive cabinet/power supply as illustrated, two drive cable and the version of DOSPLUS indicated.

•Drive 1 package includes one bare disk drive and self-contained single-drive cabinet/power supply as illustrated.

If it's a dual-drive system you need, then take advantage of our dual-drive package and
SAVE a further $40 on the price of two single-drive packages ...

DRIVE TYPE	No. of Tracks	No. of Heads	Capacity	Dosplus Version	Price
2 x MPI B51	40 ea	1 ea	2 x 100K	3.3	$874
2 x MPI B52	40 ea	2 ea	2 x 200K	3.4	$1125
2 x MPI B92	80 ea	2 ea	2 x 400K	3.4	$1454

Dual-drive package includes two bare disk drives, self-contained dual-drive cabinet/power supply as illustrated, two drive cables and the version of Dosplus indicated.

NOTE: All 40 track drives are completely compatible with 35 track operating systems such as TRSDOS. DOSPLUS allows you to realise an additional 14% capacity compared with TRSDOS. Under DOSPLUS 3.4, 80 track drives can read 35/40 track diskettes.

All disk drive components are still available separately:

BARE DRIVES — MPI drives offer the fastest track-to-track access time (5 milliseconds) available. All drives are capable of operating in double density for 80% greater storage capacity.

	Price		Freight
MPI B51 40 track, single-head, 100K	$399	New, Reduced Price	$5.00
MPI B52 40 track, dual-head, 200K	$449		$5.00
MPI B92 80 track, dual-head, 400K	$619		$5.00
Simple, wrap-around cabinet	$12		$2.00
Separate, dual-drive power supply	$85		$8.00

	Price	Freight
Self-contained, single drive cabinet/power supply	$99	$5.00
Self-contained, dual-drive cabinet/power supply	$135	$5.00
Two drive cable	$39	$2.00
Fan drive cable	$49	$2.00
DOSPLUS 3.3	$99.95	$2.00
DOSPLUS 3.4	$149.95	$2.00

Prices are FOB Adelaide. Add $5.00 freight for single drive package, $10.00 for dual-drive package. Prices are in Australian dollars. Freight is road freight anywhere in Australia.

All items carry a 90-day parts and labour warranty. Repairs to be carried out in our Adelaide workshops.

MICRO-80

LEVEL 2 ROM
ASSEMBLY LANGUAGE TOOLKIT
by Edwin Paay
FOR TRS-80 MODEL 1, MODEL 3 AND SYSTEM 80/VIDEO GENIE

This is a new package consisting of two invaluable components:

- **A ROM REFERENCE** Manual which catalogues, describes and cross-references the useful and usable ROM routines which you can incorporate into your own machine language or BASIC programs.
- **DBUG**, a machine language disassembling debugging program to speed up the development of your own machine language programs. DBUG is distributed on a cassette and may used from disk or cassette.

Part 1 of the ROM REFERENCE manual gives detailed explanations of the processes used for arithmetical calculations, logical operations, data movements etc. It also describes the various formats used for BASIC, System and Editor/Assembly tapes. There is a special section devoted to those additional routines in the TRS-80 Model 3 ROM. This is the first time this information has been made available, anywhere. Differences between the System 80/Video Genie are also described. Part 1 is organised into subject specific tables so that you can quickly locate all the routines to carry out a given function and then choose the one which meets your requirements.

Part 2 gives detailed information about each of the routines in the order in which they appear in the ROM. It describes their functions, explains how to use them in your own machine language programs and notes the effect of each on the various Z80 registers.

Part 2 also details the contents of system RAM and shows you how to intercept BASIC routines. With this knowledge, you can add your own commands to BASIC, for instance, or position BASIC programs in high memory — the only restriction is your own imagination!

The Appendices contain sample programmes which show you how you can use the ROM routines to speed up your machine language programs and reduce the amount of code you need to write.

DBUG: Eddy Paay was not satisfied with any of the commercially available debugging programs, so he developed his own. DBUG: allows you to single-step through your program; has a disassembler which disassembles the next instruction before executing it or allows you to bypass execution and pass on through the program, disassembling as you go; displays/edits memory in Hex or ASCII; allows Register editing; has the ability to read and write System tapes and all this on the bottom 3 lines of your screen, thus freeing the rest of the screen for program displays. Four versions of DBUG are included in the package to cope with different memory sizes.

The best news of all is the price. The complete Level 2 ROM ASSEMBLY LANGUAGE TOOLKIT is only:

— Aus. $29.95 + $2.00 p&p
— UK £18.00 + £1.00 p&p

SPECIAL OFFER TO OWNERS OF THE LEVEL II ROM REFERENCE MANUAL ...
UPGRADE TO THIS ASSEMBLY LANGUAGE TOOKIT FOR ONLY $19.95!
Send back your original Level II ROM Reference Manual plus a cheque, money order or Bankcard authorisation for $19.95 plus $2.00 p&p and we will send you the new ASSEMBLY LANGUAGE TOOLKIT

MICRO-80