

DRAGON USER



July 1988

The independent Dragon magazine

Contents

Letters

Help wanted with software experience ... what was in the back issues? ... we will pay for better games ... more projects please ... stand up and write ... Gemini needed ... feature updates.

News desk

Take your Dragon and shout about it.

Expert's Arcade Arena

The second half of the Christie Egg Screen Designer before TE goes on his hole.

Crossword

Crossword eight with tapes to win

Dragonsoft

Quadrant's Kingdom and Orderings of Cruth from Orange Software, Nite Jail from Precision, Lullabyer Mitter from John Piers, inder Postfall from Compuserp and Everything you wanted to know about DSG from Jason Shouler

Memory and assembly

Pam D'Arcy maps out the Dragon's memory for machine code users.

Winners and losers

Gordon Lee discusses the solution to the industry competition.

Data grows on trees

Jonathan Cartwright uses trees and branches to store and retrieve data.

Getting into Prestel

Ron Smith tells of his early experiences on BT's main database.

Compressing the screen

Jonathan Cartwright squeezes his screens into smaller spaces.

DOS copier

Rudy Guck copiers files in any direction on Dragon Data and Burnhard DOS files.

Write: ADVENTURE

Pete Gerard homes in on mistakes made by adventure writers in game planning.

Adventure Trail

A solution to Pyramids of Doom, some adventure maps, and a headbanging for software writers.

Competition

Gordon Lee counts the time it takes to make your first million.

The Answer

Gordon Lee's solution to the April competition.

Classified Ads.

Dragon Answers

Brian Cadogan on saving programs to disc, a mysterious problem with Dream, inspiring information into a running program and diverting TRON to the printer.

Editorial

What an irresistible opportunity — I mopped up practically everything worthy the name of news last month by going round the Desert Show and phoning up half the Dragon world beforehand. Results: three quarters of a page to three around this month.

I've done two things with it: one is to reproduce a 'universal Dragon User' which I hope readers with a suitable opportunity will be able to display on a notepad or in a window. The other is to offer tips a line for news reporting. Don't all fall over at once — it must be new news which we haven't had from primary sources, but think about it: find two lines of publishable news and you've paid for the stamp ...

The Expert is tied up with exams for about three months, so if anyone else with a genuine addition problem and a bit of know-how fancies themselves as columnist for a month, here's your chance. Send an outline if you don't want to write columns on spec, to me, at Dragonpicks.

If anyone else thinks we owe them a Don Stoker by the end of next week, please write. I'm still having a spot of bother collating files after the move. What I need is a reverse Hoover — one which disgorges things I thought I had dealt with just as I discover I haven't. Now, about those questionnaires ...

Telephone number
(01) 576-8305

Editor
HELEN ARMSTRONG

Production Editor
HELEN ARMSTRONG-GARTNET

Administration/Advertising
BOB HARRIS

Publisher
DRAGON PUBLICATIONS

Subscriptions
UK £14 for 12 issues
Overseas (surface) £25 for 12 issues
ISSN 0265-177

Address: Dragon Publications, 49 Alexandra Road, Hounslow, Middlesex TW3 4NF, United Kingdom.

Published by Dragon Publications 1988
© Dragon Publications 1988

Typeset by Arbel Limited, London NW1

Printed by Hoadley Brothers Ltd, Ashford, Kent

Registered at the Post Office as a newspaper. Dragon and its logo are trademarks of Burnhard Ltd.

How to submit articles

The quality of the material we can publish in Dragon User each month will, to a very great extent, depend on the quality of the submissions that you can make with your Dragon. The Dragon computer was launched on to the market with a powerful version of Basic, but with very poor documentation.

Articles which are submitted to Dragon User for publication should not be more than 5000 words long. All submissions should be typed. Please leave adequate margins and a double space between each line. Programs should, whenever possible, be computer printed on plain white paper and be accompanied by a tape of the program.

We cannot guarantee to return every submitted article or program, so please keep a copy if you want to have your program returned; you must include a stamped addressed envelope.

Letters

This is your chance to air your views — send your tips, compliments and complaints to Letters Page, Dragon User, at 84a Catford Road, Haurville, Middle T9D 4NP.

Old news in demand

It is possible to give a brief summary of the contents of Dragon User for the past three and a half years so that I could order back copies? I am particularly interested in firmware routines (would be exceeding grateful).

R Palmer
26 Linton Road
Rams
Russett, Wymondham
NR14 4GG

Well, hello, naturally in this office has been close a complete index of Dragon User, for the very good reason that we never have time, but we know it's man who has... Publisher Software have published Magazine for Dragon User, which gives you a complete index of your fingertips, costs under a fiver, and is updated periodically. For like it from me to interleave upon the fruits of their good work. Publisher can be contacted at 28 Foshill, High Compton, Stroud, Glos GL2 7HQ.

We will pay for quality

I would like to agree with Ken Smith in the May edition when he wrote that he would be prepared to pay a higher price for some top class American software. I have been enquiring whether there was any body at present buying software from American, and whether it was practical to do so.

Unfortunately, it appears that there is very little being produced in the USA for the CoCo 2, which is the equivalent of our trusty Dragon. The main interest now is in the CoCo3, which has a far larger memory, and is therefore not compatible. I would love to be joined writing on this matter.

It was a still good software I would be interested in forming a collective to identify sources and import the software. Could anyone inform me as to what

Every month we will be shelving out a game or two, courtesy of our supplier, to the reader's who send the most interesting or entertaining letters. So send us your hints and your opinions, send us your hi-sses and suggestions. Send us your best Dragon stories. What d'you think we are, mind readers?



We want articles on how to use software

Many thanks for keeping DU going. A possible idea for a few lines might have been to use some of the more complex programs, eg Flex, Dynabook spreadsheet. I find that one has to spend so much time trying to get things running one runs out of available time when a few good worked examples would clear up most of what are really just silly faults.

I am not knowing programs, but it is just that instructions only become clear after you don't need them. Maybe that's Harris has a few tips giving his BASIC programs — there must be some things not covered by the other sheets.

J J Gilling, 235, John Road,
Claverdon, Avon, RG21 7TG

RARELY have I seen the 'documentation dilemma' stated so succinctly and with such dignity. I am in total agreement, although I will say in defense of people who write computer manuals — not that most of them do it — that users do occasionally come up with some very imaginative questions; you know the kind of thing — how do I use my spreadsheet to write a novel? Willing can I get my exercise out of the disc drive? And other things which reduce programmers to tears. Serve them right, too.

Bob is buried deep under a heap of paper at the moment, but he may find something to add in due course. Meanwhile, if anybody wants to write a definitive guide to using any of the major packages, please drop me a line and volunteer, stating your software and your experience. This is potentially a very useful angle, and one which DU hasn't touched on for some time.

Is this possible or even legal as a group basis?

Finally, if any of the companies still supporting the Dragon are considering arranging the import of such software, I would like to assure them that I am prepared to pay £100-£150 for good software.

Ian Burford
79 Herford Road
Gloucester
GL2 8SL

WP at rest

WONDER upon wonder! This morning I receive the long awaited next issue of Dragon User, and what do I see but one

massive article about using the printer port as a controller for external circuitry. Brilliant! Seriously if you keep this up you'll start giving people like me ideas and one day a cheap graphics card may arise.

I can see that what the Dragon has been lacking all along is a good resolution with eight colours or so, and a programmable word generator. It shouldn't be too difficult to interface a graphics processor and a sound processor. For example, the Texas Instruments TMS320C25 HL graphics chip (which isn't incorporated in the TK 8044A, the Emulator and the 8020 machines) comes complete with 16K ram, so that a minimum of interfac-

ing and external circuitry is required. The point chips which most of the other computers (eg the Oric, Emulator) have are similar to the AY 3-8910 which provides three voices and works with programmable envelopes etc. I'm sure such a chip could be easily interfaced. When I think of the sort of power affordable it, say, the newest 16 graphics processor for instance chips, I see it reviewed in 8/11 it would provide graphics resolution capability. However, I am still learning about electronics, as it is a hobby and I'm busy enough without it.

Thanks for clearing up the 'obscure dream' in the letters page. Concerning my word processor, my Dragon is not in the bank, not in the waste processor. What I lack is a display! I said my self as I am studying and a TV would completely obliterate my chances of learning anything new to begin research next year. Yes, Pete Gerrard was right about my being male — girls tend to have a 'hyperactive' school, so they're the ones to copy lecture notes from! I hope he said correct about the delay, in three years' time, maybe.

So until I get hold of a computer video monitor I will have to use a pen.

P V Skid
118 Polhemus Court
Catherine Street
Liverpool
L69 7SD

SUCH education is an example to us all. How do you live without 'Wogan'? Your information was wrong about the 'typewriter' snail, by the way, unless he means 'hand-writing' determining due to constant use of keyboard. Anybody who has received an editorial red ink letter will know what I mean.

Stand up and write

AS a result of placing a letter in Dragon User about my new magazine, Dragon Logic, I've had a grand total of five letters. In fact, one was from a Dragon enthusiast who didn't own a

News desk

If you have any new products for the Dragon — software or hardware — send us your press release

AS most of the forward news in the Dragon world was reported in the aftermath of the Great Drive (see last month's issue of Dragon User), there isn't a great deal of product news to report this month. Grange Software are continuing to extend their range of disc software (Grange Software, The Girth, Star Fleet, Navy Dory, Averagentry, Great MIT 90P). They usually have an up-to-date sheet with a guide to available formats on it, and sometimes supply unfilled formats to order ... Bernd Neuner of the Siegfried Computer-Gruppe reports having seen an advertisement for Dragon 44s for \$69 at California Digital, 17702 Figueroa St., Carls, California, CA 90048, USA. Write to California Digital for information and postal charges. The Siegfried Computer-Gruppe publish an AS magazine in German and can be contacted care of Bernd at Dins-Emsbacher-Strasse 23, D-44024 Neunkirchen am Brand 1, Bundesrepublik Deutschland.

Why an ad?

Below this column you will see what looks (or should look) very much like an advertisement for Dragon User.

So why are we advertising in our own pages? And what has this to do with you?

Stick 'em up

The answer is very simple. Major press advertising would swallow up DU's entire budget in a couple of bytes, but we have always regarded this advertisement as a few thousand readers, and that is in a few thousand readers. Many community buildings, clubs and colleges have noticeboards or advertising broadsheets where special interest notices can be displayed, sometimes free, sometimes (as in the ubiquitous newspaper's window) for a few pence a week.

We are not demanding that you cut up your Dragon User! Most libraries have photocopies which will make a good copy for 10p. Copy shops are more expensive, at around 16-18 pence a single copy. Or you may have access to a friend or colleague's copies.

There are other things you can do, as well. Write a letter to your local newspaper or community magazine, or to another computer magazine, general interest magazine, or national newspaper (aiming for the computer club page if there is one), and simply tell them that DU exists, that it is an independent, non-profit-making publication supporting the Dragon and SBCs community, and where people can find us.

Reporters, please

Well, something is true. If you have your hands on any current news which is pertinent to the Dragon, Tandy and SBCs world, send it to the editor marked NEWSDESK, and if we use it we will pay you 10p a printed line (or £1 per printed and given a minute).

The news must be reasonably timely, and must not involve any business or private interest of your own or a colleague's. Please include your source where possible. If you draw our attention to current items in other publications, so that we are able to follow the lead and verify it ourselves, we will pay half the fee above. Payment unpublished news only. First come first served by date of postmark; we cannot use material drawn from press releases which we subsequently receive from the manufacturer unless they arrive after we have published. Replies to readers' letters and non-Dragon-specific product information won't be accepted as news. All copy will be edited as necessary, and the editor's decision is final.

Of course, if you have a product, your group or other Dragon-related activity you want mentioned in your own behalf, we will publish it as usual if you send a press release to the Newsdesk. Copy date is around the 10th of the month every month.

FOR THE DRAGON COMPUTER

DRAGON USER

is the only independent, professionally-produced magazine for Dragon 32 and 64, Tandy Colour Computer and 6809 users in the UK and Europe.

Published monthly, Dragon User carries information features on programming in Basic and machine code, utility and entertainment programs, game and adventure playing, hardware projects, reviews of new software and hardware, answers to technical questions, and a monthly competition.

Dragon User is available by subscription from Dragon Publications, 49 Alexandra Road, Hounslow, Middlesex TW3 4HP. Tel: (01) 570 8335

Expert's Arcade Arena

Write to "The Expert" at Dragon User
48 Alexandra Road, Hove, Sussex,
Wicks TN9 4HP.

HELLO and goodbye! I've got to go now, to leave room for the necessary edited highlights of the Chuckie Egg screen designer instructions. See you in October, cheerio now. Take it away, ladies!

Chuckie Egg designer

The program consists of three programs, **listings one, two and three**. **Listing one** recodes the original screens to a fixed length format, upon which the editor can operate, and a second program, **listing two**, is then required to convert these formatted screens back to the original format, when the game is played. The new set of screens is held high in memory, above Chuckie Egg, and was necessary because the original screens were of random lengths making them almost impossible to edit.

To enter the machine code you must, firstly type in the basic loader program. You must then RUN it and use it to enter listings one, two and three (the instructions for doing this are detailed in the initial issue of the program). The loader, **listings one and two**, and the first part of **listing three** can be found in the June 1985 issue of Dragon. **Listing three** continues with the full page of data 30894-32323, and the short passage 32324-32576.

The code should then be committed to tape as follows:

Listing one:
CSAVEFM "1",30894:32323
Listing two:
CSAVEFM "2",32324:32576
Listing three:
CSAVEFM "3",30894:32576

Once you have typed in and recorded these three programs, you can then compile Chuckie Egg+ by typing the commands below:

1) POKEM13:EXEC 49004
2) "Ready your Chuckie Egg original
3) INCH=C0F0:READ:
POKE13:30894+37 NEXT
DATA13:30894,77,126,94,95,
4) RUN
5) EXEC32706
6) "Ready your cassette containing listings 1 to 3
7) CLEAR99,32345

8) CLEAR0M "1"
9) EXEC 7660
10) CLEAR0M "2"
11) CLEAR0M "3"
12a) CSAVEFM "CH-EGG+"
32324,32324,32576
b) CSAVEFM "CH-EGG+"
32324,32324,32576
13) CLEAR99,32345
EXEC 76400

In the above steps, **listing three** contains the commands for saving and provided snippet 13, the second of which will produce an automatic saving (which can be deactivated by typing CLEAR99 before loading). Both versions should be loaded using CLEAR0M and the ordinary version is executed by typing EXEC.

Instructions

Upon execution the available screen will be displayed and the time will be played. There are then three extra options. If ... Enter the screen editor (not available if the current set of screens has been "locked"). ... Turn the title time on/off (no noticeable immediate response). ... Load a complete set of screens from cassette. Upon entering the screen editor, the first screen will be displayed, and the cursor will flash in the centre of the screen. The cursor can be moved around the screen using the arrow keys. The other functions are as follows: **BREAK** Moves to the next screen CLEAR ... Exit the screen editor ENTER ... Saves screen to cassette facilities and is followed by:

- 1 ... Save single screen only
- 2 ... Save current set of screens, unlocked
- 3 ... Save current set of screens, locked. Looking at screens keeps out unauthorized eyes, as they cannot be viewed or edited.
- 4 ... Load single or set of screens. If a loading error occurs, the editor will return control to the main program.
- Any other key returns C
- Clear all the blocks in the game playing area. T ... Turns the screen lock on/off. When the screen is on, the current character will still be cursor as it is moved by the cursor keys. ... Repeat key D ... Draw all the blocked and the farmer. These will be removed as the cursor passes over them, but

their positions will remain in memory. ... Draw only the chickens which will appear on the screen at the start of each game. ... Increase the number of initial chickens on the screen. ... Decrease the number of initial chickens on the screen. R ... Restore key. The screen as displayed is only in temporary storage and is permanently stored when BREAK. Clear or Enter is pressed. To erase the screen layout in temporary storage and restore the last permanent screen, press the R key. Chickens are not displayed on the screen layout because they are larger than the other elements of the game, and they can be placed on top of other elements. The elements of the game can be placed on the screen by pressing the appropriate keys, but they must be placed according to the rules of screen design. Some of these rules are checked for all the keys are pressed (immediately) when attempts to leave the screen — by pressing BREAK, Clear or Enter. If these conditions are not met, you will not be able to leave the screen.

Mad ducks

To place a wall onto the screen, press the "W" key. This will press the "5" key on a standard game, and the spacebar for a space. These elements can be placed anywhere in the game area (the game area includes the top three screen lines, the area taken up by the mad duck and the platform

upon which the farmer begins). All screens must contain at least one seed grain, but so more than chicken.

Ladders must always be placed as pairs of ladder blocks, since ladders are twice as wide as other elements. To place a ladder block press the "L" key. Ladders laid right cannot be placed at the extreme left edge of the screen, as these positions cannot be accessed by the farmer (because of his far turn — Expert). Ladders cannot be placed on the bottom row of the screen as chickens would descend them and leave the program. Where a ladder intersects (or connects with) a wall, press the "I" key to place an interception block. For the interception to be available, the ladder must protrude by two levels in height above the wall, and the computer will check for this. Each screen must contain at least one ladder.

To place an egg onto the screen layout, press the "E" key. Each screen must contain twelve eggs.

The farmer always begins at the bottom of the screen, just left of centre. This position cannot be changed and the wall underneath the farmer cannot be removed.

To place a lift on the screen, press the "T" key when the cursor is on the left-hand side of the required lift shaft position (the lift shaft is two blocks wide — although the original author always used a width of four blocks in order to make the game more difficult). A lift shaft is indicated by an "L" symbol in



SCREEN SHOT OF CHUCKIE EGG EDITOR

the bottom left-hand corner of the start and this block should be replaced by another element of the game to remove the lift. Placing a lift on the screen removes any other elements in its path, and the lift cannot be placed at the edges of the screen or intersect at a third stack to occupy the former's starting position. Only one lift operated on each screen.

Each screen contains five chickens, but not all of them start on the screen at the beginning of each game (the

number can be changed, see above). Chickens can be placed side and two blocks high. To move a chicken to another location on the screen, press the number 1-8 referring to the chicken you wish to relocate when the cursor is in the top-left-hand corner of the space where the chicken would take up. The positions of the chickens are stored in memory separately from the rest of the screen contents, so the chicks can be placed on top of other elements, without removing them and the

chickens are only eaten temporarily (they are also not affected by the RESET/CRC keys). The program creates if the chicks cannot move, so they must have a wall under either foot, or chickens overlapping with their top half (that is why the chicken on level three appears to be flying).

Forbidding the 2828 series of data means most of a day's task for anyone. Paul Burgin says he can supply a cassette containing listings and to those for £2.50, payable to Paul

Burgin, 26 Moorcroft Road, Sheffield S10 4DS (UK only please). Come on, lift &, and take into Burgin regular quantities of food about another lift and an extra crop of whatever... Have fun!

(On its goodbye from him, and it's goodbye from me, goodbye).

The Expert will be back later in the year if anyone wants to contribute their deeper thoughts on the finer points of exotic games to the magazine, write to the Editor.

```

30894 + 00001034163899FD660C66 = 2278
30895 + 01043888289C9D02F73516 = 1813
30916 + 0E343F155E792F06C94668 = FC5
30927 + 774F687762837974537F73 = 1048
30938 + 347117897E228C77914370 = 1582
30949 + 104478782C788C38796448 = 1683
30960 + 78943170803270548237F48 = 262E
30971 + 3470F63570F057779874979 = 2828
30982 + 702077914C7794C45779883 = 1E9C
30993 + 77944479C0883188E7C08 = 13E6
30804 + 8051028C7C628018278C8E = 1297
30885 + 7C5380518276781F1F2138 = 29C
31026 + 888800404032043089288 = 179F
31037 + 7878133F213886289C9C28 = 1905
31048 + F43887F04838881F21381F = 1888
31059 + 1F189329C41FC11F26E828 = 198F
31070 + 882820001F3138881F1883 = F95
31081 + 29C41F3D3AC37888C828C = 2867
31092 + 887C7194284C848777287E = 8838
31103 + 77E3867CF17E52C86811F = 1838
31114 + 2187648847038471C97718 = 12F7
31125 + 88782884F87E781586C828 = 119F
31136 + 884F288384F0281848C28 = 1895
31147 + 88888188884F44188CF883 = 188F
31158 + 1F7331F08F329C41F1827F8 = 1903
31169 + 57C11F1827F8319432C886 = 183E
31180 + 88240488488818827448C88 = 1874
31191 + 6628F81F08F329C41F1827F8 = 1954
31202 + 8873F818827F8318827F8F1 = 8828
31213 + 8827F8C18425C788C84F78 = 1831
31224 + 0C8787878488078C888888 = 1237
31235 + 78883888378C9C5EFA888 = 1835
31246 + 38888847848878C81F217E = 1887
31257 + 78137878188828C78FA88C = 2086
31268 + 9838888878217E78843478 = 1868
31279 + 1F189329C41F384F5848484 = 1448
31290 + 5F4C3888878C8848288C = 1883
31301 + 08888878C9C8388884888 = 28F8
31312 + 88288784F188288882848 = 0C7F
31323 + F84818488578AC787484918 = 1982
31334 + 1888478847818787C14184 = 018
31345 + 88848484840C40C33C7C8C3 = 198C
31356 + 1818888888888888888888 = 2386
31367 + 8F88848888888888888888 = 1E78
31378 + 1888848888888888888888 = 1C33
31389 + 8837878887883948288788 = 2170
31400 + 8F88878887878888888888 = 1D98
31411 + 8788188C9C9C8C8C8C8888 = 1C88

```

```

31422 + 88C88888248C884871948 = 1798
31433 + 1C4881C828C81E12888887 = 1848
31444 + 28888881381E1288C8788 = 238E
31455 + 28E3888C8C253188888786 = 1868
31466 + 0C34C8888C8C8C8C81F88C = 1881
31477 + 8038C88888478888178488 = 188E
31488 + 8428888C8C8C8C8C88888C = 0E37
31499 + 3718888C8C8C8C8C8C8C8C8 = 1888
31510 + 8828C8888F78888878888 = 244C
31521 + 8C28C88787818888888888 = 2825
31532 + 88888C88888C8888888888 = 2828
31543 + 8888888888888888888888 = 2168
31554 + 8C88888888888888888888 = 2798
31565 + 0C818F0888888888888888 = 1883
31576 + 888C88888888888888888888 = 0838
31587 + F88888888C88888888888888 = 2113
31598 + 88888188888888888888888 = 1988
31609 + 188888888888888888888888 = 2887
31620 + 88888C8C8F7C8C8C8C8C8C8 = 2448
31631 + 348C9C2C318888888888888 = 1F88
31642 + 188888888888888888888888 = 1784
31653 + 88888F888888888888888888 = 88C1
31664 + 88C1888413C88888888888888 = 2438
31675 + 38F388888888888888888888 = 1488
31686 + 881888C88888888888888888 = 2848
31697 + 884188888C18888F487878 = 2873
31708 + 178888C8F84878848888888888 = 177E
31719 + 81888888888888888888888888 = 148E
31730 + 88888C84888888888888888888 = 8F86
31741 + 88787888418887C1888C888C = 19E2
31752 + 38888881882787818818888888 = 8888
31763 + 88888888888888888888888888 = 888F
31774 + 48888888888888888888888888 = 187F
31785 + 8C888888888888888888888888 = 121C
31796 + 88843888888888888888888888 = 1288
31807 + 52328F38888888888888888888 = 1188
31818 + 8852828F328488888888888888 = 1826
31829 + 58488888888888888888888888 = 1828
31840 + 48888455588888888888888888 = 1318
31851 + 488F8888888888888888888888 = 1E48
31862 + 888888F248888888888888888888 = 128E
31873 + C1FC888888888888888888888888 = 2888
31884 + 343888C188878888888888888888 = 2887
31895 + 8888888888888888888888888888 = 8888
31906 + 27774E3188888888888888888888 = 1811
31917 + 8834271388888888888888888888 = 1888
31928 + 2888888888888888888888888888 = 8888
31939 + 877E888888888888888888888888 = 1E78

```

31950	1236689773336252169888F	=1560	32234	7925828925941337898888	=3183
31961	8078002887646780348882	=5239	32247	3390430088338883382588	=1790
31972	2680C1823784C182368888	=86F	32250	5889888F3481121F189329	=F78
31983	3333872818818237888C88	=1FF7	32269	3484C41F58588721358444	=1316
31994	4881811278F81823788C181	=1887	32280	564456C4F8E7847E787888	=2804
32005	2787C18237837878F8C844	=C462	32291	70718E7C188F5182888C2	=C468
32016	0848848181348888888881	=1F93	32302	5893485883484C48934848	=1284
32027	8827F13818218236888888	=1F88	32313	483148F888888888888131	=188F
32038	088182378481821884F888	=1978	32324	277C813318278888814818	=1187
32049	60887C7028888888888189E	=2148	32335	27887C814C1823F888881E	=1884
32060	235F88888F8F1E84841838	=1484	32346	88888788FC1E8881823448	=1948
32071	88488884182388843818938	=8888	32357	882748C18232418F787881	=1884
32082	281E129E2F88898887881E	=182F	32368	882788C182378488773884	=8238
32093	58384E121F18732838884C	=121E	32379	088377488788818881882	=1149
32104	4FE72135848848884F884F	=1328	32390	88887888C778832F78874	=88C4
32115	87848C1E8F843818818727	=F84	32401	1827F8388888C787818882	=888F
32126	F88188288C1F1883884718	=F78	32412	88888F7838F8881E8888887	=2848
32137	8E38E78888C788C888887C	=2288	32423	887818881288178888881E	=888C
32148	808828FF88187E2588888C8	=2832	32434	88188E2588888788818F4	=28F3
32159	88888483848888888888834	=188C	32445	28F7878388888888888881	=1788
32170	84888848888888888888848	=1783	32456	F888888888888888888888	=1888
32181	88884848C188C888348888	=888C	32467	8F1888C28888C888888877	=888F
32192	8887888C8C8888881882388	=18E2	32478	8F18888888888888888813	=8833
32203	33888888888888888888833	=1284	32489	38887888C78F73478887C	=2825
32214	18932F8484C41F87888884	=171C	32500	8F48228E3388C88887878C3	=C88C
32225	584F884F84F887888C888C	=1F8C			

Crossword

Please get your answers in to Dragon User Crossword Department by the end of the month on the front cover.

The eighth Dragon Crossword emerges from the bottom of a tea chest, a little crumpled but none the worse — oh, my mistake, that's the editor! The sixth Dragon Crossword is spoken in secret verse by G. Wigham of Dunblane, who would like "Pantomime Pantomime One, as there's no point in coming to anything less than the best, is there?" and Ian Adair of Cherley Wood, who would like "anything I haven't got yet, as long as it's good". The phrase is "MAGIC LANGUAGE".

There will be a couple of free tapes from the Editor's Magic Bottomless Box for the first correct entries out of the hat each month. You can't be telling as which tapes you'd like — you never know, we may have them.

And you don't have to cut up your Dragon User — entries on a photostat or a plain piece of paper will do.

1. Side by side — I step out a fine oddity enough (10)
2. No mice on me, with nothing on — about an alien satellite (7,4)
3. Alone again — play with long sax and a bass (8)
- 4 and 5. Puzzle about the celebrity from Indonesia? (12,3,4,4)
6. A moored oddity by Mexican Peter (5)
7. Driven as hard as intruders (8)
8. Bottomless-war chambers leading to a terrible tale? (12,4)
9. What the Union soldiers called the Confederates (6,5)
10. There's a tiger caged at German POW camp (6)
11. Keep a villain into a trap with food (7,4)
12. Get not rank confused, and boldly went where no man, etc (5,4)
13. Reminded on TV? — His letters lie all around him (10)



by Terry and Derek Probyn

All this month's answers are names of Dragon software. When the crossword is complete, the column marked with an arrow will spell out a phrase.



By no means just your average Zap

Program: Lucifer's Kingdom
Supplier: Orange Software
Price: £5.99 cassette

THE picture on the cassette box is just like the title of this game — very misleading, as it shows the horned head of a demon. At first glance, this appears to be another space-type shoot-em-up game (and is even described as such on the cassette itself) but this is very far from the whole truth. Although my two sons (and even myself when I first played it) loved it as a pure zap-em-up game, this should be played more like a graphic adventure game. By that I mean that you have to plan out your actions and not just necessarily shoot all the baddies.

—There are other objects which can be shot, and greater scores can be achieved by destroying some of these things. The most important of these are the crystals which are placed under characters on the screen and gradually appear as the characters are shot away. They can then be collected by flying over them and a certain number are needed before you can progress from one region of six planets to the next.

One thing you cannot do, however, is to forget about the baddies in your quest for

crystals, as these are a variety of types of these which come up in a variety of ways and also fire at you differently. My pet hate are the ones that descend straight down the screen until they reach your level, and then come horizontally across at you, forcing you to retreat or move further up the screen — if you can! Others circle you in a fan fashion, firing at you as they move around. Also if you are not quick enough to destroy one wave of baddies, then the next bunch seems down upon you and they may or may not be the same type. The colour of assailants and the number of them appears to be random, and also seems to have been generated in a way different from the usual poor Dragon randomiser.

If your ship is destroyed, then you start from that point and are not returned to the beginning, which I personally prefer. However, though, if you have not got past the last of the baddies which you were fighting when you were destroyed, as you will then be attacked by the same type again at the restart. Incidentally, you get five lives and I have not yet discovered a way of getting more, although you appear to have unlimited health and ammunition.

Only the very middle of the screen is used in this game, and so much of the screen is used for the score, etc. The movement is very smooth indeed, as is the scrolling and the speed-up poke has not been used to get the increase in momentum, just good programming. I was amazed at the response to the joystick in this game. Why isn't every game like this? Some of the baddies come at you down the very edge of the screen and these cannot be shot, no matter how far over you steer your ship. At the end of each planet, which is not very long in the early stages, the game is paused while you receive your bonus points and normally I do not like pauses in the middle of games, but believe me you need one.

I would not surprise me if this game caused some controversy in the manner of *The Force* about bugs, as at times it appears that you have shot straight at the enemy without harming it, but this is caused by parts of the terrain which can be shot over and over again, amassing points, but which reverse as you hit them: the first time you can fire through them at the enemy, but not when they have turned round. Also there is a slight problem in the fire button on

the left joystick, which is the one that you use in this game, will fail to work when the right joystick is connected to its own port. At times there is so much happening on the screen that you may not notice some of the enemy bullets which come in all directions.

Also some bullets appear to be directed out of nowhere, and not really from the baddies. You'll see what I mean when you play it. Another thing which throws me is that it says Player 1 (Indesapable) handles one of the screen with your score alongside it, and as yet I have been unable to access a two-player game as an afterthought. All in all this is a game which I really enjoy playing, and will also be enjoyed by those of you who like pure zap games. Many people will say that it is too similar to *Utopia*, but I have found that the two games need completely different tactics, and enjoy playing both. I would have liked to have given this program five Dragons, but hesitate to give the maximum to a game I have not yet completed. If the rest of the game reaches the same high standard it is certainly worth five.

Mike Short



Find an enemy in your inventory

Program: Underbelly of Ooth
Supplier: Dragonfire Services
Price: £4.00

THIS is an old adventure originally released by Marston Software around 1985, and now revived by Dragonfire. Loading, as usual from cassette, is very slow and the loading screen is not very inspired.

However, when the adventure is loaded you are treated to what must be one of the best screen presentations of any adventure. The text is set out on an old piece of parchment and scrolls nicely up the screen as inputs are made. The drawback to this is that if you forget your surroundings "LOOK" has to be typed to re-

mind you of the scenery. The instructions state that input can be in sentence form, and it took me quite a while to work this out. As you cannot "do one thing AND do another", instead, the word THEN has to be used between the two actions.

This proves to be very useful as I played it more and more, though you have to be careful with your inputs as you have to add for the computer to go through the two separate actions before you can make another move. Another touch I like very much was the automatic putting of a capital to the first letter of your input and email ones to the others. This generally gave it a very professional appearance and in very

pleasing to the eye. The instructions claim that the words are understood, and I must say that I got less "I don't understand" than usual.

As for the storyline, you play the part of an explorer trapped initially in a cage over a pit somewhere underground, and have to find treasures and then escape. There are some nice touches during the adventure, and nothing too outrageous for you to have to do in order to escape, although you are presented with some tedious situations. As one of the warnings: take careful note of what is at each location. There is an ape-woman at one location, and I returned to this location having completed another task, and no mention was

made of her. However, I did what is expected and to and behold she was there, like a cat-in-the-hat to a certain location, as I have lost my life this way several times. The story has a very outstanding setting and some very subtle problems which can, however, be solved logically. One major drawback is that the trash bag is not deleted, and I have completely lost the game several times and I have to admit that I'm one of those people who rarely saves their positions in an adventure game. (That reminds me — G2.)

Mike Short



Just one step to the real thing

Title: *Larkspur Walkoff Is Trapped*

Supplier: John Pene Discovers Software

Price: £10.00

Info: The first in a trilogy of adventures with the others presumably hot on its heels (part two out now, review shortly — Ed.). As your mission as Larkspur in this game is to free yourself from a castle, I suppose the next title will be Larkspur not trapped any more but has landed himself in more trouble.

The task — getting out of a castle — is not very original. The back on built-in-rescreen is divided into five sections: top left is a picture of Larkspur himself and the title, which remains throughout.

So free Mr Walkoff you are told that you will need a

parachute. Carried top is a view of your location, although there are limited to certain points and vectors; circling is often used for several different locations — for instance, while swimming in the lake, despite moving in different directions, you get the same view.

Top right is the inventory — not in a tabular list but a list of accessible directions, and then the rest of the screen is devoted to location description and your response.

The starting point is a courtyard and you move through rooms such as kitchens, cells, an attic, where there is a parachute which is typically out of reach. As per normal

there are objects around such as a shield, radioactive waste and a herring which turns out to be a red one — I'm sure I've seen that somewhere else in an adventure.

Admittedly entries such as ALL etc. are allowed, although backspacing is replaced by the clear key, which ignores the typed command.

Other standard adventure features include the maze which in this case is a complex of tunnels, and is as frustrating as normal. One thing that I thought would be frustrating was the fact that when you're fatally located against an autorescue of rocks or whatever, you have to load in data from side B of the cassette. However, this is done so quickly that you hardly notice. Anyway, good players can't get killed.

All in all, I'm a bit underwhelmed about this one, as there's nothing here that hasn't been done before. However, it is as apparent as the title would suggest, and what there is is more than a successful as to say The Queen for Life or The Godfather incident. Yet I'm a feeling this little chap might grow on you as he progresses through his troubles, because his creators can certainly construct a good program, they just need a spark of real originality to turn it into a real adventure.

Philip Shaw



Parrot goes down 3-0 to Dragon

Title: *Indoor Football*

Supplier: Computape (also Quickbeam)

Price: £10.00

"This session is mounting" here in the stadium, as the teams prepare, knowing that "at the end of this day" only one will be victorious and leaving in mind that "I only takes a second to score a goal".

Just a few football manager clichés to introduce the setting of this game — the indoor stadium. However, you don't have to wear a sheepskin coat and fluffy jockey pants because you control your team by the tick of a joystick.

Before you are allowed to kick off, however, you have a few tactics to decide. Firstly, the strength of the game is on the relatively short two minutes to a goal including half time. Then you can decide whether to play the computer or a human opponent. Also, there is a choice to the number of players in the team, from seven to eleven. Unfortunately, the opposition has to have the same number of players!

Having made these decisions, you are thrown straight into the battle of the blue-

versus-reds, each displayed as teams of animated players on a sideways scrolling 30-style pitch very similar to International Soccer released several years ago for the Commodore-64. In indoor football you simply get one of your players and his joystick controls only, but if the opposition has possession of the ball, you have to press your joystick's red button when the ball is risk. A warning, however, for certain

strength of the kick is determined by how long you held down the button. The ball will then fly (or trickle) down the pitch to get your super striker on the ball, and to do this you once more press the button. The nearest man to the ball will start to flash, enabling you to nudge him in a goalward direction.

Well, that's the idea. My first attempt, resulted in an interval score of around 10-0 needless

A goal that was so perfect that Peter Beardsley would have given his front teeth for it

to say, I had the nil. Beating a mind around other well-worn clichés that "Football is a game of two halves", I remember the second period only to find that the opposition had lost some of their scoring ability. Full time 3-0.

Several further games against the computer resulted in similar though not quite such bad results. Admittedly, playing on a very slow side of the screen

showed beating human opponents was right, but not as satisfying as trouncing the computer.

Beating human opponents was right, but not as satisfying as trouncing the computer.

Finally, it happened. A long pass descended straight to the feet of my attacker (just two defenders and a chip over the keeper. A goal that was so perfect that Peter Beardsley would give his front teeth to score it — if he had any. Perhaps he already has.

Since then, I happen to add, I have scored more goals, several of them from kick-offs, where sometimes it seems incredibly easy to dissect the whole of the opposing team's defence.

My aftermatch report is that the graphics are good, not much to shout about, but not too simple. The game, though, is in a totally superior league to Grey Foxes, although not as addictive as Superkit.

Meanwhile, I'm still trying to beat the computer. I'm getting there, but to finish with the most famous cliché, until that time I'm "as sick as a parrot".

Philip Shaw



Papa was a rollin' . . . airball?

Title: *Acid-bell*
Supplier: Preston
Price: £2.99
An arcade-adventure called *Acid-bell* you might think you'd have been called *Acidbell 2*. That is a fair description of this program.

The time is not yet a distant planet inhabited by tribes of vicious cyber and peaceful robots who are in grave danger of becoming an extinct-as-the-dodo. You have the three remaining light globes with which to save the species and to do this you guide them along numerous pathways, picking up pieces of jigsaws (I've come to the conclusion that balls must have been used after playful intentions to form a logo at the top of the screen). Get all the pieces and you fly happily ever after, get jumped on three times by the cubes and you're as dead as a moose. I've used such strategy and I don't need to be stressful.

Thus all square pretty much like a flat floor, but there are differences. Some are the same and others are different. The walls are made of stone tiles, some are white and some are dark. There's also a lot of wood. What there is a good deal of wood, there is a good deal of stone. These things don't hang about and the pathways consist of more or less straight lines and convolutions leading you to various places. One will find a small stream, a little pond, a large one.

Control is totally by the keyboard, no joystick input and no digital options to get around with in such a reflex-oriented program. Up is the left is Q, down to the right is A, up to the right is W and down to the left is Z. These directions are necessary because of the 3D style perspective that the game is played in. Additional keys control the space bar to push up pieces of the figure, enter to pause, clear to exit the screen when on a loss and finally break which breaks with a wheel.

© 2000 Blackwell Science Ltd *Journal of Internal Medicine* 247: 395–401

However, if you do return to the closest key or instead where ever you like a life, you can not just put back on the previous ROM's but are allowed back to the starting screen which means you have to spend through several screens and've previously laid down together back to where you originally were. The ironic thing about this procedure is that you often lose another life while waiting for the ROM's to be loaded.

The graphics are in-sepia black and white without any minute detail — like, water and clouds don't exactly need precision artistry, but manage to look impressive by the complexity and depth the pathways suggest. There's about nice little hint to introduce the game, but one thing that has been missed is a score feature, the only guide being the building up of the jigsaw, and this disallows at this point.

The more I play this the more I like it, at first it's a little too quick when using the unfamiliar keys. Once this is overcome through the challenge of exploring more and more algorithms requiring greater ingenuity and speed drives you on. There are forty-one pieces of fissure to collect, with only three lives, so the odds are a bit stacked against you, although you can sail screens without getting all the pieces, so you can play your tactics and draw your margin in advance of completion.

This game has obviously been influenced by *Asterix* and is still very well written, but the moral is that when you look like your father it's hard to follow in his footsteps with your own talents. I'm looking forward to the sequel, though!

1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26



All the answers, but where do you look?

Title: Everything you always wanted to know about COQ
Supplier: Peter Jansen Shovel, LLC, 75 Victoria Road, Parkstone, Rhode 04123 MA
Price: \$19.95

SHOOTER is a late comer to the plains and pleasures of OS-B, I must admit I have been finding it hard going. One possible solution could be Jason Shuster's book *Everything you always wanted to know about OS-B but didn't know where to ask*. Priced at £18.95 it seems a bit expensive, but it does include a disc containing OS-B utilities.

It's not celebrity as performance in the past that looks, moves, etc., with little like this, surely but so is their life. This look is no exception — regular readers of the American magazine *Playboy* will recognize the contents as *MSNBC* OS-6, a regular monthly feature in the magazine. The *MSNBC* is an acronym for *Keep It Simple Stupid*. The series was meant to help *Tandy CoCo* users get to grips with what was then a new operating system. *Janet*

Book is in fact a compilation of three articles reprinted, bound in a plastic clip and with a cardstock cover cover.

The book's biggest failing is that it is not indexed, so it is necessary to read the whole book through to find out the answer to what might be a relatively simple question. For what is obviously supposed to be a reference work, this is a really serious fault (Hart, *How* — 40). To make matters worse, as with our beloved *Granger's Opera*, *Handbook* was subject to the occasional printer's error. These errors are usually corrected in later editions, but in a serious republication like this one of articles, the errors would have been corrected out. As it is, you could spend hours trying to make one of the projects work, only to find that the information is incorrect and is corrected in one of the seven later

A prime example of this is the graphical address book to run forty tasks drives on a Tandy 600-B system (the standard for Tandy is 386). We are told that this is a fairly simple program to

just change the data held at two addresses. Unfortunately, the addresses given are not right, and are corrected three pages later.

One of the more pleasant things about magazine articles is that the writer answers readers' queries. When attempting to learn a new system it is easy to get the impression that only you are having these problems and perhaps it is nearly beyond you. It is comforting to know that others have problems with the system. Anyone who has read the manual supplied with the original system (brought from Dragon Data) will appreciate the way the articles are written. They are in a language that even I can understand (well, almost) and presented in a way that is both well explained and quite readable. The real prize here is that there really is a wealth of information contained in this book. If you are prepared to searching for it, with editions devoted to Macintosh, Personal and the use of the C compiler as well as other tools to the IBM PC.

used. However, there is no real continuity and you might find a large gap between one Plasticid article and the next.

To summarise, this country is a very useful book, spoiled by the lack of any editing or proof-reading. At 199p it does seem somewhat overpriced. However, the size of the market must be taken into account. Five years ago it would have been edited, indexed and printed by the thousand for half the price. How times change! In these days, Jason will be lucky to sell a couple of hundred and will probably only just cover his production costs.

You may have noticed that so far I have made little mention of the children disc. This is because so far I have found no details of what they do or how to use them. By now you will have jumped to what I really need - CDS-9 for absolute beginners.

1000



Memory and assembly

Pam D'Arcy maps assembler tools into the Dragon's memory

FIGURE 1: MEMORY LAYOUT WHEN ASSEMBLING

00000	System workspace (A) 160000 bytes
40000	Test screen (B) 160000 bytes
	DragonDOS disk disk system workspace (C) 160000 bytes
80000	Graphics screen pages (default 48 switch on in 4 pages, 8 80000 bytes) (D) 161000 bytes
A0000	Basic program area (actually starts at 90000) (E)
	Basic program private variables area (space as required to higher addressed memory) (F)
BFF56	System hardware stack (lower as required to lower addressed memory) (G)
BFFD7	Basic string data stack (length=first value of first CLEAR statement) (H)(default 48000 bytes)
BFFFF	Reserved machine code area (length=second value of first CLEAR statement) (I)
	DeluxeDOS disk system workspace (J) 160000 bytes
C0000	Basic interpreter ROM (K)
C2000	Cartridge area (ROM) (L)
C4000	Possibly unused area below end of cartridge software (ROM) (M)
DFFFF- FFFFF	System workspace (N)

YOUR FIGURES

WHEN at this stage a reader has the nerve to admit that he still hasn't managed to assemble anything more than using the assembler software in the machine code articles, one cannot help but feel that he cannot be stupid and something should be done about it! A few other readers have arisen concerning memory, so it seems right to give it another go.

Unmodified Dragons have 65536 individual accessible memory locations—bytes uniquely identified by referring to them as having memory addresses (the postal addresses) 0 through 65535. On a Dragon 32, apart from 256 bytes at addresses 65280-65535, the top half of memory (addresses 32768-65279) can be locked if a read = PEEKed test, rather like removing the piece of plastic from the spine of a cassette to prevent it being overwritten. It is made of material that cannot be overwritten (=ROM, Read Only Memory). Corruptible memory is known as RAM (Random Access Memory). Switching on a Dragon 64 gives an identical memory map to the 32, including the ROM. There is further comment on the 64 at the end of these jottings.

All into memory

I am only really familiar with one assembler (Dragon) so some of the following comments may not be applicable to your software. However, generally, there are two phases: interleaving test (source code) as in the nature of editing a Basic program. There will usually be a SAVE/DISK test facility, and assembly to converting the test (=source code) to machine language (=machine code/binary code). There may be a save binary code facility built in to your assembler or, if using an assembler like Dream, you cut the program and save it like any other piece of machine/binary code (SAVE/DISK or equivalent command).

To achieve an assembly one needs to be able to fit into memory the assembler program and its workspace, the source code (=test), and the resulting machine code (=object/binary code).

The assembler workspace will include what is known as a symbol table or list that is created while the source code is being converted to object code. The symbol list contains all the names that appear in the label column of the test and the memory address/value that the assembler has worked out (=assigned) for it that are subsequently used in the resulting object code.

All these components of assembling need to be in separate areas of memory—if they start spilling into each other, corruption will occur, either because the assembler program itself has been over-

written or because overwriting of the symbols table/object code while assembling will give indeterminate results and almost certainly incorrect object code. These areas also need to be kept clear of system workspace, including disc workspace, to avoid catastrophes such as file corruption caused by involuntarily auto-loading drives.

Figure one shows a memory map split into sections referred to by reference. Not down your own figures for your system on the right-hand side of the chart. Note that sections (C) and (L) are only present if a disc cartridge is attached and only one of them is then relevant depending on whether the DOS software being used is DeltaDOS (L) or other — DragonDOS, Gamma, SuperDOS all utilizing the same workspace area (C). Both types of system use the same length of workspace (6600 bytes) and addresses of areas proceeding following on from these respectively are suitably adjusted.

The documentation of your assembler should give an indication of where in memory text (source code), object code, and possibly its workspace (including the symbols table) are stored. With some assemblers, the text (source code) is made to appear as a Basic program or source code appears alongside Basic statements. If you are using an assembler written in Basic, it may well be that generated object code is stored in the graphics page's memory or one is asked to issue an appropriate CLEAR statement before commenting (or one may be included within the assembler program itself).

Dragon's reply

The following is with specific reference to Dream and in answer to queries arising from its use in previous articles.

All versions of Dream are machine code programs supplied with documentation that includes a memory map for the variant purchased. If the Dream cartridge is being used, the cartridge is plugged while the machine is switched off and does not act as an alien like a games cartridge. Dream is not activated until the appropriate BASIC has been typed in. Before using the cartridge or loading in a cassette or disc version of Dream, a CLEAR command needs to be typed in to reserve memory for the Dream program itself (if not on cartridge) its workspace and text/symbols table/object code areas. Cassette and disc versions of the Dream program are loaded into the highest free area of memory that it can be accommodated in (avoiding DeltaDOS workspace for such systems).

I didn't realise that I had been inconsistent in my articles in that some mention after CLEAR 20029000 whereas others are after CLEAR20034600. The amount of memory claimed has nothing to do with the version of Dream being used but is the amount of workspace reserved. I was using round figures and tend to use 20000 (which is 48320) if typing in a memory value in decimal and 445000 (20480) if using hex. I do tend to use hex more often as assembler language based, if

FIGURE 2 AFTER CLEAR20034600

84F37	System hardware stack (H)
84F39	Basic string data (H)
95000	Reserved machine code area (L)
	One DeltaDOS attached? (L)
95000	Basic interpreter (L)

FIGURE 3 DREAM WORKSPACE ANALYSIS

95000	Not used
95001	Object code (grows as required to higher addressed memory)
	Symbols Table (Starts at lowest address and of text area and grows as required to lower addressed memory)
950FF	Text (source code) statements (Starts at address 950FF and grows as required to lower addressed memory)
95C00	Other reserved Dream workspace (always 3200 bytes long)
95E00	Disk/DREAM program
END	
(C) PAN D'ARCY	20000000

must say that I've spent some years ago on a calculator with decimal hex displays was money well spent; they are still readily available at that sort of price — otherwise for hex/decimal conversion you can always type in the hex of 19400 (decimal number) or 54400 on the Dragon keyboard).

To save then, the example CLEAR20034600 immediately after switching on (no disc drive) leaves memory as in Figure two.

An advantage of the Alphacon system is that all the space in section (L) (95000-9FFFF — 12288 bytes) is available for text/symbols table and Dream space. Dream on cassette or disc otherwise depots this available space. Using my DISK/DREAM, the program created at address 95C00, meaning that 95000-95DFF (3200 bytes) are available for text

etc. Dream uses the workspace dynamically — that is, it doesn't affix artificial limits to the size of each of the text/symbols/object categories of memory. Figure three is a diagram of the machine code area (L) when using DISK/DREAM.

Continuing with Dream as the example, on then loading up, type BASIC and reply N to 'text is memory' prompt. Although I am a great advocate of regularly saving source code to cassette/tape, it is very useful to be able to flip in and out of Dream retaining existing source code in memory (N to prompt) rather than to have to reload a saved file afresh each time. On entering Dream for the first time (or after saving N in order to clear old source from memory), you are presented with a blank screen. This is the 'edit source' mode where you type in and edit source code statements (without line numbers). This starts filling the workspace

memory, growing down in memory addressing from the original highest available memory address (BASIC address 32001-32497 in this example). The symbols and object areas only come into play when assembly is requested—that is, when pressing Break/Enter to assemble the source code currently contained in the text area.

Should this workspace become filled, Dream gives the message FULL, allowing you to quit the program and reserve a larger workspace, say, CLEAR200, 3249999 returning to Dream with Y for test in memory).

Save the current source text then type break/Enter to assemble the program. As it assembles the source, the symbols table will be compiled growing down in memory addressing from the end of the text area then the generated object (machine) code will enter memory starting at the address +1 of the second value of the last CLEAR statement (eg 32501 if the last CLEAR was 32500) — an exception being if the PUT directive has told the

Dream to put it elsewhere in memory.

Assuming that the code has been assembled to address 32501 and the assembler prompt has, press break/Enter to return to the source (edit) screen then press break/Enter again to quit Dream. Now the generated machine code can be saved to cassette (cd) (SAVECD filename, 32501 and address, etc address or equivalent command.)

This article is only intended to be a pointer, so as it is already larger than expected, I cannot add any further detail. However, if you start running short of space, progressively reduce the CLEAR 200 address figure. If OIM occurs, reduce the number of graphics pages (one can achieve POLEARN by a few judicious POKEs as mentioned in Dragon User from time to time). The 250 in the CLEAR statement could be reduced but I do not recommend reducing it to much below 150. DREAMREAM users have the lovely facility to split the source code into sections that can be called from disc for assembling a program whose source code is larger than

could otherwise be accommodated on a Dragon32. Without DREAMREAM, it is up to the user to arrange source code into separately assembled, linkable modules.

The additional RAM can be used with Dream, I have never enquired as to whether Dream is relocatable and suspect that it is as far as load address is concerned but it probably uses 32K mode ROM calls so cannot be used in 64K mode. As I use discs, which do not use 64K mode, just the technique shown in Dragon Answer from time to time where RAM memory is mapped in, copying the contents of the 32K Basic ROM and cartridge (Dragon DOS addresses) onto the identical addresses in RAM. This will leave 32000-40959 (2056 bytes) free for PUTting object code into, leaving more lower memory space for source code text. I write programs in relocatable (position independent) code so that they can run unchanged at a lower address and by using the 32K mode Basic ROM routines have Dragon32 and Dragon4 (32K mode) compatibility.

Winners and Losers

Every month,
Gordon Lee will
look at some prize programming

Hi the mail this month comes a letter from Jim Finlay of Renford.

"I'm knocking on for 47 and had never touched a computer twelve months ago. My son then gave me a Dragon 32 for his birthday (thinks on you, Finlay at it, with a couple of text books and some games to play. I soon wanted to do more than play games and started on a text book and rapidly got bored because its instructions seemed to lead only to trial and error, and it had errors anyway. I simply didn't have guidance on how to do the things I wanted to do. Turned to the official Dragon manual left me even more confused."

"Another textbook seemed a speeded thing; it gave a series of modules, explaining how they worked. There was only one thing wrong with them—they didn't work! Even I managed to spot some of the bugs but had finally to give up on the things. It was at this stage that I found that there was a magazine called Dragon User and promptly became a subscriber."

Jim then makes some complimentary remarks about the magazine and goes on to say that he has learned more about programming from DU than from any other source. This has enabled him to put his programming to the test. His first attempt was the November puzzle — probably the most tricky puzzle set in recent months. However, with enthusiasm undimmed, Jim has since tackled the more recent problems and says that he looks forward to future challenges.

"The difficulty of learning programming (or anything else — ditto) from books alone has been a constant feature of readers' letters asking for advice, and I have given hints and tips in past issues. Clearly, the ability to program even simple routines with confidence is an advantage, not only

in regard of the competition, but for other applications for which commercial software is not available. Listings from books are all very well, provided that they do the task you require, and that they are themselves bug-free.

Books on programming I would recommend to the beginner are:

Easy programming for the Dragon 32 and Further Programming for the Dragon 32 both by Ian Stewart and Robin Jones (Pitman Publishing), and Programming the Dragon 32 by Peter Lafferty (Newnes Microcomputer Books).

The first two of these, which are best used as a pair, explain and enlarge upon most of the commands outlined in the manual. Each is illustrated with a short routine showing the command in action, most of the listings being under a dozen clearly printed program lines. Unfortunately, these may now be out of print and only be available second hand.

However, if there is time to choose just one book (with which to be stationed on a desert island) then my choice would be Peter Lafferty's book. This provides a broad outline of Dragon Basic, and is obviously written by someone well versed in the little idiosyncrasies of the Dragon (and which frequently trip up unsuspecting competitors to the competition questions).

On the subject of programming generally, the important thing is to be thoroughly familiar with each of the commands, and exactly what each one does, and the result obtained. Once understood in isolation, the effect of combining these commands to build program lines, and from there routines, can be readily appreciated. To give an analogy if a physicist

understands the effect of altering the shutter speed on the camera, and the effect of adjusting the aperture, he can combine the two to give him the result that he requires.

Of course, what no book can tell you is how to translate the task to be performed into the program. This is just a matter of experience in which the Basic commands form a 'toolkit', and it is up to the programmer to select the right tools for the job. The knack of 'seeing' how a problem can be adapted into a well-organized, and the important bridge between problem and program. The simpler the routine, the clearer the effectiveness. For example, can you devise a short routine which will exchange the values held in two variables? In other words, given two variables X and Y, place the current value of X in Y and Y in X. (These variables are assumed to hold a different value.) If you are unable to see the solution at once you will probably end up with the same value in both variables, the second value having been lost. However, if the same problem is presented graphically with everyday objects, the solution becomes glaringly obvious: suppose I have a red glass containing white wine, and a white glass containing red wine. The task is to switch the wine so that the colour matches the glass. Clearly a third glass (variable) is needed to save the contents of one of the glasses while the switch is being made. Call this variable Z, and the routine becomes:

```
Z=X: X=Y: Y=Z
```

Little need be said this month about the January competition, which, judging by the number of entries, provided little difficulty. The only 'catch' is likely to be the 'ghost' characters formed by STAB command (see The Answer, Dragon User April 1988, p. 23).

Data grows on trees

Jonathan Cartwright spells out data storage on the Dragon 32

Listing 1

```
10 REM NUMBER(100)
20 REM TREE(1000,2)
30 GOSUB 1000
40 GOSUB 1000
50 GOSUB 1000
60 GOSUB 1000
1000 REM INPUT NUMBERS
1010 LET C=0
1020 PRINT "NUMBER:";C
1030 INPUT "ENTER VALUE 1-999 TO END:";NUMBER(C)
1040 PRINT
1050 IF NUMBER(C)=999 THEN C=C+1:RETURN
1060 LET C=C+1
1070 GOTO 1030
1500 REM CREATE TREE
1510 LET TREE(C,1)=NUMBER(C)
1520 FOR LOOP=2 TO C
1530 LET V=NUMBER(LOOP)
1540 LET C2=1
1550 LET V0=NUMBER(C2)
1560 IF V<V0 THEN GOTO 1620
1570 IF V=V0 THEN GOTO 1670
1580 REM GREATER THAN
1590 IF TREE(C2,3)<V THEN C2=TREE(C2,3):GOTO 1580
1600 IF TREE(C2,3)=V THEN TREE(C2,3)=LOOP:C2=C2+1
1610 TREE(LOOP,1)=V:GOTO 1670
1620 REM LESS THAN
1630 IF TREE(C2,3)>V THEN C2=TREE(C2,3):GOTO 1580
1640 IF TREE(C2,3)=V THEN TREE(C2,3)=LOOP:C2=C2+1
1650 TREE(LOOP,1)=V:GOTO 1670
1660 REM EQUAL TO
1670 NEXT LOOP
1680 RETURN
2000 REM SEARCH
2010 INPUT "VALUE TO BE SEARCHED:";V
2020 LET C2=1
2030 LET V0=TREE(C2,1)
2040 IF V=V0 THEN GOTO 2090
2050 IF V<V0 THEN GOTO 2120
2060 REM GREATER THAN
2070 IF TREE(C2,3)<V THEN C2=TREE(C2,3):GOTO 2040
2080 IF TREE(C2,3)=V THEN GOTO 2150
2090 REM LESS THAN
2100 IF TREE(C2,3)>V THEN C2=TREE(C2,3):GOTO 2040
2110 IF TREE(C2,3)=V THEN GOTO 2150
2120 REM EQUAL TO
2130 PRINT"VALUE ";V;" FOUND."
2140 GOTO 2160
2150 PRINT"VALUE ";V;" NOT FOUND."
2160 PRINT
2170 INPUT"SEARCH AGAIN (Y/N)";A$
2180 IF A$="Y" AND A$="N" AND A$="H" AND A$="h"
2190 THEN GOTO 2170
2190 IF A$="Y" OR A$="y" THEN GOTO 2040
2200 RETURN
2300 REM DISPLAY TREE
```

The main topic I intend to investigate is that of trees. Trees normally mean oak, pine and Christmas, but in computing a tree is a way of storing data. Basic, as such, cannot handle trees, although Pascal can. So what is the use of thinking about it? Well, it's a useful technique that can be "fudged" through Basic, and in any case it won't hurt you to know the theory behind data storage.

If you would care to cast your eyes over **figure one**, you should get the gist of what a binary tree is. Basically, it is a very efficient way of storing and searching data. When the numbers are put in the tree, they are organised so that:

- 1) Numbers greater than that in the current node are put to the right hand side.
- 2) Numbers lower than that in the current node are put to the left hand side.

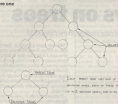
If this is a little confusing, then take a look at **figure two**.

Now we come to the important idea of search through the tree. Because all the values higher than that of the parent node are to the right, and all the values lower than the parent node are to the left, then we can instantly rule out a large chunk of values. If, using the example in **figure two** we are searching for the value 80, then we can instantly rule out all the values less than 80, it is all those to the left. We can continue doing this at every node until we either find the number or decide that it is not in the tree at all. This is a distinct advantage over normal searching procedures, where we would go through every value until we either ran out of values or found the value that we were looking for. Using the tree on small amounts of data you will not notice any great increases in speed when searching. However, when I originally wrote this program on a mainframe, I was able to enter vast amounts of data. When searching such large amounts of data, you will notice a great improvement.

At this point it would seem appropriate to give you a program of some sort. Listing one is that program.

The program simulates a binary tree in Basic. It will allow you to enter up to 100 numbers into the tree, and subsequently search it. I have "grown" the tree as a two-dimensional array TREE(80,3). The 80 is the amount of numbers that you can put

```
3010 FOR M=1 TO 2
3020 FOR V=1 TO 3
3030 PRINT TREE(M,V)
3040 NEXT V
3050 PRINT
3060 NEXT M
3070 RETURN
4000 REM END
4010 END
```



```

graph TD
    2((2)) --- 8((8))
    2 --- 11((11))
  
```



The Beverly Hills Hotel (1001 N. Beverly Dr.) from the west. The hotel, which opened in 1928, is a landmark of the city.

```

13 The first
   value is always
   the parent node
8
10
14
20
13 The other values
   are daughter nodes
4
6
10
20
2

```

Element	Value	Less than point	Greater than point
---------	-------	--------------------	-----------------------

[illegible]

The pictures
refer to the
concrete situation

```

graph TD
    2((2)) --- 1((1))
    2 --- 14((14))
    1 --- 4((4))
    14 --- 13((13))
    14 --- 20((20))

```

The number 11 is to be added to the sum. As it is greater than the present mode 11, we know that it must go on the right hand side. The next mode is the right is 14. Of course, 11, is less than this, so it must go on the left. There is no mode to the left, and so we cannot say anything more about it.

pointer works in the same way. Again this may be confusing, so for all these examples stamped by pointers refer to Figure 10.10.

You might notice that in search is binary tree, all how do you create one to begin with? This is very similar to searching. The only difference is that if the required value isn't in the tree then you tag it onto the end, taking into account whether it should go to the left or right. Figure four is presented for those in trouble.

From your own place in the personal history

and use it. In this form it isn't a particularly useful program. However, it demonstrates the principle of trees and it may come in useful.

To demonstrate how trees can be used on a IBM Dragon, I'm going to give you a more useful program which is, in fact, a working compiler:

If I need to store a large number of words in the database of my Dragon[®], I would be better served if I tried to store each word in a separate string. What? Well, strings take up a lot of memory and are inefficient if there are lots of similarities. So how am I going to do it? Well, I'm going to use an array here. Btw, it's an array, not an array.

Each node of a binary tree may have up to two daughters. In an *n*-ary tree, each node can have up to *n* daughters, where *n* is any number. Discuss at Figure five at this point.

From now on it gets a little harder. For simplicity, I am going to only show you how the word **HOUSE** could be learned in an ordinary case. The "paired code" is going to be A. I'm going to have A daughter B. B has a daughter D, who has a daughter G. Thus the four letters are linked together. Figure 6a should make things clearer. We now know how the letters in each word are linked together. What else do you need to know? Well, let's take the word **HOUSE**.

Figure 10a

```
graph TD; A(( )) --- B(( )); A --- C(( )); A --- D(( )); A --- E(( )); B --- F(( )); B --- G(( )); B --- H(( )); C --- I(( )); C --- J(( )); D --- K(( )); D --- L(( )); E --- M(( )); E --- N(( )); E --- O(( )); F --- P(( )); F --- Q(( )); G --- R(( )); G --- S(( )); H --- T(( )); H --- U(( )); I --- V(( )); I --- W(( )); J --- X(( )); J --- Y(( )); K --- Z(( )); K --- AA(( )); L --- AB(( )); L --- AC(( )); M --- AD(( )); M --- AE(( )); N --- AF(( )); N --- AG(( )); O --- AH(( )); O --- AI(( ));
```

Listing 2

```

1 REM SPELL CHECKER
2 CLEAR10000
3 MAXLENGTH=20
4 DIM B(128,MAXLENGTH+1)
5 GOSUB 10
6 GOSUB 17
7 GOSUB 24
8 GOSUB 33
9 STOP
10 REM ===== CLEAR AREA
11 FOR I=1 TO MAXLENGTH+1
12 FOR J=1 TO 28
13 LET B(I,J,0)=STR$(0,20,48)
14 NEXT J
15 NEXT I
16 RETURN
17 REM ===== MORSE
18 INPUT "ENTER WORD TO BE ADDED TO THE DICTIONARY:";W$
19 GOSUB 21
20 INPUT "ANY MORE WORDS (Y/N)?:";A$
21 IF A$="Y" AND A$="N" THEN 20
22 IF A$="Y" THEN 17
23 RETURN
24 REM ===== SEARCH WORDS
25 INPUT "ENTER WORD TO SEARCH:";S$
26 GOSUB 40
27 INPUT "ANY MORE WORDS (Y/N)?:";A$
28 IF A$="Y" AND A$="N" THEN 27
29 IF A$="Y" THEN 24
30 RETURN
31 REM ===== ADD TO DICTIONARY
32 L=LEN(W$)
33 IF L=0 THEN PRINT "word too short!!!!";RETURN
34 IF L>MAXLENGTH THEN PRINT "word too long!!!!";RETURN
35 FOR I=1 TO L-1
36 S=ASC(STR$(W$,I+1,1000))
37 Y=ASC(STR$(B,I,1,1000))
38 IF S=Y THEN P=I+1 ELSE P=0
39 IF P=0 THEN MID$(B,I,1,1000)=S
40 MID$(B,I,1,1000)=S THEN P=0
41 NEXT I
42 RETURN
43 REM ===== SEARCH DICTIONARY
44 L=LEN(S$)
45 IF L=0 THEN PRINT "word too short!!!!";RETURN
46 IF L>MAXLENGTH THEN PRINT "word too long!!!!";RETURN
47 GOTO 48
48 FOR I=1 TO L-1
49 S=ASC(STR$(S,I,1,1000))
50 Y=ASC(STR$(B,I,1,1000))
51 IF S=Y AND MID$(B,I,1,1000)=S THEN P=1
52 IF MID$(B,I,1,1000)=S THEN P=1
53 NEXT I
54 IF P=0 THEN PRINT "word spell incorrectly. try it!!"
55 PRINT "word: " & S$ & ", found: " & P
56 INPUT "DO YOU WISH TO ADD THIS WORD TO THE DICTIONARY (Y/N)?:";A$
57 IF A$="Y" AND A$="N" THEN 50
58 IF A$="Y" THEN 50
59 GOSUB 21
60 RETURN
61 REM ===== PRINT TRACES
62 FOR I=1 TO MAXLENGTH+1
63 PRINT "PASS:";I
64 PRINT "SEARCH:";FOR J=1 TO 28;PRINT CHR$(95+J);
65 NEXT J
66 PRINT "M:";M
67 FOR J=1 TO 28
68 PRINT CHR$(95+J);B(I,J,1)
69 NEXT J
70 RETURN

```



Figure seven shows this, along with *abcs*. This is getting more complex. Before we try to write a piece of software, I'm going to drop you in at the deep end and add the word *BAGS*. This can be seen in figure eight. It may look as though *BAGS* is a valid word while *BAGB* is not. This problem can be avoided by having a 'terminator' in *abcs* to signify that the word can end here or clearly on. In addition to this we also need a 'special terminator' to signify that the word *MUST* end here. Look at figure nine for an example. Now you can see that things are much more clear.

You may or may not have noticed that using this system of storage you can 'load' the dictionary into thinking that a word exists when it doesn't. For example, looking at figure nine you could be forgiven for thinking that the word *BAGB* exists. This is one of the drawbacks with this kind of data storage. However, when using real-worlds it does not present too much of a problem.

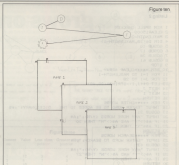




This system allows you to store more words than you would normally be able to do in 32K.

All that remains is for me to reveal how I'm going to 'tidy' this messy tree. I can best explain that by a demonstration. Let's take the word DADA. I'm sure you can now imagine how this is going to look as a tree, but for those with a limited imagination I have included it in figure ten. Also in this busy drawing are three pages. It is using these 'pages' that I presently use. On the first page you will see that there is a number (zero-ordinates) DAs and As are our first two letters, so that explains the location. The 1 means that D followed by A is allowed in a word. The second page has a 1 at (A,T). Again this means that A followed by T is allowed. On the third page we have a three at (T,A).

The three means that this is the end of the word. It is one of our 'special terminators'. So, to recap, in our dictionary we



are allowed D followed by A followed by T followed by A to end. This spells the word DADA. Now we could add all the other thousands of words in the dictionary. I fear, however, that our beloved editor might frown upon the idea as she doubtless wants something else in this month's issue. (At this precise moment, dear people, one issue of *Oregon Journal of Art* Cartwright's words would be the answer to my prayer! However you (the readers) might not be too happy about that and, in effect, we wouldn't win the drawing. So onwards...).

And now for the last! Yes, that 72 line listing entitled *Testing Two* is my special checker! And you all thought it would be a six page job, didn't you? Well, this shows

you how efficient data storage can show your programs and NOT your lifespan. You can enter words into the 'dictionary' and subsequently search for them.

At the moment, the program allows words of up to twenty letters in length, although this can be altered, memory permitting. If the word that you're looking for isn't there then you are given the option of adding it. You might like to try to get the program to load in your word processed files and let you know which words are not spelled correctly. I've not done that because a) it is too easy to haven't got the time and b) I wrote the routine for *Electronic Author* then all of you with *Starwriter* will tell me for neglecting you.

The Dragon on Prestel

Ken Smith introduces BT's Prestel database

THE day I bought a modem, I was filled with visions of being able to pull information in from all over the globe, chatting to other computer users and looking seats for Dire Straits without standing in the queue. Unfortunately, life is rarely that simple and having solved one problem by acquiring the equipment, I came to the next: who do I ring? After much thought, the most logical step seemed to be to try out Prestel, a huge database, operated by British Telecom.

Now, I would have thought

that since BT operate both Prestel and the telephone system, either the operator or, at least, directory enquiries would have a list of access numbers. That might have been the case if BT really wanted people to use the system. In reality, the only number I managed to get this way was Prestel head office in London, a number which is permanently engaged. Even the demonstration number, given in one of those interior magazines which never mention *Dragon*, proved to be

useless. All this time I knew that all the local access numbers were listed on Prestel but would only be available to me once I had made contact. Eventually I found what I was looking for by logging on to a bulletin board and pleading for help.

Love/hate

That was how my love/hate relationship with Britain's biggest database began. A lot of data has flowed along the bus since then, I have now become

a regular subscriber to the system and have come to know its pains and its pleasures.

Information is displayed in the form of pages on a variable system which allows the use of text and graphics on the same page. Unfortunately, as designers have to be content with a monochrome display since only MODEM gives a high enough resolution, to be sure of using a local number you will need a VDD (VDD/576) modem. VDD (VDD/576) is being phased in

Continued on page 38

Screen Compressor

Jonathan Cartwright squeezes screens into smaller spaces.

HAVING seen screen compressor programs on much larger computers than the Dragon, I decided that it might be a good idea for me too. I'm surprised that it hasn't been done before, as when you've only got 32K to play with, 16K for a full-screen is a large chunk of memory to write out.

Before I start with a program, it might be a good idea to explain how a compressor works. Generally speaking, there's quite a lot of blank space on a graphics screen. Obviously, there is less if your picture is complex. This vast amount of data is very wasteful on memory. For example, you could draw an 8x8 top half of the graphics screen, and it would still take up 64K. What my program does first is to "read" the screen. What it's looking for is a collection of bytes, one after another, which are the same. If it doesn't find such data, then it merely stores the screen byte for byte elsewhere in memory. However, if it DOES find what it is looking for then it can get to work. What it does is store a "runway or repeat" value to signify that the next few bytes relate to compressed data. The run-

ny value it uses is the number 255. After this, it places the value of the bytes which are the same. After that, it places the number of bytes which are the same.

The compressed screen is placed elsewhere in memory so that it can be called up, in its decompressed state. I have made three screen stored at locations 26623 onwards, but you can put it anywhere you like, so long as you CLEAR space and alter the programs accordingly.

The compressor program itself is written in Basic. This is partly for my convenience and it also means that the program is easily readable. To use it, first you've typed it in, LOAD up the screen you wish to compress. Then RUN the program. It will, after quite a long time, tell you the start and end addresses of the compressed screen, and with the percentage saving and the length of the screen in bytes. Because I use a disk drive I have defined the top of the graphics screen as location 5072 and the bottom +1 as 9010.

Use this program without DOGS attached, subtract 1000 from these values.

Remember to alter the machine code decompressor too.

The test program is a machine code subroutine which does the job of putting the screen back together again. Again it assumes that the compressed screen is at location 26623, but this can be altered to suit your requirements. I have located the routine at 2070, but you can put it anywhere you like. The routine is simply EXECed into action.

If you want to get adventurous then you could store several screens at different locations in memory, and run a "slideshow". This would simply involve changing the store location used in the machine code, before calling up each screen.

The compressor program can make some fantastic savings, although I have sometimes got savings of as little as 4 per cent. My letter-head now takes up nearly half the space it used to, and if I have compressed screens in disc, instead of expanded ones, I can save memory there as well. I hope that this program is of use to people.

```
4: CLEAR 300, 35523
5: REM **SCREEN COMPRESSOR
6: REM **©1980 STARSHIP SOFTWARE
10: PEEK4,1:GOTO 11
20: LD=2072:REM TOP OF SCREEN
21: L1=9016:REM BOTTOM OF SCREEN
30: ST=26623:REM STORE LOCATION
55: ST=ST
40: IF PEEK(LD)=PEEK(LD+1) AND PEEK(LD)=PEEK(LD+2) THEN GOSUB 1000:GOTO 40
50: IF PEEK(LD)=255 THEN REM 9008-1000
60: POKST,PEEK(LD)
70: ST=ST+1
80: LD=LD+1
100: IF L1=L2 THEN GOTO 40
110: GOTO 2000
1000: REM=0:REM NUMBER OF BYTES THAT ARE THE SAME
1010: L2=LD
1020: IF PEEK(L1)=PEEK(L2) AND REM<255 THEN REM=REM+1:L2=L2+1:GOTO 1020
1030: POKST,255:REM 95-COMPRESS VALUE
1040: ST=ST+1
1050: POKST,PEEK(LD):REM NUMBER OF BYTES
1060: ST=ST+1
1070: POKST,REM:REM NUMBER OF BYTES
1080: LD=LD+8:ST=ST+1
1090: RETURN
2000: PRINT "SCREEN COMPRESSED"
2010: PRINT "START ADDRESS:";L1
2020: PRINT "END ADDRESS:";ST-1
2030: REM=ST-1-8:REM LENGTH OF SCREEN IN BYTES
2040: PRINT "BYTES USED:";REM
2050: PRINT "SPACE SAVING:";"(100-(861/6144*(100-REM))%"
2060: END
```

Screen Compressor

©COPYRIGHT COMPRESS LTD 1983 DASH 2.00

(3) ALL

2401	30	774
2400	40	089 9214
2405	50	104 8367
2402	60	121 82822
2407	70	138 174
2408	80	155 2205
2409	90	172 84274
2400	100	189 7
2407	110	196 7

2411 2780

2412 24

2413 1000

2414 2079

2415 2000

2416 2780

2417 2000

2418 2000

2419 2000

2420 2000

2421 2000

2422

120 8000 874 5

140 8000 874 5

160 8000 874 5

180 8000 874 5

200 8000 874 5

220 8000 874 5

240 8000 874 5

260 8000 874 5

280 8000 874 5

300 8000 874 5

320 8000 874 5

340 8000 874 5

360 8000 874 5

380 8000 874 5

400 8000 874 5

420 8000 874 5

440 8000 874 5

460 8000 874 5

480 8000 874 5

500 8000 874 5

520 8000 874 5

540 8000 874 5

560 8000 874 5

580 8000 874 5

600 8000 874 5

620 8000 874 5

640 8000 874 5

660 8000 874 5

680 8000 874 5

700 8000 874 5

720 8000 874 5

740 8000 874 5

760 8000 874 5

780 8000 874 5

800 8000 874 5

820 8000 874 5

840 8000 874 5

860 8000 874 5

880 8000 874 5

900 8000 874 5

920 8000 874 5

940 8000 874 5

960 8000 874 5

980 8000 874 5

1000 8000 874 5

1020 8000 874 5

1040 8000 874 5

1060 8000 874 5

1080 8000 874 5

1100 8000 874 5

1120 8000 874 5

1140 8000 874 5

1160 8000 874 5

1180 8000 874 5

1200 8000 874 5

1220 8000 874 5

1240 8000 874 5

1260 8000 874 5

1280 8000 874 5

1300 8000 874 5

1320 8000 874 5

1340 8000 874 5

1360 8000 874 5

1380 8000 874 5

1400 8000 874 5

1420 8000 874 5

1440 8000 874 5

1460 8000 874 5

1480 8000 874 5

1500 8000 874 5

1520 8000 874 5

1540 8000 874 5

1560 8000 874 5

1580 8000 874 5

1600 8000 874 5

1620 8000 874 5

1640 8000 874 5

1660 8000 874 5

1680 8000 874 5

1700 8000 874 5

1720 8000 874 5

1740 8000 874 5

1760 8000 874 5

1780 8000 874 5

1800 8000 874 5

Starship Software

24 System, Paul, Charles, Mike, George, 24 18.



The above system-based has a space saving of 47.4%

Rele-Ball



RELE-BALL SYSTEM

By Rele-Ball system based a computer 24.24 saving.

Continued from page 18

and is already available on many numbers. Where VSE is available the system will automatically adjust to match your equipment so the only way to find out is to try. The system is so designed it is easy to get lost, so much so that I find it necessary to provide a directory, listing all the information providers (PPI) as they are called. When a system this big is largely made driver then I can take a long time to get it where you want to go, so there has to be a way of taking short cuts. This is normally done by entering the relevant page number. Now a page number has eight digits and if you enter less, then the system simply adds enough zeros to bring it up to eight. Of course you can always look up the number in the directory, but wouldn't it be much easier to enter "MAIL" for the mailbox facility or "weather for the Met Office."

Prestel was originally conceived as a business database. However, the advent and rapid growth of the micro computer changed all that. A whole

new market appeared and it was decided to expand the network to exploit this. There are now bulletin boards to cater for a wide variety of interests, many of them having nothing to do with computers. For instance, one section deals with pop music and allows you to have your say if you think Rick Astley's records are a pain in the earholes then you can say so and leave your vote for all to read. Dead users (and Astley fans?) — (X) have their own section, known as "Bastley", where they can publish riddles pertaining to the hearing impaired. I have even found an advertisement for "Chrissie" and yes, you can book tickets for concerts at Wembley.

Game changes

Most of the games available on the system are charged for in addition to the subscription and line charges. Shades (a multi user dungeon game), for instance, costs an extra 2p per minute. Many of the games and quizzes offer cash prizes

but it is my experience that they are more designed to take your money than to entertain.

Working as I do, at the sharp end of a service organisation, the thing that I find most useful is the telex link. Most companies have a telex machine and using this system, my company can leave me a day's work and I can call it off at my convenience. (Where do you keep your Dragon? I can send my pet that evening and it will be waiting for them in the morning. Telex link is part of Mellow, the electronic mail system. This was apparently included as an after-thought. Perhaps this is why it is a bit crude, for instance no word-wrap, no real edit facility and typing at a transfer rate of 76 baudsecond can be painful at slow.

Subscribing to Prestel will set you back £500 a quarter. Think of it as five pence a week and you will see it is not bad value for money. During peak usage times, it is 10p to 50p, there are additional charges of 5p per minute, outside these times there are no time

charges. There are additional charges for some services but there is adequate warning before you incur any additional cost. To get a real taste there are free demonstrations available on the numbers listed below. When prompted for your ID number, enter 4514444444, then for your password use 4444. This should get you started, so why not give it a try, it will only cost you a phone call.

Finally, listed below are numbers that are supposed to give the demonstration. At the time of writing the only one that does not work is the London number if you live in the south and cannot get any joy, then try 0207 455400. Bees of Lark and if you do subscribe, why not give me an M50 on 019900408 and tell me what you think.

Prestel access numbers

South (01) 688 1111
Midlands (02) 688 1111
North (05) 688 1111
Scotland (04) 688 1111

Copy to order

Rudy Duyck presents a versatile file-copying program for Dragon Data and Eurohard DOSes.

COPY is a utility to copy any number of files any number of times from one drive to another, and works with all Dragon Data and Eurohard DOSes, including Super-DOS. It is written in Basic with one very short machine code routine (relocatable) used to scroll the screen downwards and stored in the string MCB in line 18, using the data in line 1900. Because of this it is impossible to use the program as it stands with any form of high resolution text screen. COPY makes its own directory of the source disc and offers you the options of copying all files, a number of selected files, or all files except for a number of selected files. Selection of files is straightforward through the use of a cursor positioned with the up and down arrow keys. The order of copying files can be determined in this way too. I found the information needed to write this program in Grosvenor Software's Dragon-DOS: a Programmer's Guide. If you want to understand how COPY creates its own directory, you will also find interesting information in Paul Dagmar's article *In the Directory in the May 1987 issue of Dragon User*.

The COPY utility will ask you to enter the number (1 or 2) of both the source and the target discs (drives). These must be different; it is unfortunately impossible to copy files from one disc to another using only one drive under Dragon-DOS. After being given this information COPY will access the source disc and create a directory in the array H2B. Only valid files will be included.

When the end of the directory is reached (found by checking for the end of directory flag in line 180) the Dragon generates three MCBs to which you assign your three options: copy all files, select the files you want copied, or select the files you want omitted. Next you are asked whether you want all the files copied to be protected. The default option, also selected if you press ENTER and indicated with a star, is no protection. If you selected the second copy option you will now be able to determine the order in which files are to be copied. With both the second and the third options you get the first screen of the directory and a cursor which you direct with the up and down arrow keys (which auto-repeat). You select the files by pressing the space bar and you end the selection process and begin the copying with ENTER. Pressing "J" will

erase all the selections you have made so far and enable you to start all over again if you have made a mistake. The screen will scroll up or scroll down as needed.

After the copying is done you are asked whether you want to copy the same files once more, to copy different files from the same disc, to copy files from a different disc, or to stop. If you want to copy from the same disc the program continues by using the directory already in memory. All the copying takes place in one go and COPY will warn you with three BEEP's when it is finished so that you can do something else during the copying (it does not take very long, though). Should any error (except a disk or connection FULL error) occur, the copying ends and you get an error message. If the error occurs because the target disc is full (GF error, code 148) or because the directory is full (FD error, code 148) you will be asked to insert another disc for the other files to be copied onto. The main limit only partially saved file list copied is killed to clean up the full target disc.

You can use COPY as a faster alternative to BACKUP (especially in all versions of Dragon-DOS after 1.0: they are slower!) by using a blank formatted disc as target disc and selecting the copy all option. This makes it also possible to backup between different disc formats, if that is what you have the standard Dragon Data disc drive unit with a higher-capacity drive added (in my case a double-sided, 4096-track personal drive) this comes in quite handy! It is also possible to have all files protected in one go.

COPY enables you to re-organise discs by copying files from one clustered up disc to a newly formatted disc in a different order. Even if you choose to copy the files in the same order as on the original disc you should be able to save space because often-used discs are not especially fragile in their state of space. You may wonder why I have not built in an option to sort file names alphabetically. The reason is very simple. I do not see the use of this since quite often a particular software package consists of several programs or include data files and all of these would end up protected all over the disc and the directory if you sorted the files alphabetically. In this way the logical connection between the various components of a package would be lost.

It is easy to merge existing discs onto a blank one using COPY too. A final use for this utility is for people, eg members of a computer club, who want to distribute programs they have written in a number of fellow students, or who want to circulate their translations in disc (as some clubs do). You can do this far more easily and in a less time-consuming manner with COPY because the program allows you to copy the same items time and again.

For those Dragon users who are interested in the way COPY is programmed let me draw attention to a number of features which may provide inspiration for your own programs. The routine in lines 10 to 130 to produce a directory can quite easily be adapted and adapted to your own programs that aim to manipulate disc files, eg to full, restore, or delete programs or masses. The routine in lines 630 to 870 checks for keys being held down without using INKEY\$. The major advantage of this use of the keyboard roll over table in lower memory (300-340) is that the keys auto-repeat. A final remark concerns the error routines in line 1900. Most people seem to believe that error trapping is only useful for preventing wrong user input from interrupting a program. However, error trapping can also be used to advantage in disc operations, eg to prevent writing to the same or full disc (or when aborting the whole program). If you direct the program from an error to a routine that can diagnose the error and take action accordingly you produce a program that offers more user-friendly and that can deal with disc errors that occur frequently when handling files. In COPY the error routines handle the disc and directory full errors that one must expect to occur when copying from a larger-format disc to a smaller-format disc for instance.

I shall answer any questions in connection with COPY you might have if you include a self-addressed envelope and one International Reply Coupon if you live in the EEC or two if you live outside it. If you do not feel up to the job of typing in the listing, I can send you a disc with the program on it. If you send me 275 Belgian francs by International Money Order (no cheques or postal orders please, the costs doubled by the bank) are too high for a small sum paid this way. My address is Rudy Duyck, Nieuwelanden 8-C, B-8290 Brugges 2 (NL, Antwerp, Belgium).

```
5 "copies files from drive B to drive d1
6 "copy right rudy duyck 15.03.87
10 CLEAR: CLEAR:10000:ERROR:GOTO10000:DIM HMB(140),W(140)
15 FOR I=1 TO 17:READ H:HC=H&"CHR$(H):PRINT I:GOTO100
20 HMB(I)=H:W(I)=H&"CHR$(H):HC=H&"CHR$(H):STR$(I),I:W(I)=HMB(I):GOTO100:PRINT
21 COPY HMB TO H2B:IF PR THEN PROTECTION H2B
```


Write: ADVENTURE

Pete Gernard generates an error message

THERE can be no greater crime for the adventure writer than to release a game that hasn't been fully tested into the unsuspecting public. People who have paid good money for a program have a right to expect that program to be working correctly. A bad game will not make a living, a plumber who doesn't do his plumbing properly will not get paid, and equally so the adventure writer who doesn't care to check a game does not deserve the support of the adventure playing public.

And yet, games released by the largest of companies often contain an unbelievable collection of mistakes. It is not the fault of the company handling the distribution, since they will be paying money in one fashion or another to the company or individual who wrote the game; it is the first place. Under contract, more often than not, and contracts are meant to be legally binding. This little programmer leaves errors unchecked that not only is he treating the public with contempt, if the contract contains some such phraseology as "will deliver a fully debugged, error-free adventure" then he is also breaking the law.

Thus it is of vital importance to have your games checked and playtested before either selling them yourself or submitting them to the company who asked for them. This should preferably be done by someone who knows nothing about the game, and while the initial and final checking should be done by the programmer the all-important middle stages should be handled by someone else. Let me spread an article such as this, for example. I know what it contains, and am therefore likely to miss the odd mistake or two. In other words, I will probably see the world I want to see rather than what is actually written down. I suppose the programmer will almost inevitably play the game as a reluctant (not player), and not go down all those wonderfully obscure paths that make the playing of a competent adventure so enjoyable.

Seized up

With the ever-increasing complexity of adventure games more obvious paths can lead to mistakes that are often hilarious if caught before the game is released, but a death trap if they are seized upon by a reviewer in one of the popular magazines. Such a reviewer could well be my brother Mike or I, and as one of the games that we produced by working as a team we made some narrative mistakes. All, fortunately, caught before the game was actually released.

It was a two part adventure (based on Homer's epic *The Odyssey*). Mike did the design for the game, and I did the programming, and we tested it between us. This worked quite well, for he didn't know how

the program was put together and I was just following instructions blindly and thus didn't know the correct route to get from problem (a) to problem (b). In order to fit everything into memory I was doing some text compression. This involved changing words and commonly occurring groups of letters, so that one character replaced three or more. Thus 'tree' would be represented as 'tre', and 'by' another, and so on. Mike was rather astonished in one early version of the game while he was exploring a king's palace and went somewhere where he shouldn't have. Immediately he was 'surrounded' by several large gophers! Ah, the perils of coding two letters mixed up. Wonderful word through gulfon might be, a sort of snail.



haling creatures, I would imagine, it would not do for a reviewer to find a gulfon marauding through his copy of the game.

Flags, counters, call them what you will, they play a vital part in any adventure. A flag incorrectly set, or just not set at all, can produce a minor disaster. In order to escape from the Cyclops' cave in *The Odyssey* our intrepid hero has first of all to blind his ogre and then climb under the belly of a sheep and get out again at the end of the things. The blinded Cyclops only feels the tops of the sheep and fails to find Cyclopses clinging grimly on underneath one of them. When he has successfully escaped he can then tell the sheep to go away, and carry on the game. Also and flag is forgot to set the 'sheep-gone away' flag, and so whenever Odysseus went he was followed by a flock of sheep are grazing close by you'. He could scale a castle wall, and there would be a flock of sheep on his side. He could swim to his ship, and find a flock of aquatic sheep grazing on the water. That was soon corrected, but not before a subtle hint from Mike had put me

into the problem. It was something like 'I see/gophered these ***** sheep! Literary talent will out, I always say.

A brick dropped

The role of a playtester is not an easy one, and I always think that they should be highly rewarded if they perform a competent job and successfully eliminate all errors. One of the most common mistakes (by the programmer) seems to be the (silly) that can never be dropped.

[DROPPED BRICK

Oh, you drop the brick.

[BYE MOTHER

You are carrying a brick.

[DROPPED BRICK

Oh, you drop the brick.

And so on, for ever and ever. A little bug not getting set somewhere. Likewise we have errors that can be continually taken, and are always present in the score description. Conversely, we have objects that are there, and yet you are told they are not there when you try to take them. Again, objects that vanish into limbo when you drop them and which are never seen for the rest of the game. The playtester must attempt everything possible with every object in every location. In an ideal world, and if the game is question doesn't contain a save feature then send it back to the programmer with the politely worded request that you will not test the game until it does.

SAVE and LOAD are two commonly used words that easily lead to errors, and thus I prefer to use SAVE and RESTORE, making sure that the gamer recognizes much more than the first few letters of each word if possible, or have a special condition for the word RESTORE on its own. Players might want to have a rest, not restore a previously saved game. They might want to load a revolver, not load an old file. Once more, the playtester must read these out.

I was fortunate enough on one game to have a nearly efficient playtester. The game had originally been written on a micro computer by my brother Mike and I, and then I transferred it over to another one. This meant that it needed testing all over again, and so out went, with the inevitable sinking feeling that I might, after all, have overlooked something. Needless to say I had. The game concerned those two indolent characters Holmes and Watson (now out of copyright and open to anyone to write an adventure about), and featured a dog called Toby. On examining Toby you were told that you could see some fleas. Examining the fleas told you that you could see some smaller fleas. Having a bit

more memory left on the second computer than the first one continued the examination of files to a never-ending degree, so that examining the smaller files led you to be told that you could see some that were even smaller. This went on and on, and thank goodness for the competent playtester. He examined more files than I had ever imagined possible, with the end result being that the machine finally ran out of memory and crashed. That was definitely a case of having too many bugs in a game!

Never be tempted to put anything like that into your games, no matter how much memory you might have left, without putting some kind of limit on things beyond which the files will not get any smaller, or whatever. Someone, somewhere, will still feel free and find an error.

When it comes to searching out errors, which after all is what a playtester should be doing, one of the things that is often overlooked is the way that save of a game. Adventure players are a contrary lot, and on that first save have you checked for someone typing in COME? or AGAIN, if you've allowed them to repeat a command? Or HISTORY from MAM when there isn't a previously saved position? A program crash on the first input will not entice anyone to your adventures.

Removing every mistake is, I would imagine, impossible in a highly complex game. You are just not going to be able to

foresee all the possibilities, and that is why you ignore pretending at your peril. A good reviewer will spend many hours on an adventure before writing his or her review, and if they find anything within that time that strikes them as an error, particularly if



it's a crucial one, then believe me that is what the reviewer will concentrate on. I know, I've done it myself! It's always difficult to come up with something new when writing a review, but a critical mistake is a godsend. Don't give anyone the opportunity to lambast your game by leaving in silly errors.

Something that is often overlooked, and again here a playtester is essential, is your spelling and grammar. The most common use of ITS and IT'S (THINK) or THEIR, is irritating. Always have a dictionary by your side when writing or testing a game, and any doubtful words should be consulted straight away. It's (he) always difficult writing about grammar, because someone will pick you up on the inevitable errors, but adventure games should contain themselves with sticking to the simple things and getting them right. By and large a balance between "You are on a road" and a Barmed Latin-type 500 word noun description that contains just one sentence.

The simple things are always the best. "You are tramping noisily along a gravel path" is far better than "You are making a lot of noise as you walk along a path with a lot of gravel on it. That's what is making the noise, the gravel, as you walk along it, on the path!" I've seen them, I've seen them, too many! A good playtester will tell you off for writing something like the latter example, and quite right too.

A good playtester is invaluable, and you should never, ever, sell or submit an adventure until it has been tested by at least one person other than yourself. You know how a game should be played, but the great public that control your financial future do not, and they have a right to do so in free games. Test it!



I was going to devote this month's column to some of my story solutions, involving our old friend Professor Doornick exploring a little something called the Pyramids of Doornik an adventure that many people seem stuck on (so is that it)?

Steam up

However, a letter arrived in tandem with an adventure program written by the author of the letter, who signed the adventure reviewed within three scored pages. Perhaps it will be reviewed by someone else, I don't know, but in these days of slowness by Oregon software houses there are certain ways to go about giving a game a reasonable chance of getting reviewed. That an adventure will not be reviewed? The author of this letter has taken the letter

course, and I hope that all you adventure fanatics out there will bear with me for a while as I digress into the world of right and wrong.

If I am asked to review a game of the adventure variety like those at least a partial map and solution. The time involved in exploring a new game is such that, unless the game proves to be particularly wonderful, it's not worth doing. Reviewing an adventure game takes up a lot of time for a day, and without a helping hand you are in the same position as someone who has just bought it all the shell or ordered it through the post. Having said all that, I will stick to a game as long as the accompanying letter is of at least four pages.

A good number of years ago I converted three adventures that I'd written so that they would work on the Oregon. I had little knowledge of the computer, but my

publishers insisted on the conversion as the adventures were accompanying a book. Three listings were given in that book, but the publishers really wanted a tape as well, in case people couldn't be bothered typing in page after page of listings. Fair enough, and I spent a reasonable amount of time getting those three adventures up and running on the Oregon. To send a game in for review that has been recorded over one of my old tapes is not the best way of entrusting yourself to the person who is hopefully going to review the game.

Exclaim disclaimer

The letter that came with this afternoon's tape gave me a brief summary of the plot. Nothing scoring there, but perhaps a great adventure lurked beneath the said

outline. After this summary came the quote (a brief and simple adventure by my standards) "Okay, perhaps the chap is a mile big-headed, but that we all could fall into that particular category at times. I am told that someone else is 'fucked on her early review-copy?' So? I haven't seen the game yet."

The exclamation marks are all in the letter, by the way, they can't miss.

I am told that the adventure features a **WISIT** command that gives a list of verbs used by the game. Good idea. Such verbs as "unlock & brief" are quoted, alongside the author's comments "rather more additional?" Integers have been doing it for years, albeit not on a Dragon.

Another quote: "Danger situations exist, prompted by 'What should I do?', and only one response works" if there is anything that I hate in an adventure is games more than anything else it is the instant death situation. It is so pointless. Adventures are meant to be a test of a person's logic and, perhaps, lateral thinking, and to find yourself being killed off for no apparent reason other than that the game's cryptic underhand a particular sentence structure irritates me in the extreme. One response in an instant death situation? Come on, there are better puzzles to be made than that.

I shall quote again: "Two maze areas exist, the mine area and the marsh. Thorne needs mapping as it has loads of blind areas with no exits and all the exits are hidden; these being those I think" I dislike mazes at the best of times, but can see their point in a well-structured adventure and have indeed spent many an enjoyable hour working my way through a particularly difficult maze. However, blind areas, no exits, exits hidden? An awful way of quickly programming a difficult adventure.

Get it writo

Finally I get the comment "I hope you'll give it a fair bit of space in D. User, please, all I have not been mentioned for months..." obviously not been reading the review in *Dragon User*—PG. 1... 2000 Software need a boost as they cannot afford D. User advert rates? You have not been mentioned for months, and you will not be mentioned unless I get a reasonable letter. Give it a fair bit of space! There's nothing better than free advertising, you know who you are, so climb off that pedestal and admit that other people are of interest in the adventure world as well as yourself.

If you go for that matter anyone else could like a game reviewed here, then please send me a brief a partial map and solution, don't demand acres of column space, don't put pointless quotes in it, and above all don't boast.

And what if I've got all that off my chest, I shall climb down from my pedestal as well and, somewhat out in the face, continue with the Adventure Trail.

Space now forbids a story solution to Pyramids of Doom, so as this is being written in a rush plenty more deadline forward two weeks. Either that or failed to get

last month's copy... leaves space for comment from the little Must Be Obscured... "OK, OK, I apologise! Next time be a dear and put it in an obituary more than two inches apart? Please? You will have nothing less than the full solution, courtesy of the admirable Martin Edwards and Richard Edwards."

Y: Get the pole (a shovel) and GO POOL. Get key, get liquid (in container) and E, N, E: DRS, get the small key, S, DRG again and GO HOLE. Unlock the door then return U: again. Drop the shovel, get the stone. Unlock the main door and go in, then drop both keys and light the flashlight. You are in the Pyramid of Doom.



Z: Open the sarcophagus. Go M, E and get the key. Go M, S, Sand play the flute here. Examine (use) LOOK the fireplace to find a coal (there's old Magicist Scrolls play this role through and use it in Guit of Thieves!). LOOK fireplace (some action) and examine ashes to find a necklace. Get the NECKLACE. Go passage then go H and E. Read the hieroglyphics (if you can spell them) and drop the stone and the necklace. Get the jelly (what a wonderful thing).

Q: W: that N and drop the jelly. The crystal will eat the jelly, dropping a pearl, so you should get the PEARL, go halfway R, drop pearl, W, H, and also you should drop the flute here, too.

Q: Go W, R and go sarcophagus. D, clean Herodotus's sword, but don't treasure it! and your water to extinguish burning leaves. Now you can get the TAPETESTER. Go above, look box, look box (yes, again) and get the iron claws. Get the skull and remove it. Go to GOLD TEETH.

DYH, S, and drop the skull. WCHH H, H, and wear the gloves. You'll probably drop a (heavy) flask, you can pick it up and reveal it. Hit doorway and remove the glass.

Q: Get the rope and go S, S, go ladder, then N again. Throw rope (up to the ceiling to help you get up there) and throw ruby (to

help kill the pharaoh) then get the GAPPY-HOLE.

T: I shall LOOK FURNISH then look at the explorer list. Get the PIN and the CARP. M: then go W and D. Drop the carpet and get the saw.

S: N, U, W, N. Saw table, drop saw and get NECKLACE. E and go archway. Drop the tapestry, teeth, necklaces, pin, carving and sapphires. Go W and H.

S: W, get saw, S, get key, get key. Go sarcophagus. D, S, go ladder, S, get rope and open the chest. Get the CROWN. Put the chain. Go stairs, saw table, drop saw, wear gloves, unlock coffin, drop both keys. Look coffin and get the BRACELET. Go window, get BAR, go W, D, M, D, N, M, go door, feel floor and get COIN. Go E, light flashlight and get the SCARAB. Go W, W, light the flashlight, S, U, W, N, E, go arch, E, drop saws, coin, bar, crown and bracelet.

W: SCORE.

Adventure mags

And there you have it, traps and chaps, please, the complete solution to an adventure that conjures up some other obscure problems at times, but all should now be revealed in order to enable you to fly away another game as 'completed'.

Finally for this month we have two changes of address to note (apart from the subscriptions). I've mentioned before in this column a couple of home-produced adventure magazines known as *Scottrayer* and *Adventure Probe*. Both were set up by the hard-working Sagan Sharkey, who deserves much praise for all the effort she put into launching them and getting them onto a regular monthly basis. If you want complete maps and solutions, *Scottrayer* and *Adventure Probe* are the other best, if you prefer the shorter approach with scene links, tips, reviews, stories, a veritable plethora of adventure information, then *Adventure Probe* is the magazine for you. Once upon a time they were both produced in the great metropolis of Wigan (Challenge Cup Winners, well-known local), but recently, owing to a variety of circumstances, they have moved elsewhere. For *Scottrayer*, you will need to contact John Barnsley at 32 Merivale Road, Rising Brook, Salford, Salford, S7 11HAA. I have the first part of that address, thanks to something out of Lord of the Rings, written by that chap whose name I used not to be able to spell. For *Adventure Probe*, your editor is now Marilyn Rodriguez, (please, despite the name), who resides at 24 Mossy Lane, Llandudno, Gwynedd, LL30 1UE. A single-page address when you get used to it. Sample issues are currently available at £1.25 for *Scottrayer* and £1.00 for *Adventure Probe*, if you fancy sampling their wares.

End of space, as usual. To the unnamed person who sent the game in for review, don't despair. To that person, and the rest of you, bye for now.

A lifetime in 48 days

How long to make your first million? says Gordon Lee

Prize

Reviewed in last month's *Dragon* (p.9), we now have ten copies of *Super Numb* from Omega Software for the stars of the July competition.

Rules

It isn't what you say, it's how you say it. Say it with your Dragon: print out your working, add any comments you wish to add, put them in an envelope marked **JULY COMPETITION** and send the lot to the usual address which is now (as we remind you, *Dragon* Publications, 40 Alexandra Road, Hounslow, Middlesex TW9 4HP).

And now the fairer side — *Super Numb* is tough! I wouldn't have time to think of a solution. Let's think. Everybody has a favourite number. Or an all-favourite number. Tell us about your favourite (or unfavourite) number. The best ones get a prize.

April winners

Teddy's one, this, However, a fair number of readers had the right answer up their sleeves. The lucky winners are:

I.A. Newman of Appleton, S.A. Stoddart of Clevedon, Graham Barber of Sutton Coldfield (the man who has everything except a new printer ribbon), R. Haine of Salford, Don Robertson of Epsom, Patricia Hill of Cranston, Bedford, D. Harley of Cleckheaton, Dave Lennar of Glasgow, Arian Henderson of Bromsgrove and Fred Wilks of Hatfield. Denis M. is spreading his love rumours about the telephones again, but loves by a nose (see, there is a man with a new ribbon).

Some of the telephones were too close to survival for comfort. I observed that the best ones were mostly from non-winners. Can Denis be right after all? But not the one saved. The ribbonless but victorious Barber presents:

"The Editor said, 'They'd like their drink better if it were decolourised.' I thought one said, 'They're likely to think better if they were decolourised.' So I said, 'Off with their heads'."

An understandable mistake! said Alice.

Commendations to everyone who mentioned heads, and everyone who mentioned decolourised (well, the Editor an apple).

The prize this month is a gift bag of their recent releases, *Underbelly's* of Crotch and *Pyramidventure* and discount vouchers from *Dragonair* Services. Selections will be on their way soon.

Solution

See opposite page.

WHEN the British mathematician G.H. Hardy called on the Indian mathematical genius Srinivasa Ramanujan, he is reported to have remarked that the number of his taxi, 1729, was not a very interesting one. "Not at all," replied Ramanujan, "it is the smallest number expressible as the sum of two cubes in two different ways!"

While not suggesting that this is typical of encounters between mathematicians, it does illustrate the fact that some numbers are more interesting than others. But what exactly constitutes an 'interesting' number? Can any number be said to be 'uninteresting'? Of course, a number need not be interesting purely in a mathematical sense. No doubt it will have a personal 'top ten' of favourite interesting numbers — even if they are only the ones we use to do the football pools.

Amongst the lower numbers it is not difficult to find an association with any given number. For example, the number 10, the biblical might denote the 10th element, water, the basic element on which life depends. To the historian, that same number might conjure up thoughts of the sixteenth Henry VIII, or even the six King Georges, while the musicologist would think of 'Les Six', a group of French composers with their own distinctive style. The mathematician would undoubtedly note that 10 is the smallest 'perfect' number — that is its whole number divisors (1, 2, and 5) also add up to its own value.

But what about 'uninteresting' numbers? Are there any of those? Let's suppose that we make a list of all numbers, starting at 1 and progressing upwards. On the left-hand side of the page we put all of the 'interesting' numbers. This, in its early stages, would be fairly extensive. Sooner or later we would come to a number which, despite all our attempts to find some interesting aspect about it, we could be unable to do. Suppose that this number is 32,768. We have already placed 32,768 on the left-hand list as being interesting as the precise number of bytes that is to be found in a 32K RAM such as the one on the Dragon! However by placing this number on the right-hand list it now becomes 'interesting' as it is the first number to appear on this side. This compels us to declare it more, after all, interesting, so we have to delete it from the right and place it on the left side of the page. This leaves the right-hand list vacant once again — almost uninteresting numbers is placed there, when it too will need to be transferred. Thus, by this logical (or illogical) process all numbers become interesting — so matter how high you care to go.

Selection of high numbers reminds me of the story, supposedly true, of the American teacher who told his class that it was impossible to count up to one million in a

normal life span. Whereupon the mother of one of his pupils proved him wrong by not only doing it but also producing the result typed out in full. This, presumably, was in number form rather than in words. But even so (allowing for a space between each number) there would be nearly seven million typed characters in the final list, occupying 1000 double-sided sheets of paper. Even at a typing speed of five characters a second, working eight hours a day, and allowing no time for thought, the task would take about 48 days. If you have a couple of reams of printer paper to spare, so could the computer do the job in a fraction of the time!

It would be interesting to speculate the speed the task would be typed out in the form of words — i.e. beginning one, two, three, four, and continuing up to nine hundred and ninety nine thousand, nine hundred and ninety nine one million. A conservative estimate would indicate that this would require 80 million characters — an increase of almost tenfold on the figures given above!

This month's competition is to devise a simple program to convert a number input in digital form into its 'word' equivalent. The routine should be capable of handling numbers of up to nine digits — i.e. in the range 1 to 999,999,999. So if the number input was 123456789, the printed display would read:

ONE HUNDRED AND TWENTY THREE MILLION, FOUR HUNDRED AND FIFTY SIX THOUSAND, SEVEN HUNDRED AND EIGHTY NINE

Usage will be judged on compactness and their ability to produce reasonable English phrases for the range of numbers specified.

Finally, the answer to Srinivasa Ramanujan's problem given earlier is 12 cubed, or 10 cubed plus 1 cubed. Both add up to the same total, 1729. Readers who are interested might like to try out the following listing to compute this answer. If the program is left running it will produce a number of other values greater than 1729, that are also the sum of two cubes in two different ways.

```
10 A=3
20 FOR B=1 TO A
30 T=A*A*A+B*B*B
40 FOR C=1 TO A-1
50 IF C=D THEN G0
60 J=T-C*C*C
70 B=Int(J/(A*A*A+B*B*B+1)/3)+1
80 IF B=D*B*B*B THEN
PRINT A;"+"B;"="C;"
"="A;"+"B;"="C;"
90 NEXT C
100 NEXT B
110 A=A+1:GOTO 20
```

The Answer

This is Gordon Lee's own
reaction to the April campaign
and page 28 for results.

ANSWER: The cards are lying clockwise from the outside in, so 2, 3, 4, 5, and 6.

Solution: In the listing the values of each of the possible combinations of cards are stored in the array `Cards`. `Cards` is the first card, running clockwise round to `Cards` (the Queen). Card "K" is at `Cards`.

The three FORTHEART hospital lines 2016-20 represent cards 1210, 1275, and 1281. Since these have been given values and 1210 can be calculated, as we say, and that 1275 and 1281 have the same total as 1210,

and $C[5]$. Providing that $C[4]$ is in the range 1 to 13 and is not 4, the computation proceeds. Line 60 calculates the value of $C[3]$ as we know that the queue and its neighbors total the same as $C[2]$ and the circle either side of 4. Again this must be in the range 1 to 13. The next step is to test for valid alliance counts and reject all values not in the range 02.

The new information tells us there is only one pair of cards in the routing of lines 120 to 160 checks each card against each of these offices to record the number of calls.

We then test to see if there is a rising period in the current values. Note that in line 210 we only report sets of values in which there is no rise. This information as given does not exclude the possibility that there is a rise of rises.

If the program is run it prints out six sets of different values. However, the final clue states that if you know the value of `card1[0]`, you could find the answer. (Only in one case (when '0' is a four.) does this result in a unique answer. For any other value of 'i' there will be more than one possible

```

100 DIM C(6)
110 C(0)=0
120 FOR A=0 TO 12:GOTO 140
130 FOR B=0 TO 12:GOTO 150
140 FOR C=0 TO 12:GOTO 160
150 C(4)=C(5)+C(2)+C(3)
160 IF C(0)=1 OR C(4)=12 OR C(4)=9 THEN 320
170 C(0)=C(5)+C(6)+C(1)+1-C(2)+C(4)+1
180 IF C(0)=1 OR C(4)=12 THEN 320
190 T=C(0)+C(2)+C(4)+C(5)+C(6)+C(1)+C(3)
200 IF T=9 THEN 320
210 GOTO 120
220 END

```

```

220 FOR F=1 TO SIZE:PAIR=F TO N
230 IF C(F)=C(F+1) THEN PAIR=PAIR+1
240 NEXT F
250 IF PAIR<1 THEN GOTO 260
260 MINE=0
270 FOR F=1 TO N
280 IF C(F)=2 THEN MINE=MINE+1
290 NEXT F
300 IF MINE=0 THEN GOTO 310
310 PRINT C(1);C(2);C(3);C(4);C(5);C(6);C(7);C(8);C(9);C(10);C(11);C(12);C(13);C(14);C(15);C(16);C(17);C(18);C(19);C(20);C(21);C(22);C(23);C(24);C(25);C(26);C(27);C(28);C(29);C(30);C(31);C(32);C(33);C(34);C(35);C(36);C(37);C(38);C(39);C(40);C(41);C(42);C(43);C(44);C(45);C(46);C(47);C(48);C(49);C(50);C(51);C(52);C(53);C(54);C(55);C(56);C(57);C(58);C(59);C(60);C(61);C(62);C(63);C(64);C(65);C(66);C(67);C(68);C(69);C(70);C(71);C(72);C(73);C(74);C(75);C(76);C(77);C(78);C(79);C(80);C(81);C(82);C(83);C(84);C(85);C(86);C(87);C(88);C(89);C(90);C(91);C(92);C(93);C(94);C(95);C(96);C(97);C(98);C(99);C(100);C(101);C(102);C(103);C(104);C(105);C(106);C(107);C(108);C(109);C(110);C(111);C(112);C(113);C(114);C(115);C(116);C(117);C(118);C(119);C(120);C(121);C(122);C(123);C(124);C(125);C(126);C(127);C(128);C(129);C(130);C(131);C(132);C(133);C(134);C(135);C(136);C(137);C(138);C(139);C(140);C(141);C(142);C(143);C(144);C(145);C(146);C(147);C(148);C(149);C(150);C(151);C(152);C(153);C(154);C(155);C(156);C(157);C(158);C(159);C(160);C(161);C(162);C(163);C(164);C(165);C(166);C(167);C(168);C(169);C(170);C(171);C(172);C(173);C(174);C(175);C(176);C(177);C(178);C(179);C(180);C(181);C(182);C(183);C(184);C(185);C(186);C(187);C(188);C(189);C(190);C(191);C(192);C(193);C(194);C(195);C(196);C(197);C(198);C(199);C(200);C(201);C(202);C(203);C(204);C(205);C(206);C(207);C(208);C(209);C(210);C(211);C(212);C(213);C(214);C(215);C(216);C(217);C(218);C(219);C(220);C(221);C(222);C(223);C(224);C(225);C(226);C(227);C(228);C(229);C(230);C(231);C(232);C(233);C(234);C(235);C(236);C(237);C(238);C(239);C(240);C(241);C(242);C(243);C(244);C(245);C(246);C(247);C(248);C(249);C(250);C(251);C(252);C(253);C(254);C(255);C(256);C(257);C(258);C(259);C(260);C(261);C(262);C(263);C(264);C(265);C(266);C(267);C(268);C(269);C(270);C(271);C(272);C(273);C(274);C(275);C(276);C(277);C(278);C(279);C(280);C(281);C(282);C(283);C(284);C(285);C(286);C(287);C(288);C(289);C(290);C(291);C(292);C(293);C(294);C(295);C(296);C(297);C(298);C(299);C(300);C(301);C(302);C(303);C(304);C(305);C(306);C(307);C(308);C(309);C(310);C(311);C(312);C(313);C(314);C(315);C(316);C(317);C(318);C(319);C(320);C(321);C(322);C(323);C(324);C(325);C(326);C(327);C(328);C(329);C(330);C(331);C(332);C(333);C(334);C(335);C(336);C(337);C(338);C(339);C(340);C(341);C(342);C(343);C(344);C(345);C(346);C(347);C(348);C(349);C(350);C(351);C(352);C(353);C(354);C(355);C(356);C(357);C(358);C(359);C(360);C(361);C(362);C(363);C(364);C(365);C(366);C(367);C(368);C(369);C(370);C(371);C(372);C(373);C(374);C(375);C(376);C(377);C(378);C(379);C(380);C(381);C(382);C(383);C(384);C(385);C(386);C(387);C(388);C(389);C(390);C(391);C(392);C(393);C(394);C(395);C(396);C(397);C(398);C(399);C(400);C(401);C(402);C(403);C(404);C(405);C(406);C(407);C(408);C(409);C(410);C(411);C(412);C(413);C(414);C(415);C(416);C(417);C(418);C(419);C(420);C(421);C(422);C(423);C(424);C(425);C(426);C(427);C(428);C(429);C(430);C(431);C(432);C(433);C(434);C(435);C(436);C(437);C(438);C(439);C(440);C(441);C(442);C(443);C(444);C(445);C(446);C(447);C(448);C(449);C(450);C(451);C(452);C(453);C(454);C(455);C(456);C(457);C(458);C(459);C(460);C(461);C(462);C(463);C(464);C(465);C(466);C(467);C(468);C(469);C(470);C(471);C(472);C(473);C(474);C(475);C(476);C(477);C(478);C(479);C(480);C(481);C(482);C(483);C(484);C(485);C(486);C(487);C(488);C(489);C(490);C(491);C(492);C(493);C(494);C(495);C(496);C(497);C(498);C(499);C(500);C(501);C(502);C(503);C(504);C(505);C(506);C(507);C(508);C(509);C(510);C(511);C(512);C(513);C(514);C(515);C(516);C(517);C(518);C(519);C(520);C(521);C(522);C(523);C(524);C(525);C(526);C(527);C(528);C(529);C(530);C(531);C(532);C(533);C(534);C(535);C(536);C(537);C(538);C(539);C(540);C(541);C(542);C(543);C(544);C(545);C(546);C(547);C(548);C(549);C(550);C(551);C(552);C(553);C(554);C(555);C(556);C(557);C(558);C(559);C(560);C(561);C(562);C(563);C(564);C(565);C(566);C(567);C(568);C(569);C(570);C(571);C(572);C(573);C(574);C(575);C(576);C(577);C(578);C(579);C(580);C(581);C(582);C(583);C(584);C(585);C(586);C(587);C(588);C(589);C(590);C(591);C(592);C(593);C(594);C(595);C(596);C(597);C(598);C(599);C(600);C(601);C(602);C(603);C(604);C(605);C(606);C(607);C(608);C(609);C(610);C(611);C(612);C(613);C(614);C(615);C(616);C(617);C(618);C(619);C(620);C(621);C(622);C(623);C(624);C(625);C(626);C(627);C(628);C(629);C(630);C(631);C(632);C(633);C(634);C(635);C(636);C(637);C(638);C(639);C(640);C(641);C(642);C(643);C(644);C(645);C(646);C(647);C(648);C(649);C(650);C(651);C(652);C(653);C(654);C(655);C(656);C(657);C(658);C(659);C(660);C(661);C(662);C(663);C(664);C(665);C(666);C(667);C(668);C(669);C(670);C(671);C(672);C(673);C(674);C(675);C(676);C(677);C(678);C(679);C(680);C(6
```

Communication

We'll share your problem on the COUPON BOOK (make it as brief and legible as possible) together with your name and address and send it to Communication, 48 Alexander Road, Hockessin, Delaware 19930.

100

1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 2680, 26

1000

Adventure Contact

To help puzzled adventurers/author, here is an Adventure Helpings — simply fill in the coupon below, stating the name of the adventure, your problem and your name and address, and send it to Dragon User Adventure Helpings, 49 Assembly Road, Mountain Meadows, TN 37133.

	1980	1981	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029	2030	2031	2032	2033	2034	2035	2036	2037	2038	2039	2040	2041	2042	2043	2044	2045	2046	2047	2048	2049	2050	2051	2052	2053	2054	2055	2056	2057	2058	2059	2060	2061	2062	2063	2064	2065	2066	2067	2068	2069	2070	2071	2072	2073	2074	2075	2076	2077	2078	2079	2080	2081	2082	2083	2084	2085	2086	2087	2088	2089	2090	2091	2092	2093	2094	2095	2096	2097	2098	2099	2100	2101	2102	2103	2104	2105	2106	2107	2108	2109	2110	2111	2112	2113	2114	2115	2116	2117	2118	2119	2120	2121	2122	2123	2124	2125	2126	2127	2128	2129	2130	2131	2132	2133	2134	2135	2136	2137	2138	2139	2140	2141	2142	2143	2144	2145	2146	2147	2148	2149	2150	2151	2152	2153	2154	2155	2156	2157	2158	2159	2160	2161	2162	2163	2164	2165	2166	2167	2168	2169	2170	2171	2172	2173	2174	2175	2176	2177	2178	2179	2180	2181	2182	2183	2184	2185	2186	2187	2188	2189	2190	2191	2192	2193	2194	2195	2196	2197	2198	2199	2200	2201	2202	2203	2204	2205	2206	2207	2208	2209	2210	2211	2212	2213	2214	2215	2216	2217	2218	2219	2220	2221	2222	2223	2224	2225	2226	2227	2228	2229	2230	2231	2232	2233	2234	2235	2236	2237	2238	2239	2240	2241	2242	2243	2244	2245	2246	2247	2248	2249	2250	2251	2252	2253	2254	2255	2256	2257	2258	2259	2260	2261	2262	2263	2264	2265	2266	2267	2268	2269	2270	2271	2272	2273	2274	2275	2276	2277	2278	2279	2280	2281	2282	2283	2284	2285	2286	2287	2288	2289	2290	2291	2292	2293	2294	2295	2296	2297	2298	2299	2300	2301	2302	2303	2304	2305	2306	2307	2308	2309	2310	2311	2312	2313	2314	2315	2316	2317	2318	2319	2320	2321	2322	2323	2324	2325	2326	2327	2328	2329	2330	2331	2332	2333	2334	2335	2336	2337	2338	2339	2340	2341	2342	2343	2344	2345	2346	2347	2348	2349	2350	2351	2352	2353	2354	2355	2356	2357	2358	2359	2360	2361	2362	2363	2364	2365	2366	2367	2368	2369	2370	2371	2372	2373	2374	2375	2376	2377	2378	2379	2380	2381	2382	2383	2384	2385	2386	2387	2388	2389	2390	2391	2392	2393	2394	2395	2396	2397	2398	2399	2400	2401	2402	2403	2404	2405	2406	2407	2408	2409	2410	2411	2412	2413	2414	2415	2416	2417	2418	2419	2420	2421	2422	2423	2424	2425	2426	2427	2428	2429	2430	2431	2432	2
--	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	------	---

1998



1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26

CLASSIFIED ADS

DRAGON 32, other drivers, software, books. (450 040) Tel. Rochester 875-0000

DRAGON 82. DragonDOS, Touchmaster pad, joystick, Shimadzu/CFA80 printer, magazines, books, software (games and utilities) i-486 will open. SAE for list, or ring 800-800-0000.

© 2004 Pearson Education, Inc. All rights reserved. Printed in the United States of America. This publication is protected by copyright. Permission is granted to reproduce this document for personal or internal use, on the condition that the copier pay the stated fee directly to Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923.

DRAGON 84, Coming this fall, drive, controller, printer, cassette player, selection games £750 VHS Spill, Graham 02572 80044 machines.

NEW Curious disc camcorders, CSD plus \$2 postage. Dragon DOD plus update ROMs, and Dataconverters, ITDSD. Drives from \$43. Send envelope and two x10p-stamps to: PPHC TECH Systems (SBS), 25 Polaris Road, Danbury, Conn. 06810-0001

HOUSE moving sale: Sir. 5.75m disc drives, controller kits, computers, SAE for list. Mic Systems, 114-Crescentway, Burlington Heights, Markham L3R 9L4

ACCOUNTANCY program by goldstar for Edit+, 12, to Michael McCullough, 109 Oregon Road, Detroit, MI, 48206-24.

[illegible]

Reprints available from the publisher at a special price.

[illegible]

1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26

1000

10¢ per word. Addressers must be paid if required, including post costs. Please send this form to Classified Dept., Dragon Publications, 49 Alexander Road, Haverhill, Massachusetts 01830.

Dragon Answers

If you've got a technical question write to Brian Cade. Please do not send a S&M as Brian cannot guarantee to answer individual inquiries.

Save to disc

I would like to be able to change the load and save game routines in some of my machine code programmes the door rather than the tape. Could you please tell me how to create the equivalent of the SMT "file" start, end, entry and SMO commands in machine code?

San Antonio
7th International Avenue
Stevenson
90001-1000



Echoes of Tron to printer

I would very much appreciate TON devoted to the printer, with or without screen display. I had hoped that your "rule screen to printer" thing, if never put in two ago would do the trick, but it won't work if DDCs connected, as both use the 350 hook. Is there a way with DDC connected, please?

A. Davis
100 Broadway Drive
West Des Moines, IA 50319

the hypothesis to impact functions while
a statistically significant trend was

1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 2038, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 2061, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 2072, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083, 2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 2095, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 2106, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117, 2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 2129, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 2140, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151, 2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 2163, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 2174, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185, 2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 2197, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 2208, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219, 2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 2231, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 2242, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253, 2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 2265, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 2276, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287, 2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 2299, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 2310, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321, 2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 2333, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 2344, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355, 2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 2367, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 2378, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389, 2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 2401, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 2412, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423, 2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 2435, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 2446, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457, 2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491, 2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525, 2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 2548, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559, 2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 2571, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 2582, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593, 2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 2605, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 2616, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627, 2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 2639, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 2650, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661, 2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 2673, 2674, 2675, 2676, 2677, 2678, 2679, 26

1000

1000

1000

100

THE above routine does not work because the function definition should not be a string, but rather normal Indentation Basic. Therefore, you cannot enter a new function while the program is running (as you cannot indent and store it). The only solution is to have a number of separate subroutines to define all your required functions, and call these as necessary.

The routine listed below will do the trick. It echoes screen text to the printer, but only if `TERM` is activated (by checking the flag at location `074`). Also, the hook at `004` is copied and used to call the routine, so it will work with or without `TERM` attached.

From the basic loader program to install the code, and then load the program you want to trace. The assembly language listing of the code is also given for anyone that's interested.

The simplest way of doing this is really a lot of a hassle, but it works. Set up the statement pointer at 1000101 to point to a listing containing the required name and save address, length etc. (This must end with a zero byte), then call the Basic SAVE command to save. The listing below shows how to do this (SAVE's address is 14091, LOAD's is 14410 in BruggenBOS 1.0). A call to 120601 is made after the SAVE to close the file, normally.

You can use a similar technique to simulate the LOAD command from machine code.

ADDITIONAL that other files (basic, binary etc.) load and save without problems etc. appear to have a problem with streams that I have not come across before. The examples were standard cassette 10 min calls, and so there should be no special problems present that is a very short header length which previously published papers (see below).

Have any of our readers had the same problem and found a cure? If so let me know and I'll publish the solution.

Non-stop DREAMing

After loading a tape straight through the Oursart editor and assembly package, it starts OK but it never shows when to stop. I have tried all the poses given in your replies in previous inquiries, but to no avail. All the poses refer to altering the head; my problem appears to be a lack of 10% motion.

Dr. William
and Mary
College
Petersburg, Virginia
22404

Input on the run?

All the moment I am working on a program which, among other things, contains mathematical functions. However, I cannot find two-argument functions, while the program is running. I have tried the following, but it does not work:

THESE RESULTS ARE DISCUSSED IN SECTION 4.

[illegible]

```

00 ADDI THREE, ZERO, TWO-SEVENTEEN
01 ADDI FOUR, ZERO, SEVEN
02 CLEAR TWO, THREE
03 TWO TWO-SEVENTEEN TWO-THREE
04 THREE TWO-SEVENTEEN TWO-THREE
05 THREE TWO-SEVENTEEN TWO-THREE
06 THREE TWO-SEVENTEEN TWO-THREE
07 THREE TWO-SEVENTEEN TWO-THREE
08 THREE TWO-SEVENTEEN TWO-THREE
09 THREE TWO-SEVENTEEN TWO-THREE
10 THREE TWO-SEVENTEEN TWO-THREE
11 THREE TWO-SEVENTEEN TWO-THREE
12 THREE TWO-SEVENTEEN TWO-THREE
13 THREE TWO-SEVENTEEN TWO-THREE
14 THREE TWO-SEVENTEEN TWO-THREE
15 THREE TWO-SEVENTEEN TWO-THREE
16 THREE TWO-SEVENTEEN TWO-THREE
17 THREE TWO-SEVENTEEN TWO-THREE
18 THREE TWO-SEVENTEEN TWO-THREE
19 THREE TWO-SEVENTEEN TWO-THREE
20 THREE TWO-SEVENTEEN TWO-THREE
21 THREE TWO-SEVENTEEN TWO-THREE
22 THREE TWO-SEVENTEEN TWO-THREE
23 THREE TWO-SEVENTEEN TWO-THREE
24 THREE TWO-SEVENTEEN TWO-THREE
25 THREE TWO-SEVENTEEN TWO-THREE
26 THREE TWO-SEVENTEEN TWO-THREE
27 THREE TWO-SEVENTEEN TWO-THREE
28 THREE TWO-SEVENTEEN TWO-THREE
29 THREE TWO-SEVENTEEN TWO-THREE
30 THREE TWO-SEVENTEEN TWO-THREE
31 THREE TWO-SEVENTEEN TWO-THREE

```