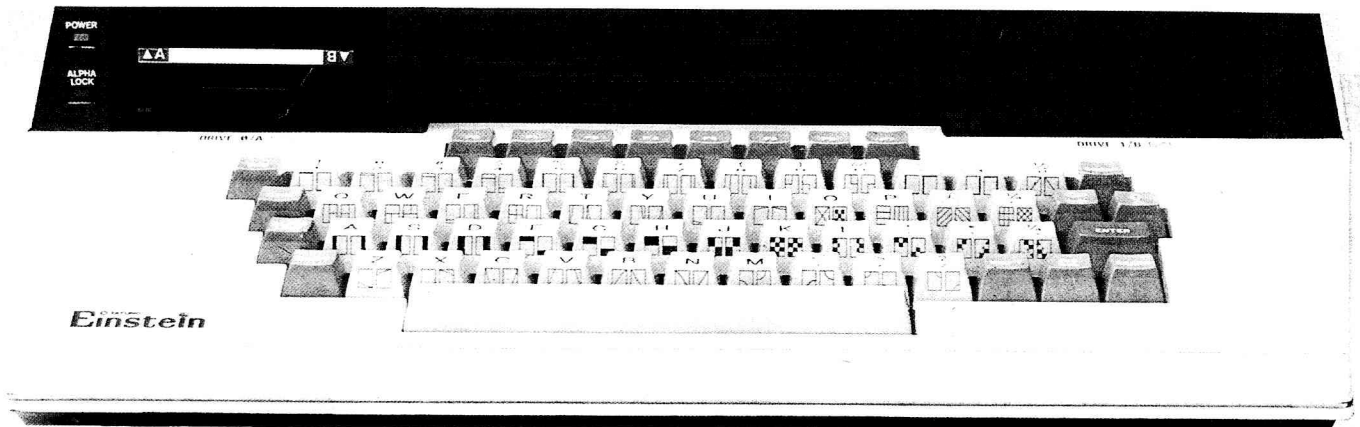
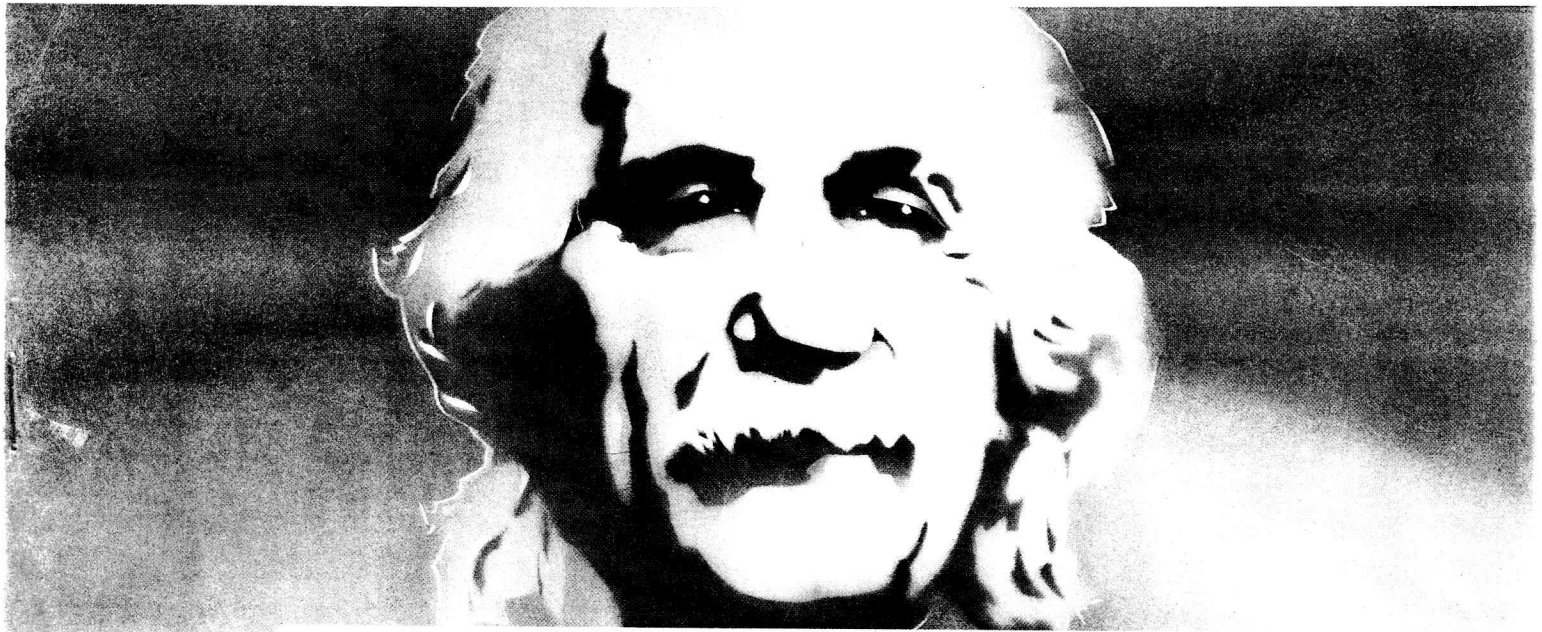


Albert's Bi - Monthly Review



November 1990. Show Issue

Albert's Bi-Monthly Review
November 1990...Show Issue

Well Hello again! Once again I find myself writing another magazine, this time I will try to make even more variation! In this fun packed magazine I have got games reviews, reviews on disc drives and even reviews of IBM 486 compatible machines!

First of all, a bit of explanation! This as I said is the show edition, it is a bit like the limited editions that you get with posh cars, that is that there are only 150 of these magazines in existence at this moment in time, there will be more printed, however it is not really important to know that! If you are a first time reader you will see 2 things that members won't. At the end of the magazine are 2 extra pages the first page describing the group in detail and the last offering you a chance to join the group for a measley £7.00 a year!

Right! Back to business. As many of you know there is one of the biggest computer shows for "Alternative" computers this month. This show is held in Stafford's Bingley Hall. It has proved to be bigger and better than the "All Format Show" and a much more popular event. It should prove to be a show not to be missed.

Well, I have just completed writing 2 new packages for the Einstein, the first being Tubes 6.6, I have added many new functions to this package, and I hope that I have almost perfected it now! (That's what they all say, Ed!). The second package is a joint package between Roy Prime (Me), Ted Cawkwell and Brian Parson. This is an interesting game pack including Dice (A version of Y***zee), Pontoon, Einstein Golf, Romulans, Pairs ,Inbetween and Plank. This is good for the kids, however it gives the Adults time to have a quick play as well!

If you need some cheap software this month, check the bargain pages, you might be a little shocked as to how much software I actually have for sale now, however this only goes to show that I try to do my best for the group!

A quick Apology. For everybody that has been experiencing delays. Sorry folks but with the number of enquiries that I'm dealing with every day, and the fact that I'm working hard at work means that it takes me a long time to get round to answering letters, so please be patient!

I hope you enjoy the magazine.

Roy Prime.

Teac...The Review!

Teac are one of the leading disk drive manufacturers, they produce everything from the Einstein 3" drives to the some of the old style 8" drives!

The drive that I'm going to review is the 5.25" drive. Model FD-55GR.

The TEAC drives are well designed and appear to be very reliable. The standard model is half height (About 6 inches wide by 1 and a half inches in depth by 8 inches long).

The connections are the same as the Einstein drives, one 2 inch edge connector and a 4 pin power male power socket, so you can run one drive from the Albert's power supply, however this is not a recommended idea!

When you buy the drive you will find that you don't get a lot for your £55. The drive comes with some very nice instructions, however is not cased, has no leads and only gives details of a £25 power supply unit! If this might put you off a little, I have written down the total cost of getting an up and running 5.25" Teac Drive.

Actual Drive £55 - £65 depending on supplier.
Disc case £7 - £15 again depending on the supplier.
Power Supply £10. A good power supply (I.e Byte Drive which can power 2 disc drives) can cost about £20
A suitable DOS to run the system £15 - £50.

The total unit cost me £90 including the dreaded VAT. This system may not look very pretty, however it works well.

What benifits does a TEAC drive offer?

Well, for a start it gives you anything from 790k to 834k storage on 1 floppy disk (A lot of space! Ed), and for the greedy, you can "Daisy Chain" them and have 2 linked together to Albert giving you a massive 1580k to 1668k (Depending on Dos System used). They are also quick and quiet.

Software Reviews

Very little has come out on the Einstein recently, so I decided that it might be a good idea to review some of the "Oldies". One of the best "Oldies" that I know of is Elite. Elite was originally produced by Acornsoft for the Acorn Electron, now however it has been re-written and "Converted" to many machines, from silly little machines such as the ZX Spectrum to Atari ST's, Amigas and even versions for the latest PC's.

I will review 2 versions of the game. Firstly on the Einstein, secondly on an IBM compatible.

Elite (Einstein Version) by Merlin Software.
Price £19.95.

This program has very much of a "Cult" following. I have spoken to many people that have gone from Elite on the Electron to Elite on the Amiga, Einstein and PC!

Elite is basically a shoot 'em up game, however, it is not like the usual Shoot 'em ups where you sit around for ten minutes, this one takes a lot of patience!

When you look at the manual you begin to realise why it can take so long to complete the game! First you have to buy yourself some equipment so you can go about doing some damage! However getting that equipment isn't as easy as it sounds. At the start you are "docked" in the Space Station orbiting the planet Lave. You are given a ship and 100 cr (Credits), it is then that you find out need a lot more money before you can even consider shooting at anything! To make money you need to trade, when you are in the space station press K and you will see a whole list of prices. In some planets you will see that there are VASTLY varying prices. For example in Lave computers might cost 98 cr per ton, while in Zaonce computers might cost only 58 cr. So you would buy from Zaonce and sell to Lave, simple! However nobody EVER lets you trade in peace and there are times when it will all go wrong, for example if you decide that profits are too small and you start trading in Narcotics or Firearms, the local Police might get a little annoyed. Unfortunately Police do not just try and arrest you, they will try to destroy you. Another example when things might go wrong is if you "Accidentally" destroy a friendly craft. You will find that you will become a fugitive and nearly everybody will try to destroy you! To give you some example of how long a task that you have ahead of you, to give yourself an almost indestructable craft, you will have to spend about 15000 cr.

(Continued Next Page)

Software Reviews

So, you have your almost indestructable craft, what do you do with it. The manual does not tell you this, but the general idea is to shoot as many bad guys as possible and pick up the title of "Elite". You start off as Harmless (When you are trading for money for equipment) and you will gradually work up to Mostly Harmless followed by Average then Poor then Competent, if you are lucky you will make it to Dangerous then to Deadly. I have played this game now for months, and I still only rank as "Deadly", so if you are very patient you might just make "Elite". Now there is meant to also be 4 missions actually built into Elite. The first being to rescue some people from a Space station (The sun is about to Nova). This is the only mission that I have so far seen!

The PC version does not differ a vast amount from the Einstein one, the only real differences I have seen so far have been that the planets are filled and also the graphics are a lot smoother, however it is not really worth going and spending a small fortune on a PC for!

What I think.

Well it is a good little (Little Ed?) game, but you need a lot of patience which not many people have.

The game is auto loading and copyprotected.

Percentages:-

Loading	:100% Auto Loading.
Instructions	: 70% Quite an Interesting read.
Graphics	: 80% O.k Fairly smooth Quality.
Sound	: 5% Zap Zap Zap Bang etc.
Playability	: 1% Several months needed here.
Overall Value	: 50% Alright if you have time to waste.

Hints & Tips

General Computer Care

As this is going to be a very general type of Einstein magazine this month I have decided to do a little article for you on general computer care.

Any computer gets quite hot when you use it and when you switch it off condensation can occur, particularly if you are using the computer in a cold place. Obviously dampness in your computer can have a bad affect on some of the computer chips and the like. I would suggest that if you are using a computer in the kind of condition mentioned above then you should purchase a dust cover.

If you don't want to go to the expense of buying a dust cover (about £7.50) then you could use a packet of silica gel. That is the stuff that you get in the box when you buy the computer or any other piece of electrical equipment. What you do is open the computer up (after switching it off first!) and stick a small packet near the speaker (Where condensation is likely to form, but do not stick it on any electrical parts! Use a good quality PVC tape to stick it though, otherwise it might fall off!

During most of the year delivered goods can arrive quite well chilled and condensation will start forming when you bring them into the house. Immediate use of the disk (if it is a disk) could corrupt some of the data and it could quite badly damage you disk drive getting the read - write head damp. Let things acclimatise for an hour or two - read the instructions for once before using it. Ignoring this might cause you to damage your disk or even your disk drive.

As I have said before computers tend to get hot when used and as the computer gets hot it expands and as it gets cold it will shrink again. this problem is one of the biggest threats to computers. If your computer breaks down it is more likely to do with the EXPAND/SHRINK factor than it is to do with "OVERHEATING" or "BAD TREATMENT" it is a fact that MOISTURE and the EXPAND/SHRINK factor kill more computers than anything else. It looks like a silica gel packet is needed again or you will have to splash out on a dust cover.

Beginners Corner

This month, I decided to take a break from DOS and decided to look at XBAS and how to make simple shapes using the SHAPE and SPRITE commands.

Sprites are strange creatures which are very common in games, however when the average human (You or I) tries to create them they are harder than they actually look. The actual sprite is defined by a command called SHAPE. This command is used with a number e.g 144 and a "string" of numbers inside quotes. As this looks even more complex than it is I will try to make it as easy as possible. first draw a grid 8 * 8 on a page, or use graph paper.

1-	:	:	:	:	:	:	:	:
2-	:	:	:	:	:	:	:	:
3-	:	:	:	:	:	:	:	:
4-	:	:	:	:	:	:	:	:
5-	:	:	:	:	:	:	:	:
6-	:	:	:	:	:	:	:	:
7-	:	:	:	:	:	:	:	:
8-	:	:	:	:	:	:	:	:

	^	^	^	^	^	^	^	^
	1	2	3	4	5	6	7	8

Once this is done you can fill in the square with what you want so, if you wanted to design a key you would do it like this:-

1-	:	:	:	:
2-	:	.	.	.	:	:	:	:
3-	:	:	:	:
4-	:	.	.	:	:	:	:	:
5-	:	.	.	:	:	:	:	:
6-	:	.	.	.	:	:	:	:
7-	:	.	.	:	:	:	:	:
8-	:	.	.	.	:	:	:	:

	^	^	^	^	^	^	^	^
	1	2	3	4	5	6	7	8

This is a rough drawing of a key onto paper. The next thing you want to do is convert the picture to binary which is the easiest bit (no pun intended!!).

Beginners Corner

1-0	0	1	1	1	1	0	0
2-0	0	1	0	0	1	0	0
3-0	0	1	1	1	1	0	0
4-0	0	0	1	1	0	0	0
5-0	0	0	1	1	0	0	0
6-0	0	1	1	1	0	0	0
7-0	0	0	1	1	0	0	0
8-0	0	1	1	1	0	0	0

	^	^	^	^	^	^	^
	1	2	3	4	5	6	7 8

When this is completed you can see how it works. Each 1 represents a dot which you have added, and each 0 represents a space so, you get a block of 1's and 0's which make up the key.

```
00111100
00100100
00111100
00011000
00011000
00111000
00011000
00111000
```

Then comes the nasty bit. On page 225 of the Introduction to Einstein there is a table, and here is where you have to do some work. You now have to look for the following combinations on page 225, 00111100.

This number 00111100 infact equals 3C in the book, so going through the whole set you will end up with this.

```
00111100 = 3C
00100100 = 24
00111100 = 3C
00011000 = 18
00011000 = 18
00111000 = 38
00011000 = 18
00111000 = 38
```

You now need to put all these numbers together in the shape command so:-

```
10 SHAPE ***, "3C 24 3C 18 18 38 18 38"
```

(*** equals any number from 144 to 225)

Beginners Corner

When all these numbers are added together they tell the computer to define shape *** to be a key shape.

Now, we have created our key, it would be nice to put it on the screen and move it about a little.

SPRITE. This is the command we will use to do this, but to use the command we need to know a little about it before we start we need to know more about this command.

SPRITE P,X,Y,C,SN

The 5 parts are as follows. The letter P in this case stands for priority (0-30). The priority is very important and can be used to make some very interesting effects. The higher the number is the lower the priority given to it. It basically means that something with a high number e.g 20 will seem to pass behind something with a low number e.g 5. The letters X and Y stand for the coordinates. X is anything from 0 to 255 while Y is anything from 0 to 191 (X is across while Y is up). C stands for colour. This can be any number from 0 to 15. Finally the letters SN. This is the shape number. If you have defined shape 144 to be the key, to tell the computer to put it onto the screen you will need to tell it that number.

So, after doing all this you can define a sprite and put it on the screen, but can you move it?

Below I have listed a very basic program which moves a few keys around, see what you think!

```
10 CLS
20 MAG 0:REM ** MAGNIFICATION SIZES CAN BE 0 OR 1 **
30 SHAPE 144,"3C 24 3C 18 18 38 18 38"
40 FOR F=1 TO 191
45 L=RND(15)
50 SPRITE 1,10,F,1,144
60 SPRITE 2,190,F,2,144
70 SPRITE 3,F,10,3,144
80 SPRITE 4,F,190,4,144
90 SPRITE 0,F,F,L,144
100 REM ** PRESS SHIFT + BREAK TO STOP THE PROGRAM **
110 NEXT F
120 GOTO 40
```

This program moves a few keys around! Play around with it and have some fun!

Bargain Basement

Key *Shand* = Second hand software / Hardware. (Tried and tested 100% ok)

****NEW**** = New pieces of software / Hardware.

Einstein Hardware and Software

<u>Description</u>	<u>:Normal Price:</u>	<u>Offer Price.</u>	
Starbase	:£10.00	:£ 7.00	*SHAND*
Sprog	:£10.00	:£ 7.00	*SHAND*
Hyperball	:£10.00	:£ 7.00	**NEW**
Jump Mania	:£10.00	:£ 6.50	**NEW**
ZEXL	:£12.99	:£ 7.00	**NEW**
AEGOS (Addictive Einstein Games of Skill).	:£12.00	:£ 9.95	**NEW**
Tubes Mouse software. Version 6.6 (Including keyboard and Joystick controls.	:£15.99	:£ 8.00	**NEW**
Tubes Version 6.6 update	:£ 2.00	:£ 2.00	**NEW**
The Cracker	:£12.95	:£ 8.00	*SHAND*
Dbase II v 2.43	:Varies	:£ 30.00	**NEW**
Dos 80	:£19.95	:£ 15.00	**NEW**
System 5	:Varies	:£ 25.00	**NEW**
Screen Plus	:Varies	:£ 11.00	**NEW**
Super Forth 1.12 (Kuma)	:Varies	:£ 15.00	**NEW**
Hisoft Pascal	:Varies	:£ 18.00	**NEW**
Tasword + Tasprint	:Varies	:£ 20.00	**NEW**
Wordstar	:Varies	:£ 8.00	*SHAND*
Basic Tutorial	:Varies	:£ 7.00	*SHAND*
CP/M 2.2 + Amtat 1 & 2 (Read Amstrad discs on the Einstein).	:Varies	:£ 25.00	**NEW**

Hardware, Media and Books.

<u>Description</u>	<u>:Normal Price:</u>	<u>Offer Price.</u>	
FLEXIDOS Rom.	:£39.99	:£ 15.00	**NEW**
Tatung TM01 Colour monitor in good condition	:Varies	:£100.00	*Shand*
Phillips Computer Monitor 80. Green Screen Mono monitor	:£45.95	:£ 35.00	*Shand*

(Continued Next Page)

Bargain Basement

Sharp 3.5" Drive inc PSU and leads	:£137.95	:£ 80.00 *Shand*
Einstein Ram Drive 256k	:Varies	:£ 90.00 **NEW**
Video 1000 mono monitor Medium Res Mono monitor.	:Varies	:£ 30.00 *Shand*
Einstein Joysticks	:£ 9.95	:£ 8.95 **NEW**
3" Internal Drives	:Varies	:£ 25.00 **NEW**
BBC Bit Printer	:Varies	:£ 25.00 **NEW**
Oki Microline 132 Column F A S T printer, with ribbons.	:Varies	:£ 50.00 *Shand*
80 Column Cards	:Varies	:£ 60.00 *Shand*
Twin 5.25" Drives, with Case & Psu. (TEAC)	:Varies	:£150.00 **NEW**
Twin 3.5" Drives, cased with PSU. (Panasonic)	:Varies	:£150.00 **NEW**
Maxell 3" Discs (Each)	:£ 2.50	:£ 2.20 **NEW**
Maxell 3" Discs (10)	:£22.00	:£ 20.00 **NEW**
Disk Care 3.5" Cleaning Kit.	:£ 3.95	:£ 2.50 **NEW**
Disk Care 5.25" Cleaning Kit.	:£ 2.95	:£ 2.00 **NEW**
Mouse Pad. Hi Quality	:£ 4.95	:£ 2.95 **NEW**
3.5" Diskette Box (10) Capacity	:£ 1.50	:£ 1.25 **NEW**
3" Pollylabels (Erasable)	:£13.95	:£ 9.95 **NEW**
3.5" Pollylabels (Erasable)	:£13.95	:£ 9.95 **NEW**
5.25" Pollylabels (Erasable)	:£13.95	:£ 9.95 **NEW**
Printer Stand. Fits ANY printer.	:Varies	:£ 5.95 **NEW**
Commodore Amiga Mouse (Winner)	:Varies	:£ 19.95 **NEW**
Commodore 64 Mouse & Software (On 5.25" Disc)	:Varies	:£ 19.95 **NEW**

Bargain Basement

Postage, I'm afraid that I now have to add P + P to items mentioned above, and they are put into the following areas.

1: Monitors, Printers and Computers + £8 (Insured Delivery).

2: Disc Drives, 80 Column cards etc. £3 (Insured).

3: Small pieces of hardware i.e Ramdrive and also cans of refresh £1.50

4: Boxes of discs (10), Poly Labels, Printer Stands, £2.00 (Insured Delivery).

5: Single discs, games, cleaning kits, 3.5" Disc Boxes 10 Capacity, mouse mats. £1.00

Phone first to make sure I still have the item you want in stock! Leighton Buzzard (0525) 210868.

Please Make all cheques payable to R.Prime

The New Public Domain Catalogue

The Public Domain Catalogue has always been shunned by any respectable computer user, because it is so disorganised and some of the programs are so bad, they are not worth the disc space.

The Beds Computer Group has changed this, I have spent considerable lengths of time throwing the junk out, and keeping the good stuff, as well as adding some new. The Public Domain Catalogue is now organised into the following catagories:-

The "G" Catagory is for games and music related programs.

The "U" Catagory is for utilities.

The "C" Catagory is for Computer Aided Design.

The "W" Catagory is for Wordprocessors.

The "L" Catagory is for Languages.

The "I" Catagory is for Information and Help files.

The "M" Catagory is for manuals. (Printed)

The Public Domain Catalogue is now arranged in that order, i.e. if you were looking for the MBASIC language, you would look under the "L" catagory and follow the page down until you found the order code. The order code for MBASIC is PD L001, so if you wanted to order this disc, all you would have to is ask for PD L001, the same goes for all the other catagories, if you wanted some games writted in MBASIC, all you would have to do is look up the games section, and follow the page down until you found the order code for that, which in this case is G003. Simple see!

This is very easy when you are at a Computer Show to find what you want quickly, and hand over your disc and your money, however, when you are ordering from home you have 2 options.

The first option is the most used. Simply send a disc and some money. I require one disc for every 2 orders, say you wanted PD L001 and G003, all I would require would be one disc, however if you wanted more, I would require more discs.

The second option is one of my "Brain waves", all you have to do is give me a ring and tell me what you want,

(Continued Next Page)

The New Public Domain Catalogue

and sending me some money, when I recieve the cheque, I will send you the disc, cutting out all the time and bother, and that way you WILL NOT have to worry about losing any of your discs in the post!

I'm also doing a manual service. Basically this means you can buy a good quality printed manual for each catagory. There will be 6 manuals in total, and each one will give reviews of the Public Domain in the catagory and give a list of the programs.

The other service that will be simular to this will be the Instruction service. I will print out all the .DOC files in a manual form, this should make it easier for you to use the software, without scanning .DOC files for instructions.

Prices:-

Public Domain now costs £1.00 per order number.

Public Domain on a disc costs £4.50 (2 order numbers and postage and packing).

Manuals will cost 25p. Each one will cover a catagory, each of these will be put in Albert's Bi - Monthly Review.

Instruction sets cost 50p each and will cover a set of programs or one particular program on one of my Public Domain discs.

Please Note: The price for the manuals, instruction sets and Public Domain on a disc INCLUDE postage and packing. If you order normal Public Domain, please inclose 1 * 20p stamp per disc.

Please make all cheques for Public Domain and Public Domain on a disc payable to Mrs C. Prime, if ordering manuals or instruction sets, make your cheques payable to Mr. R. Prime.

The Public Domain Catalogue

Games / Music Section.

<u>Order No.:</u>	<u>Description</u>	<u>:Cols</u>	<u>:Size</u>
PD G001	:Games Written in Compiled Basic. These programs require System 5's XR.COM program to run.	:40	:164k
PD G002	:Want a senseless chat with your Einstein? This is the program to do it! Based on the original Eliza program from the '70's. Also includes a program to convert your ASCII or WP file into a load a waffle!	:40/80	:186k
PD G003	:Games written in Microsoft Basic (MBASIC), from the very primitive to nice graphical games, requires a fairly new version of MBASIC to run. (Will not run them all with the PD Version.)	:80	:122k
PD G004	:Into wargames? Then this is the disc for you! This is the wargames disc. Also included is a chess program. If you are good at remembering moves, you can win the game! (No Graphics!)	:40/80	:92 k
PD G005	:The famous BBC BASIC music and slides program, is there anyone that hasn't yet seen them? Requires BBC BASIC (Z80) to run.	:40	:140k
PD G006	:Lots of Basic Games (XBAS). Demos ,Adventures and some utilities, a see for yourself disc!	:40	:148k
PD G007	:The Colussus Cave adventure. Known better as the original adventure. Programmed by Level 9 (Part 1)	:40/80	:154k
PD G008	:The Colussus Cave adenture, both parts are needed to run the adventure. (Part 2)	:40/80	:40 k

(Continued next Page)

The Public Domain Catalogue

Games / Music section.

Order No.:	Description	:Cols	:Size
PD G010	:Find Crosswords difficult, solve them with this puzzle solver. 3-8 Letter word solver. Written in XBAS.	:40	:176k
PD G011	:Same as PD G010, except is for 9-19 letter words. These can be used seperatly, ideal for people that can't spell!	:40	:158k
PD G012	:6 machine code games! Othello, Bat Attack to name but 2!	:40	: 65k
PD G013	:A large selection of music for the DAC interface. Includes the MUSIC.COM program and the INSTALL.COM program.	:40	:176k
PD G014	:Another selection of music for the DAC interface.	:40	: 88k
PD G015	:2 Exellent games, firstly 3d Golf. This is a nice golf game, but is a little slow. Secondly Fire! A tank game!	:40	:164k

148

13 U

3 CAD

3 WPS

SL

G Int

The Public Domain Catalogue

Utility Section.

Order No.:Description:Cols:Size

PD U001	:Personal Database System written in XBAS. Includes a full 22k .DOC file manual.	:80	:76 k
PD U002	:Disc catalogue utility. Will let you catalogue all your discs, and keep a record of all your software titles.	:40/80	:36 k
PD U003	:Bradford Fonts. Will allow you to make very good prints from WORDSTAR, WP40, WP80 or VDO 25 on very cheap printers.	:40/80	:62 k
PD U004	:dTune. Will allow you to convert .TXT files from dBase II to ready to run .COM files. Includes a 10k document file.	:80	:92 k
PD U005	:Information management system. An easy to use Database. Requires MBASIC to run.	:80	:188k
✓ • ? PD U006	:General Disc Utilities, ranging from a disc verifier to a Directory file sorter.	:40/80	:154k
PD U007	:Assemblers! Make .COM files out of .HEX files. Help files are included on the disc.	:80	:128k
PD U008	:Lots more disc utilities, including Xtrax and Eindisk (Disc Modifiers), and ZDIS a Z80 disassembler.	:40/80	:72 k
? PD U009	:RBBS (Modem Communications program). With help files and installation programs.	:80	:146k
* PD U010	:EINSTRAD and AMSREAD. Will allow you to copy files from Amstrad CPC's, C464's and C128's. WARNING although it will copy games, unless you have Amstrad BASIC, you cannot run them!	:40/80	:128k

(Continued Next Page)

The Public Domain Catalogue

Utility Section.

<u>Order No.:</u>	<u>Description</u>	<u>:Cols</u>	<u>:Size</u>
PD U011	:Tubes Version 2.00. The early version of my mouse program. Written in XBAS. Requires the Taurus mouse to run.	:40	: 28k
PD U012	:Mail list program, keep all your Cristmas Card lists here! Mostly machine code, but with a couple of MBASIC programs on the disc.	:80	:164k
PD U013	:MS/DOS to CP/M transfer, copy PC files to your Einstein! (As with AMSRD, there is no garantee that the program will work when you copy it). Also Intel to Zilog converter (IBM to Z80).	:40/80:	:138k

The Public Domain Catalogue

Computer Aided Design section.

Order No.:Description:Cols :Size

✓ • ? ✕ PD C001 :The "GRAF" program. Is Basically
a plotter program, however you
can get stunning results. Works
on all EPSON compatible printers.
Comes complete with a 20k help
file and a printer demo to
help you. :40/80:188k

✕ PD C002 :Eplox Program, and no I don't
know why it is called Eplox!
Should be run on a double
head 3" or 5.25" or 3.5"
drive as this program needs a lot
of files (It covers 2 discs).
Not a good idea to buy this
unless you like swopping discs!
(Part 1). :40/80:180k

✕ PD C003 :Eplox Program files, yes more
files for the Eplox program,
still a nice package, this
disc has some of the files that
you will need to run Eplox.
(Part 2). :40/80:134k

The Public Domain Catalogue

Wordprocessing Section

Order No.:Description :Cols :Size

✓ ?	PD W001	:VDO 25. An advanced text editor. I would say it is one of the best Wordprocessing Packages I have ever seen! It's better than Kuma's Wordpro and as good as Wordstar. Why was it put on the Pd list?? Includes a 24k .DOC Manual.	:80	:110k
	PD W002	:Wordpro Program written in Pascal I do not own Pascal, but have been informed by a fellow user that does, that it is fast, but not very user friendly!	:80	: 96k
✓ ?	PD W003	:The very latest version of VDE (V 2.66). Includes .DOC files and INSTALL programs.	:80	: 78k

Public Domain Catalogue

Language Section.

<u>Order No.:</u>	<u>Description</u>	<u>:Cols</u>	<u>:Size</u>
PD L001	:The MBASIC disc. Includes Version 5.2 of Microsoft Basic. (1981 Version). Will run some of the PD MBASIC programs, such as the typing tutor.	:80	:120k
✓ PD L002	:A Cobol compiler with Demos.	:80	:150k
✓ PD L003	:"C" Libraries. Disc also contains Extended SUBMIT support for small "C" Compiler.	:80	:170k
PD L004	:Maths and Financial programs written in MBASIC. Will work with the PD version of MBASIC.	:80	:112k
PD L005	:All kinds of MBASIC games, from Chess to Baseball. Works best with a later version of MBASIC. (Later than 5.2)	:80	:154k

The Public Domain Catalogue

Information Section

	<u>Order No.:</u>	<u>Description</u>	<u>:Cols</u>	<u>:Size</u>
✓ ?	PD I001	:Lots of Info! A disc full of help for DOS and CP/M utilities.	:40/80:	166k
	PD I002	:Help files for MBASIC, SUPERCALC and WORDSTAR.	:40/80:	88k
✓ ?	PD I003	:Help files for CP/M, Pascal, C, MBASIC and a help feile for a HELP.COM program!	:40/80:	182k
	PD I004	:Knitting Program. This program will get you all knitting. Written in XBAS.	:40	: 58k
	PD I005	:Demo of Simplan by Surrey Software.	:40/80:	46k
✗	PD I006	:Demo of Grafdraw 2.00 also by surrey software.	:40	:184k

DBASEII Files and Programmes.

I intend this to be a sort of introduction to some of the concepts and techniques behind the use of a database manager, specifically dBase II from Ashton Tate. On the Einstein a second drive is essential as the programs require a whole disc in themselves. An 80 column card is also an advantage as this is the standard format. My copy of dBase for the Einstein doesn't work so I have had to use another CPM machine. The principles are, however, the same and I hope to build up a system of interlocking programs based on a membership theme.

Some basic ideas.

A database consists of a FILE of RECORDS, these records are made up of one or more FIELDS. This is the same principle in both manual and computer systems. A typical record in a manual system is an index card. It is lined or boxed and information written on it and it is held in a filing cabinet in a recognised sequence. The computer system duplicates this electronically with the database management system doing the job of the filing clerk.

Starting.

<CR> Is the usual abbreviation used for the ENTER key. The program is invoked!- A>DBASE<CR> and the user is prompted for the date which is entered in the form DD/MM/YY. The program then returns a dot, this is the command level state and in some versions is preceded by a screen of licence information. The first command we will enter is .SET DEFA TO B<CR>. This is because all datafiles and all but one of the command files will be held on drive B.

We will use the CREATE command to set up a skeleton file for our membership suite. When dBase is examining an entered command it looks at the first four letters of the word.

```
Enter .CREATE MEMSKEL<CR>      The response :-  
Enter record structure as follows:  
Field   Name, Type, Width, Decimal places  
001
```

Each field is given a unique name, it is defined as containing Character(C), Numeric(N) or Logical(L) information (Type) and the number of characters it can hold is declared. In addition, if the field is used for containing numerical data, the number of decimal places is defined. For our skeleton file enter the data below :-

```
001  SURNAME,C,20<CR>  
002  FSTNAMES,C,20<CR>  
003  ADD1,C,20<CR>  
004  ADD2,C,20<CR>  
005  TOWN,C,15<CR>  
006  COUNTY,C,15<CR>  
007  POSTCODE,C,8<CR>  
008  TITLE,C,4<CR>  
009  DATE1,C,8,0<CR>  
010  COMMENTS,C,25<CR>  
011  MEMNUM,N,3,0<CR>  
012  <CR>
```

The <CR> on an empty field terminates the creation of the file structure and the program prompts :-

012 <CR>
Input data now?

Answer N<CR> and you are returned to the dot prompt having created the basic structure of the membership file which we will use later. dBase can be used directly to input and retrieve information but we are going to be a little more adventurous. The first stage is to produce a master control file called MEMLIST.CMD. dBase has its own text editor which is invoked as follows :-

.MODIFY COMMAND MEMLIST<CR> as I said earlier dBase uses only the first four letters of the command so it can be stated as :-
.MODI COMM MEMLIST<CR>

The directory is scanned and as no MEMLIST.CMD file exists the response NEW FILE is returned and the cursor is positioned top left of the screen.

Lines to be typed in to the program will be preceeded by a carat sign (^) and I will try to explain as we go. dBase allows comments to be inserted as required. The * or command NOTE are the equivalent of the BASIC REM statement and I use them liberally as you will find out.

I like to put a header on to each program to help to keep track

```
^ * *****SAMPLE PROGRAM TO MAINTAIN MEMBERSHIP RECORDS*****  
^ * Steve Ryder      14/10/89  
  
^ * Set Up Environment and look for files  
  
^ SET CONFIRM ON  
^ SET FORMAT TO SCREEN  
^ SET TALK OFF
```

These set commands define the way dBase handles inputs and tells it to look at drive B for datafiles

```
^ erase  
Clear the screen
```

```
^ * get an idea of the current date NOTE no validation done here !!
```

Define the memory variable SYSDATE. This sets up block of memory with the label "SYSDATE" filled with 8 spaces. We then ask the user to input the date, telling them the format we require. The @/say function defines where on the screen the prompt will appear and the Getpicture clause defines the format of the input. The read command tells dBase to wait for the user to input information.

```
^ store "          " to SYSDATE  
^ @ 12,12 say "Please Enter Date (DDMMYY)" get SYSDATE picture "99,99,99"  
^ read  
^ set date to &SYSDATE
```

The & is known as a global operator. Here it tells dBase to set the system date to the value held in the memory variable SYSDATE

```
^ erase
```

Clear screen again

```
^ * set up files
^ store "      " to MEMLIST
^ @ 12,12 say "What is the name of the membership file?"
^ @ 13,12 say "Up to eight characters" get MEMLIST
^ read
```

To allow for various different clubs etc. we read the name of the membership file into a memory variable(memvar) to be used later with the FILE() operator to check for the presence of an existing file.

```
^ if file("&MEMLIST")
^   use &MEMLIST index &MEMLIST
```

If the file doesn't exist we have to create it. Here we use the skeleton file we created earlier.

```
^ else
^   erase
^   @ 12,12 say "Creating File B:&MEMLIST"
^   use MEMSKEL
^   copy stru to &MEMLIST
^   use &MEMLIST
^   index on SURNAME + FSTNAMES to &MEMLIST
^ endif
```

We now have a file in use with its associated index (about which - more later).

```
^ store T to RETURN
^ do while RETURN
```

Here we ensure that when we return to this program we set up the options again

Let's set up a menu screen using the TEXT/ENDTEXT function. This simply echos to the screen exactly what you type. Also we start a conditional loop to ensure that we only get the responses we require.

```
^ * Begin input and evaluation loop
^ store T to ILOOP
^ do while ILOOP
```

Do while ILOOP. We set up a logical mvar called ILOOP with a value of T (true). The instructions between the do while and the enddo will be repeated until ILOOP has a value of F (false). This only occurs below if the response from the user corresponds to one for which we have a valid action.

```
^   erase
^ * Set up screen menu
^ TEXT
^
^           MEMBERSHIP LIST
^           =====
^
^           A. Enter Membership Details
^
```

```

^      B. Amend A Members Details
^
^      C. Print Out A Membership List
^
^      D. Print Out Mailing List
^
^      E. Quit
^
^

```

```

^ ENDTEXT

```

```

^ * get an idea of the users requirements
^   accept "          Enter Option Required" to MENOPT

```

As an alternative to the @/say function I have used accept. This does not need a memvar pre-defining, if one does not exist it is created.

```

^ * convert to upper case to ease evaluation of response
^   store !(MENOPT) to MENOPT

^       if MENOPT $"ABCDE"
^           store F to ILOOP
^       endif

```

This is where the do while is terminated. The \$ (should be a dollar sign but my machine produces a pound sign) is the substring function. If the response string (which the previous ! function converted to uppercase) corresponds to any part of the string in parentheses the ILOOP memvar is changed to false and the program moves past the enddo and carries on.

```

^ enddo
^ * End of the input/evaluate loop

```

```

^ * What next??

```

Do case instructions allow conditional branching. We have obtained a response to the input menu which is one of 5 options. Here we give control to the sub-program which will carry out the required operation.

```

^ do case

^   case MENOPT = "A"
^       do INPUT

^   case MENOPT = "B"
^       do MODIFY

^   case MENOPT = "C"
^       do PRINLIST

^   case MENOPT = "D"
^       do ADDRESS

^   case MENOPT = "E"
^       erase
^       clear
^       quit

^ endcase

```

enddo

```
^ * End of main menu program MEMLIST
CTRL W to end the edit.
```

This is the end of the master control program, however if we enter dBase directly instructing it to go straight to the program A>DBASE MEMLIST<CR> we fall foul of the dBase default which is A:. The best way out of this is a small command file on drive A: as follows:-

```
.MODI COMM A:START.CMD<CR>
```

```
^ *Program to set defaults START.CMD
```

```
^ set defa to B
```

```
^ do MEMLIST
```

```
^ * End of START.CMD
```

```
CTRL W to end edit
```

Now if we type A>DBASE START<CR> the default is set and chains into our menu program.

This is where the first part ends. If you are still with me and have typed in the program any response A,B,C OR D will return to the command level dot prompt "Do Cancelled" as the sub-programs don't yet exist. We will look at them next time. The blank lines are not absolutely necessary but they make things more readable as does the staggering of if/endif and do/enddo routines. This is more noticeable with complex nested sub-routines.

Steve Ryder

0:

DBASE II Files and Programs.Part 2

In last months article we produced a datafile to use for keeping records of club members. As a first step to creating a usable system, the program MEMLIST.CMD was written with what were (hopefully) explanatory notes interspersed between the program lines. This month I am going to extend the process to include the sub-programs to input a list of members.

If the dBase editor is invoked MODI COMM MEMLIST it defaults to a filename MEMLIST.CMD. It will however edit any ASCII text file if the correct extension is used. We are going to produce a screen layout file called INPUT.DIS. Invoke the editor
.SET DEFA TO B:<CR>
.MODI COMM INPUT.DIS<CR>

Type in the following program, I have used - to indicate spaces as these are important, again the caret ^ sign distinguishes between program lines and comment.

```
^ * Display/read membership records  INPUT.DIS
^ IF FLAG=1
^ TEXT
^ -----MEMBERSHIP RECORDS
^
^
^ -----DELETE SURNAME TO EXIT (SPACES OR CTL Y)
^
^
^ -----SURNAME-----TITLE
^
^ -----FIRSTNAMES
^
^ -ADDRESS LINE 1
^
^ -ADDRESS LINE 2
^
^ -----TOWN-----COUNTY-----POSTCODE
^
^ -----COMMENTS
^
^ -----DATE-----MEMBERSHIP NUMBER
^
^ endtext
^ return
^ endif

^ if FLAG = 2
^   @ 7,18 get MSURNAME picture "!!!!!!!!!!!!!!!!!!!!!!"
^   @ 9,18 get MFSTNAMES picture "!!!!!!!!!!!!!!!!!!!!!!"
^   return
^ endif
^ if FLAG = 3
^   @ 7,48 get MTITLE picture "!XXX"
^   @ 11,18 get MADD1 picture "!XXXXXXXXXXXXXXXXXXXX"
^   @ 13,18 get MADD2 picture "!XXXXXXXXXXXXXXXXXXXX"
^   @ 15,18 get MTOWN picture "!!!!!!!!!!!!!!!!!!!!!!"
^   @ 15,42 get MCOUNTY picture "!XXXXXXXXXXXXXXXXXXXX"
^   @ 15,68 get MPOSTCODE picture "!!!!!!!!!!!!!!"
^   @ 17,18 get MCOMMENTS picture "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"
```



```

^ @ 19,18 get MDATE1 picture "99999999"
^ @ 19,68 get MEMNUM picture "999"

^ return
^ endif
^ return

```

```

^ * End of INPUT.DIS

```

CTRL W to end edit.

This is in effect a sub routine which we can call at any time and is to be used for both input and amendment of information. As can be seen, the precise function depends on the state of a memvar called FLAG. This and the memvars MXXXXX are set up in the program which calls the routine; the picture function defines the type of input allowed. ! converts lower case to upper, X allows any input, 9 allows only numeric, punctuation, +- and spaces only. This means the user can only enter the key fields in upper case thus simplifying the finding of records.

The input program is called INPUT.COMD . Invoke the editor as usual - MODI COMM INPUT<CR>

```

^ Program to input membership records INPUT.PRG
^ store str(1,30) to MEMSPACE

```

This line will be used later for initialising mvars

We need to set up a routine for the input of standard data, here we use a small routine with 3 passes. The result of the pass depends on the state of the memvar FLAG

Pass 1 sets up blank input screen.

```

^ set index to
^ go bottom
^ store £ to MEMNUM
^ set index to &MEMLIST

```

Find how many records we have in the file, we release the index as this won't necessarily show the last physical record. It would show the last alphabetical record which could be the first record in the file. Having established the number of the last record, we store it to a mvar which is used later and reestablish the index.

```

^ store T to REPEAT
^ do while REPEAT

```

Set up a loop to allow input of records until the escape conditions are found.

```

^ ERASE
^ store 1 to FLAG
^ do INPUT.DIS

```

Here we have initialised the memvar FLAG and called the display screen, after writing headings to the screen, control is passed back to this program.

dBase is very good at manipulating memvars, so here we set up a series of them to match the fields in the datafile. To distinguish them the names are prefixed with M

We have set up mvars before using store " " to mvar

statements as in the first line below. This is clumsy and prone to errors especially when we are counting twenty spaces (in fact I miscounted myself when I was typing in this example). In line 1 of this program we set up a 30 character mvar called MEMSPACE with the digit 1 prefixed by 29 spaces. If we use the substring operator \$ we can chop as many spaces out as we need as shown in the other lines.

```

^ store " "to MSURNAME
^ store $(MEMSPACE,1,20) to MFSTNAMES
^ store $(MEMSPACE,1,20) to MADD1
^ store $(MEMSPACE,1,20) to MADD2
^ store $(MEMSPACE,1,15) to MTOWN
^ store $(MEMSPACE,1,15) to MCOUNTY
^ store $(MEMSPACE,1,8) to MPOSTCODE
^ store "Mr. " to MTITLE
^ store SYSDATE to MDATE1
^ store $(MEMSPACE,1,25) to MCOMMENTS
^ store MMEMNUM + 1 to MMEMNUM

```

Note that MEMNUM is a numerical variable so store numeric data. We checked for last record in file as we started, add 1 & this is next membership number. The store mvar + 1 on each pass through gives incrementation automatically.

Pass 2 gets the key information, the FLAG is 2 so the initial screen set up IF/ENDIF is ignored and the program allows the user to enter the key fields SURNAME AND FSTNAMES into screen areas, reads the responses into the mvars MSURNAME and MFSTNAMES (converting to uppercase) and returns control to the calling program.

```

^ store 2 to FLAG
^ do INPUT.DIS
^ read

```

exit routine. This vets for a blank input to the MSURNAME mvar. Note, if the user right-justifies the field, the routine treats that as a blank entry.

```

^ @ 20,0
^ if $(MSURNAME,1,2) = " "
^   erase
^   store F to REPEAT
^   return
^ endif

```

Vet for existing record. dBase uses index files .NDX. These allow very rapid location of specified records. The MEMLIST program took a file name supplied by the user and used the MEMSKEL file to set up the membership file. It then instructed dBase to create an index file using the SURNAME and FSTNAMES fields as a multiple key. Here, the two mvars MSURNAME and MFSTNAMES are added together and written to an mvar called KEY. The global operator is used to check for the presence of an existing record. If the find is successful, a message is flashed on the screen until the program counts from 1 to 100. The user is then returned to the start of the SURNAME area on the screen.

```

^ store T to EXIST
^ do while EXIST
^   store MSURNAME + MFSTNAMES to KEY
^   go top

```

```

^ find &KEY
^ store £ to RECNO
^ if RECNO <> 0
^   @ 20,1 say "THIS IS AN EXISTING RECORD PLEASE USE OPTION B "
^   store 2 to FLAG
^   store 1 to K
^   do while K <100
^     store K+1 to K
^   enddo
^   @ 20,0
^   do INPUT.DIS
^   read
^ else
^   store F to EXIST
^ endif
^ enddo

```

At this point the program has determined that no record exists for this name so the display routine is accessed for the last time. FLAG is set to 3 and skips the first 2 IF/ENDIF routines and allows the input of the rest of the information. The membership number is determined by the record number in the file. This means that old membership numbers remain unused unless input manually.

```

^ store 3 to FLAG
^ do INPUT.DIS
^ read noupdate
^ clear gets

```

Next we ask the user what they want to do next, offering a choice of Keep the record or Quit this input and start again. The routine loops until one choice or the other is input.

```

^ store T to KEYLOOP
^ do while KEYLOOP
^ store "K" to CDATA
^ @ 21,1 say "OK? "
^ @ 22,15 say "K = keep, Q = quit record, "
^ @ 21,10 get CDATA picture "!"
^ read noupdate
^ clear get
^ if CDATA = "K" .or. CDATA= "Q"
^   store F to KEYLOOP
^ else
^   @ 20,1 say "Invalid input"
^ endif
^ enddo

```

We now have a decision this sub routine Keeps the information input. We have all the information stored in memvars of the same type and size as the fields in the datafile. The program has looked through the file, made sure that no record exists and effectively sits at the bottom of the file. The APPEND BLANK instruction does just that. It adds an record with all its fields blank and the REPLACE lines fill in the information from the memvars into the record fields.

```

^ do case
^   case CDATA = "K"
^     append blank
^     replace SURNAME with MSURNAME

```

```

^ replace FSTNAMES with MFSTNAMES
^ replace ADD1 with MADD1 noupdate
^ replace ADD2 with MADD2 noupdate
^ replace TOWN with MTOWN noupdate
^ replace COUNTY with MCOUNTY noupdate
^ replace POSTCODE with MPOSTCODE noupdate
^ replace TITLE with MTITLE noupdate
^ replace DATE1 with MDATE1 noupdate
^ replace COMMENTS with MCOMMENTS noupdate
^ replace MEMNUM with MMEMNUM noupdate

```

Here the Quit option is taken, a message is displayed for a while and the program loops back to the start of the input procedure.

```

^ case CDATA = "Q"
^   @ 20,1
^   @ 20,1 say "OK.I'M LEAVING THIS RECORD UNALTERED"
^   store 0 to K
^   do while K < 100
^     store K + 1 to K
^   enddo
^   @ 20,1
^   go top
^ endcase
^ enddo
^ return
^ * End of INPUT.CMD

```

CTRL W to end the edit.

Here we now have the ability to enter information to our file. The program checks to prevent duplicate records in so far as a simple program like this can do. It can't of course prevent RYDER STEPHEN being entered as well as RYDER STEPHEN RONALD. The onus of checking for duplication like this is on the user.

This is a good point to pause- partly because I have typists cramp in my typing finger. The next part will build on our foundation and allow us to examine and amend information.

Steve Ryder

0:

Backpage info

Alberts Bi - Monthly Review

Back issues are available from June 1988 (Issue 01/01) till June 1989 (Issue 02/01). These cost 50p each or 77p including Postage and packing.

Please note. The source code is now available from the PD library.

Correspondence regarding the group except Public Domain software should be sent to:-

Mr. R. Prime
The Beds & Bucks Computer Group,
17 Bury Rise,
Tilsworth,
Leighton Buzzard
Beds. LU7 9PR

All cheques for software, manuals, books etc from the bargain software page and auction page (Unless stated otherwise) should be made payable to R.Prime.

Correspondence regarding Public Domain software to:-

Mrs. C. Prime (Same address)

Please note. You can now send off for Public Domain software on the following format of discs.

5.25" Single Sided 40 track.
5.25" Double Sided 40 track.
5.25" Double Sided 80 track.
3.5" Double Sided 80 track.
3 " Single Sided 40 track.

Make all cheques for Public Domain software payable to Mrs. C. Prime.

Pd. Prices. 50p per Number (One 3" disc side).

How to submit Articles.

You may submit articles on 3" disc using the following Wordprocessor formats:-

Wordstar Tasword WP80

If you have any queries regarding any of the above you may telephone 6pm and 9pm Monday to Friday and 5pm and 10pm weekends.

Roy Prime. 0525 210868.

Credits

I would like to thank my mum for her on the spot editing. Monday morning at 8.00 am!

Please Note. All articles in this magazine are copyright (c) 1990 Beds & Bucks Computer User Group. If any other group would like to use an article, all you have to do is ask!

Help-line

If you have any queries on the magazine, or software or the Einstein in general, here is a list of people that you can now contact.

Adventure Games, Pascal, Musical Applications.
Josef Karthauser, 17 Manor Close, Abbots Ann, Andover
Hants, SP1 7BJ.

Tas-Sign, Tasword and XBAS,
Roy Prime (See Backpage).

Computer evaluation.
David Thompson (0482) 412542.