MICRO-SCOPE 6

JUNE 1982

Newman College with MAPE

Contents

Editor John Lane
Board John Fair, Alan James, Roger Keeling
Design David Barlow

ISBN 0 602 22613 9
Sponsored by the Department of Industry
Correspondence to the Editor: Newman College, Bartley Green, Birmingham, B32 3NT
tel: 021 476 1181

MICRO-SCOPE is mailed directly to MAPE members

Published by Heinemann Computers in Education Ltd
in partnership with Ginn and Company Ltd
Individual copies from Heinemann/Ginn at:
Prebendal House, Parson's Fee, Aylesbury, Bucks, HP20 2QZ.

Typeset by Castlefield Press, Northampton.
Printed by Woolnough Bookbinding, Wellingborough.

MICRO-SCOPE 6
June 1982

Editorial

At the Exeter Conference a Scottish teacher was responding positively to an enthusiastic informal presentation of the PROLOG language. Suddenly she stopped short. 'How on earth,' she asked, 'can I go back and sell them this idea, when I've been hammering BASIC at them all year?'

Last year's generalised euphoria was replaced by detailed work in progress. The pace and variety of advances are exciting. But they put a strain on the lines of supply and communication. There were fewer absolute beginners at the Conference this year, and they were more easily overawed by the rapid expertise around them. The gap will widen alarmingly.

MICRO-SCOPE has an important dual role here. On one hand, we hope to maintain lively contacts with the shifting frontiers of new development, and to create links between diversifying activities. On the other, we are aware that the majority of our readers are less involved, and seek general information and guidance rather than technical detail. New readers will have important reservations and pertinent questions. We intend to keep these lines open too.

Papert's Turtle (see *MICRO-SCOPE 5*) made a big impact. We are fortunate to have in this issue a report on its use from Beryl Maxwell, who gave one of the most impressive and popular demonstrations. We also have a further note on BIGTRAK.

Our last edition contained a cryptic anticipation ('Beyond BASIC') of an issue which came live at Conference. LOGO and PROLOG may have generative power and dynamic logic when compared with the static atomism of BASIC. Even more significantly, their structures parallel many features of ordinary language which young children acquire so readily and creatively. The implications of a programming language *designed* for children to use are startling. Richard Ennals opens up this area for us with a provocative article (p. 9), and we intend to follow this theme energetically.

Commercial interests – manufacturers, publishers, software houses – were heavily represented at the Conference. Primary teachers in earlier curriculum developments like Nuffield or Bullock have not had to cope with such big guns. We wish to promote positive links, and would like to hear of examples of good practice in co-operation and commercial support for educational projects. We will also take up reports of any clash of interest.

This is our Information Retrieval issue. Dr Ross (p.10) illustrates an approach to local history involving the storing and collating of original material. We back this up by featuring data banks in the program listing (p. 16) and the BASIC article (p. 20).

There's posh, now! We hope readers like our new presentation, produced by Heinemann/Ginn. Contributors, please note the facility for photographs and illustrations. The Business Meeting at Exeter confirmed the association of *MICRO-SCOPE* with MAPE. We intend to publish an official conference report in the late summer, collated by Roy Garland and Mike Thorne.

We are pleased to acknowledge gratefully an extension of the generous support we receive from the Department of Industry. This will help us to establish national distribution of *MICRO-SCOPE* and to launch one-subject '*MICRO-SCOPE* Specials'. We are considering a 'Special' on programming languages, and one designed for children and parents. Please send ideas and material soon.

October 1 is the Editor's *deadline* for copy for *MICRO-SCOPE 7*.

Exeter Conference 1982

This review and summary of impressions was written for *MICRO-SCOPE* by Elizabeth Moore, of the Early Education Research Group, Faculty of Educational Studies, Open University, Milton Keynes. We hope to include in our next issue an account of her important project on applications of the micro to work with younger children.

There were three keynote lectures at this year's Easter Conference, held at Exeter.

Daniel Chandler

The first, sponsored by B.P., was given by Daniel Chandler, who began by reminding us of Marshall McLuhan's adage 'The Medium is the Message'. In this instance the micro is the medium and the message is at the frontiers of imagination.

As a full-time consultant on the Schools Council 'Micros and English' project, Daniel Chandler is interested in exploiting the micro as a writing tool for children to use. It is often suggested that the young child with writing difficulties should use a typewriter, as with the severely physically handicapped and their 'Possums' and sons of Possum. However, young writers have difficulty in operating word-processing packages. The enterprising micro-computing entrepreneur might like to tackle this problem.

This first lecture also discussed fact-files, with illustrations from Animal-type branching programs. As Isaac Asimov wrote last year, 'the sum total of human knowledge lacks an index'. If children are to be educated for information literacy they need the tools to build their own information environments: the microcomputer is one such tool. An advert for software was given at this point for 'Fact File', which will be available from C.U.P. later this year. The interested primary school teacher might also like to investigate the database work that can be done by middle school children using the PROLOG language.*

Daniel Chandler went on to consider how children can program adventure games for themselves and incorporate their own fantasies. He concluded on a note in keeping with Asimov, that we are merely robots if uncritically involved with our own technology. So a micro can only be justified in school if it does not conflict with two laws:

1. the computer should not be used to instruct a child to accept learning as a passive actitity;
2. the computer must accept the responses given by children because they are never wrong.

Richard Fothergill

Richard Fothergill in his capacity as Director of MEP presented a wide-ranging appraisal of primary school use of micros. He could not yet divulge DES policy on aid to primary schools wishing to purchase micro systems.

The topics he covered included parents as they grasp the nettle of the micro revolution. In this age of developing co-operation more might have been said of the partnership and sharing of information and skills between parents and teachers. There was mention also of micros for databases and micros as control systems. On the question of keyboard literacy Richard Fothergill suggested that, by the age of six, children would be able to find their way around the keyboard as well as reading the TV screens of text and pictures. If there is enough support, alternatives to QWERTY keyboards, such as Microwriters and derivatives, could come into use.

The question of helping children to cope with change was mentioned. It was suggested that computers will bring curriculum change. If this is to happen there needs to be further in-service training of teachers – not just in specific skills of operation but in achieving a balanced understanding and knowledge of the role of the computer.

David Burghes

The last lecture of the weekend was by Professor David Burghes of Exeter University. He asked what role micros should play in primary education? Maths and spelling games on the micro are fun for the children; the micro gives instant feedback and has infinite patience. Yes, the micro can be used for reinforcing and testing material that the teacher has taught. However, wasn't that the role of the ill-fated teaching machines of the 1960s?

It is logical from David Burghes' view of what micros are doing to emphasise that they may not be important. The Exeter audience listened as he said that micros may just be a novelty and they won't necessarily last. For *MICRO-SCOPE* readers who want to follow up these thoughts see: 'Microcomputers, new technology not everyone's cup of tea' in *Education*, 26 March 1982; and the latest issue of *Perspectives* called

*See article on p.9

'Microcomputers in the Classroom' from Exeter's School of Education, St Luke's, Exeter, EX1 2LU.

Presentation on the Newman College project

Evaluations of primary schools' use of micros have not yet been much in evidence, although the first MEP/MAPE case studies are now available. One of the presenters at the Exeter Conference was Roger Keeling, who gave a description of the Newman College project (discussed in *MICRO-SCOPE 4* and *5*: see also progress report on p.30). The project is documenting the use of 480Z machines by six primary schools. Lots of useful practical tips are emerging (e.g. do not pick a 22 inch monitor if your school is not on one level and does not have practising weight lifters on the staff). Over the course of the next year much more is likely to emerge whilst the use of the micros is being evaluated. Will the novelty wear off? Will the style of support offered by Newman College contribute to in-service and pre-service training in other parts of the country? After just three months' experience of having a micro in school, teachers are specifying the programs that they would like written for them. Patterns of good software documentation are being set up as part of this open-ended and adventurous project.

Impressions and afterthoughts

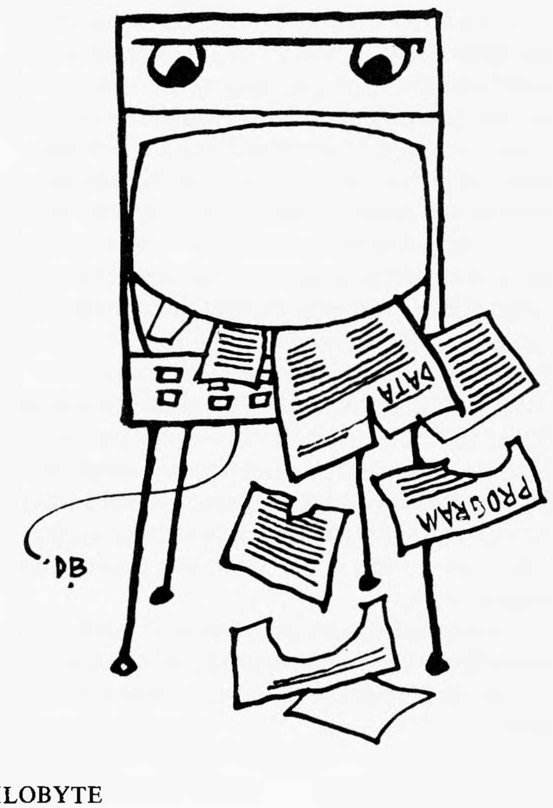
Entertaining articles about the Easter Conference (such as John Pearce's in *Education*, 2 April 1982) made us laugh. The hardened conference goer knows the unspoken rules that participants must at some point: 1. get lost; 2. find that vital presentations clash; 3. queue up. Within 24 hours of arrival the undaunted were talking avidly about next year: the changes and what must stay the same (nobody mentioned never going to a MAPE conference again although the daunted were suffering recursions of 1, 2, and 3). The flavour of this year's MAPE conference was vanilla: masses of individual ideas and people with a maze of strong interests. So there was much milling around and a level of critical discernment was about was was on offer. From the long-term committed MAPE idealists to the new initiates there was questioning as to what micros are actually doing and what they might be able to do for us in primary schools.

The publishers were well in attendance and teachers linked by contracts obviously could not divulge quite what they were doing. Great software is in store, apparently, but meanwhile we must all invent the wheel (possibly painting it the wrong colour). 'Pipeline Scintillating Software Production PLC' (pseudonym), for example, sounded as if they had ideas in parallel with the teachers they met. At least they kept saying they were doing something along the same lines for release in 1983. What great computer brains there must be around to be all thinking of the same things.

The Daniel Chandler lecture was in stirring style and crammed with an interesting set of personal preferences for sci-fi literature. Richard Fothergill, like all of the MEP directorate, delivered a very glossy presentation. Hopefully the content of these lectures and the David Burghes one will be published in due course and details advertised in *MICRO-SCOPE*. From the attendance and interest at this year's conference the next volume of papers might be subtitled: 'part 2 of the continuing saga – where is all this super software? Or, how to eliminate drill and practice on the primary school micro'.

Perhaps such large-scale conferences (over 200 participants) automatically mean one can only discover the educational concerns of a small number. More information about the style and format of interest groups might help participants to make their selections next time. Also, might national interest groups (relating to particular age ranges, subjects or themes) be set up with occasional day conferences?

The planning for this year's conference was no mean feat and the final production was very smooth and deserving of applause. The 14 regional MAPE areas will probably have their own gatherings before Easter 1983. It will be interesting to see what ideas about the next national conference spring up before then. Will a larger venue be sought? And what will the predominant flavour be?



KILOBYTE

A turtle in school!

Beryl Maxwell

A fuller version of Beryl Maxwell's report, 'LOGO at Crabtree School', is available from: The Advisory Unit for Computer Based Education, Endymion Road, Hatfield, Herts, AL10 8AU. Price 50p.

In April 1981, while attending a four-day course in Bedford, 'Computer Aided Learning' for primary/middle schools, I met the 'floor turtle'.

The floor turtle is a dome-shaped robot on wheels. It is motorised, and can be controlled by instructions typed on to the keyboard of a microcomputer. It carries a pen that can be lowered to the floor, so it draws pictures as it moves over paper.

The course started with an introduction to the RML 380Z machine, which Hertfordshire has adopted as standard for secondary schools. We had the opportunity to look at some of the available software for the primary age range. We also looked at other teaching aids, including BIGTRAK.

Then we were introduced to LOGO and the floor turtle. Suddenly I felt very excited! Children had shown me ZX80 machines, and there seemed to be a growing trend to let children program computers using BASIC. This did not seem to fit in with my own work and thinking over the last few years. Since 1971 I had been teaching mathematics each morning in a junior school in Hertfordshire. I had always aimed at getting children to understand what they were doing, believing mathematics needs firm foundations to be built on successfully. Teaching the 'tricks' of BASIC and trying to justify A = A+1, when I spend months trying to reinforce the concept of balance, did not seem to fit into this scene. On first sight LOGO seemed a much more appropriate way to start with young children.

We were given the opportunity to play with the floor turtle. Another course member and I decided to draw a stick man running upstairs. Before we knew where we were, we were busy observing the angles at the back of our knees and our feet on the steps. Here was a new slant on degree work!

At the end of the course, I volunteered to take the floor turtle into school, so the children could use it. So for seven days in June we 'turtled'.

I had been struck by the possibilities of allowing children to do their own things. After all, here was I, a teacher with virtually no turtle experience, wondering what they would do with it. So the children would have the opportunity to initiate their own learning in any direction they decided to move. How often had they ever found themselves in that situation before, I wondered.

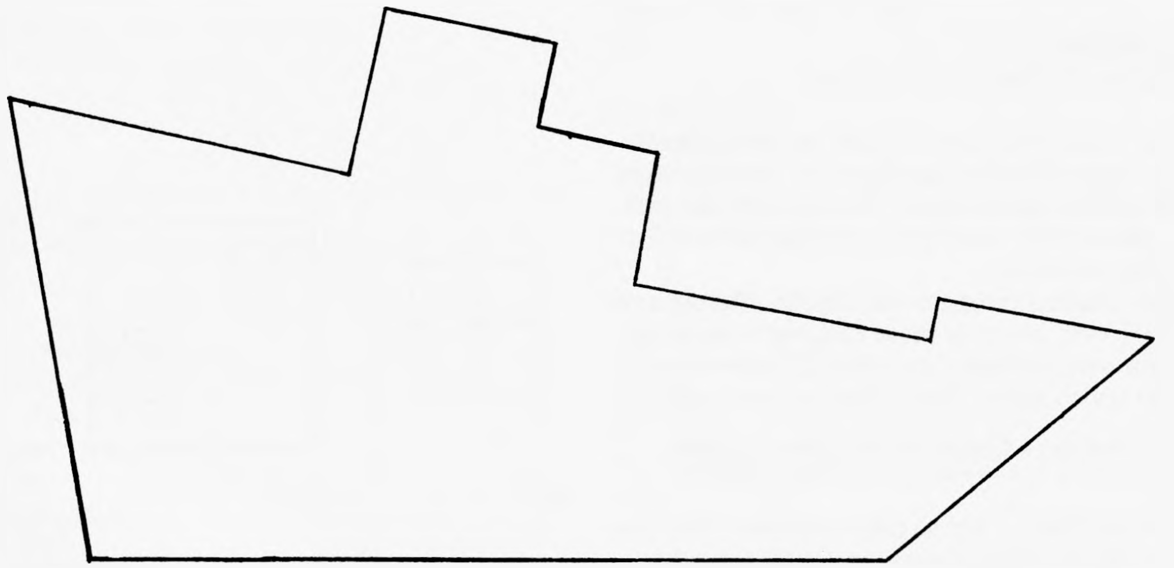
This was the first time a computer had appeared in school. We began with a couple of large group sessions, where I showed them the commands the turtle would obey, and how to use the keyboard. The turtle obeys instructions like these:

LIFT This lifts the pen up so it won't draw
PEN This puts the pen down so it will draw
FORWARD 10 This moves the turtle forwards in its current direction by 10 units – any number can be specified
BACKWARD 10 Moves backwards 10 units, leaving the turtle's 'heading' unchanged
RIGHT 90 Turns 90 degrees to the right
LEFT 50 Turns 50 degrees to the left
HOOT The turtle plays a little tune!

With these first few commands the children were soon able to instruct the turtle to draw.
Here is the first list of instructions by two fourth-year boys:

```
FORWARD 38
LEFT 40
FORWARD 16
HOOT
RIGHT 230
FORWARD 10
LEFT 90
HOOT
FORWARD 2
RIGHT 90
FORWARD 14
RIGHT 90
FORWARD 6
LEFT 90
FORWARD 6
RIGHT 90
FORWARD 4
LEFT 90
FORWARD 8
LEFT 90
HOOT
FORWARD 8
RIGHT 90
FORWARD 16
LEFT 111
FORWARD 22
HOOT
```

This gave the boys the boat shown below.



The boys wrote: 'Where our boat went wrong:
The instruction that made our boat go wrong
was the 5th instruction which was RIGHT 230.
they went wrong because the angle wasn't enough.'

Once the fault was found a new drawing with the corrected angle (RIGHT 220) produced their boat as originally intended.

The fourth-year children (a third of the year group, and the most able mathematically) worked in twos and threes. They evolved a system of sketching what they wanted the turtle to draw, and writing a list of turtle instructions. They soon became very aware of the necessity for accuracy in their measurements for both lengths and turning. Soon graph paper, protractors, rulers and scale drawings appeared at the planning stage. When they thought a list of instructions would produce their picture, it was tested by another group and often mistakes were found before they were keyed in to the microcomputer.

In no time the children became very ambitious. Elephants, aeroplanes, three-dimensional shapes, letters, words and spirals were being drawn by the turtle.

We devised a rota system to utilise all the possible keyboard time available, which included break and lunch times. The children found the floor turtle novel, challenging and absorbing.

The children were setting themselves problems. 'How do we get the semi-circular shape for our car wheels?' 'How can we draw a pattern on the turtle's shell?' 'How do we make a flat circle?'

Here were *real* problems. That gave the children the motivation to solve them. Thus the children were involved totally, taking the initiative and directing their own thoughts. They formulated ideas, tested them, corrected errors, reformulated ideas, retested them and eventually devised their own system of analysis and developed their own strategies. Mistakes led to learning and sometimes several routes were found to a solution.

'Turtle time' gave opportunities for cooperation, sharing ideas, putting forward arguments for this or that idea, and decision making. Mistakes were analysed and logical thinking was needed. Forward thinking was required and long periods of persistent concentration needed.

At the end of my seven sessions I was convinced that I had found a very interesting way of using a microcomputer in a primary school. Subsequently I was asked to have the floor turtle in school for the spring term of 1982 – but that will be another chapter!

LOGO in the primary school

Tony Mullan
Plympton St Maurice C.P. School

This is a short account of some work that took place with a small group of girls in the last three weeks of the spring term. The children had had experience with computers, but had never tried to program before.

The finished program may not be elegant or a mind-blower, but it is these children's response to their own problem, and that is the best problem to try to solve. This is their account of:

Teaching a Computer to Draw a Street
****

First we had to try to draw a house. Then we had to think of the shapes we would need to draw the house, these were a triangle (Fig. 1) and a square (Fig. 2):

Fig. 1

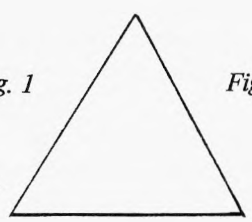
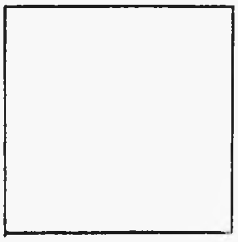


Fig. 2



When we had thought up the shapes we programmed the computer to join the two shapes together. We had a bug in the program to join the shapes together. This was our bug (Fig. 3). The solution was to move the turtle and turn it before joining on the triangle (Fig. 4).

Fig. 3

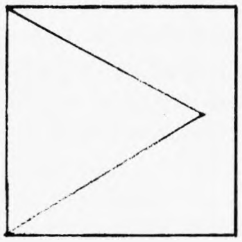
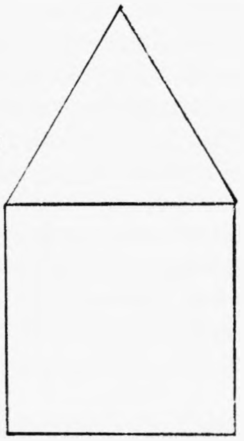


Fig. 4



We also had a procedure called FOURSQUARE which we could use for the windows. It looked like this (Fig. 5).

The first time we tried to put in the windows this is what happened (Fig. 6):

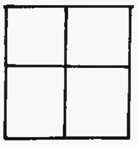


Fig. 5

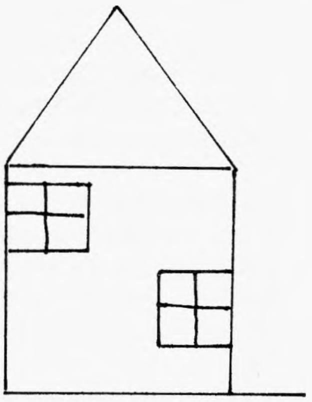


Fig. 6

We had a lot of problems in putting in the windows, but we solved them after a while. So the house would be right for all different houses we had to make all distances change with the size of the house. To solve the distance between the bottom of the windows and the bottom of the house we had to take SIDE/5 and SIDE/4 and add them together, then take that from SIDE. Then we could put in the procedure called RECTANGLE to make the door. We also had to find the distance between the side of the window and 1/8th from the middle so the door would be in right place.

Our finished house was as in Fig. 7.

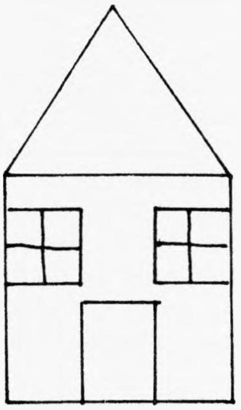
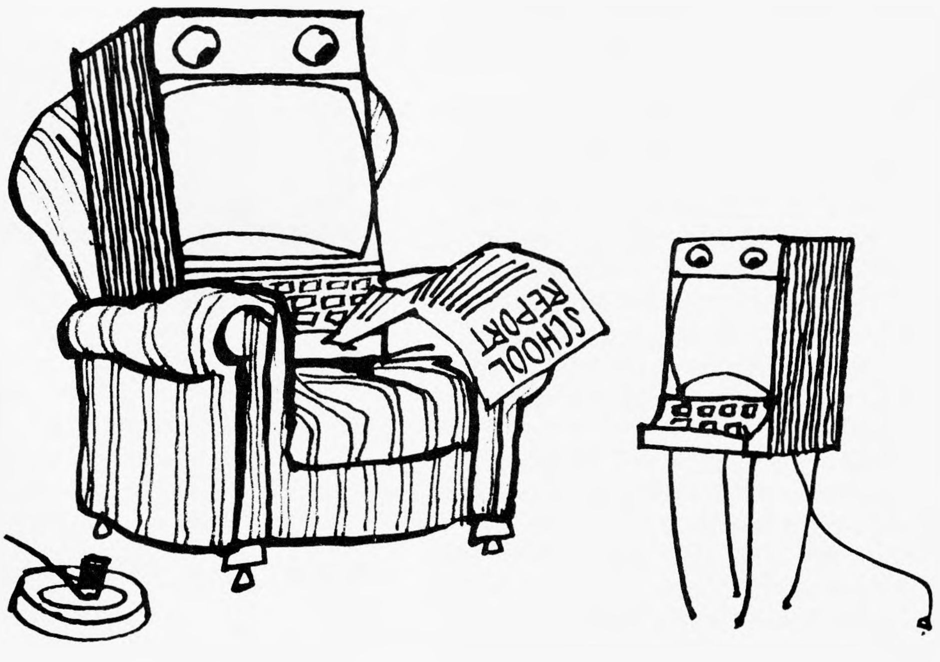


Fig. 7

We then moved the turtle across and got it in the right direction and drew house again for street.

Elaine Shepherd
Helen Jones
Kathryn Syms
Clairann Jeal

```
TO STREET :SIDE
 HOUSE :SIDE
 LT 90
 PU
 FD ( :SIDE * 7 / 4 )
 RT 90
 FD :SIDE
 PD
 HOUSE :SIDE
END

TO RECT :SIDE
 REPEAT 2 [FD :SIDE/2 LT 90 FD :SIDE/4 LT 90]
END

TO FOURSQUARE :SIDE
 SQUARE :SIDE / 8
 FD :SIDE / 8
 SQUARE :SIDE / 8
 LT 90
 SQUARE :SIDE / 8
 LT 90
 FD :SIDE / 8
 RT 90
 SQUARE :SIDE / 8
END

TO SQUARE :SIDE
 REPEAT 4 [FD :SIDE RT 90]
END

TO TRIANGLE :SIDE
 RIGHT 90
 REPEAT 3 [FD :SIDE LT 120]
END

TO HOUSE :SIDE
 TRIANGLE :SIDE
 SQUARE :SIDE
 RT 90
 FD :SIDE / 5
 LT 90
 PD FOURSQUARE :SIDE
 RT 180
 FD :SIDE / 8
 LT 90
 PU
 FD ( :SIDE * 7 / 8 )
 LT 90
 PD
 FOURSQUARE :SIDE
 PU
 FD :SIDE / 4
 LT 90
 FD :SIDE - ( :SIDE / 4 + :SIDE / 8 )
 RT 180
 PD
 RECT :SIDE
END
```



'It says you're fast and accurate, but lacking in initiative and ideas of your own.'

One week with a BIGTRAK

Henry Liebling
Park County Primary School, Colne, Lancashire

After reading Mike Thorne's article on BIGTRAK, I asked the school after assembly if anyone had one of these machines. Six hands shot up: one BIGTRAK was broken, but five were intact.

The next day during the lunch hour I was iniated: two of the boys with two years' experience patiently explained to me how to get BIGTRAK to perform its magic. A sea of eager faces peered as the five machines were put through their paces, and soon an obstacle course was constructed. I explained the need to write down programs when planning longer and more complex routes, and also so that improvements could be more easily incorporated. We found it particularly useful to write down the programs when using the 'X2' key as this excellent function requires the user to enter the number of steps that must be repeated. Our enthusiasm resulted in meeting again after school, by which time a route to the boys' toilets had been devised.

That Saturday I rushed out to buy a BIGTRAK. At home my own children and their friends were equally interested. Even after three to four hours' use a day for only one week, I can readily confirm that many of the benefits suggested by Mike Thorne and Ruth Russell are easily observed and achieved when children get to grips with BIGTRAK. Hopefully this is only a beginning.

I have for a long time feared that the greatest resistance to real change in education is found within the teaching profession. This may be particularly true for the concept of computers in the primary sector. Although the staff agreed, back in November 1981, to let me order a BBC 'B' machine (using prize money from last year's Granada Power Game and help from the local authority) most of them are apprehensive, some almost fearful that they won't be able to cope when it arrives.

Could I believe my eyes? Members of staff down on hands and knees with a BIGTRAK? Excited faces as if it was their 16th birthday? Could I believe my ears? Staff arguing as to who could take the machine home to try it out? One lady, who is opposed to Space Invaders and noisy games, and who admits to having hated computer studies at college, was very encouraged by how well the children coped, and is eagerly awaiting her next opportunity to try her hand at programming! The timer 'P' and 'turning' seemed to her good new experiences for children. Another teacher who usually stays away from things mechanical not only thoroughly enjoyed herself learning to use the machines, but very quickly pointed out what an excellent introduction to microcomputers this was.

At £30 including batteries this 'toy' must be the most cost-effective tool we've seen for ages – and not just for softening teachers' attitudes to computers.

Lest I sound like an advert for BIGTRAK, I do have reservations about the photon gun – perhaps it could be modified to do something less bellicose, like seed planting or road marking.

'I suppose it says something of this day and age.'

Logic as a computer language

J. R. Ennals
Department of Computing, Imperial College, London SW7

There are snares awaiting the enthusiastic teacher who introduces computers into his classroom. He may find himself passing on to tomorrow's generation today's failed languages based on yesterday's technology (going back to various BASICs). Microelectronics may be taught by stripping down the obsolete Model-T systems disposed of by local industrialists. Computer assisted learning packages may be tackled, like Everest, because they are there and have a manufacturer's seal of approval.

Such approaches may once have seemed appropriate, but they will no longer satisfy the discerning child or sceptical teacher, who remain unimpressed by much of what is offered in the name of educational computing.

An answer is found by having recourse to logic, which for two thousand years has been one of the bases of our Western academic tradition. Teachers of all disciplines have tried to encourage clear thinking and rational argument. University computer science departments regard these as essential qualities for computer programmers, qualities that are paradoxically being threatened by the spread of mediocre computer education.

In logic programming, using a language such as PROLOG (PROgramming in LOGic), a descriptive sentence expressed in logical form can also be regarded as an instruction to the computer. We can concentrate on the correctness of our description without being concerned about the workings of the computer. A description, or specification, of a problem will run as a program, though not necessarily very efficiently. Using only logic, and no control information, a wide range of programs can run relating to any subject matter.

The project 'Logic as a Computer Language for Children', supported by the Science and Engineering Research Council and the Nuffield Foundation, has been developing teaching materials for children from the age of 10 at Park House School, Wimbledon, using micro-PROLOG, an implementation of PROLOG for microcomputers with the Z80 microprocessor and the CP/M operating system.

The children's work has been at the frontiers of computer science research, and is fully consistent with the recently announced Japanese plans to develop a fifth generation of computers, to be based on logic programming. At Ricards Lodge School a new course in Information Technology is being developed, introducing word processing, data processing and programming through PROLOG.

Let us illustrate the basis of logic programming with a simple program describing our royal family. First a few statements of fact:

```
Elizabeth  mother-of  Charles
Philip     father-of  Charles
Elizabeth  mother-of  Ann
Philip     father-of  Ann
```

Then, a definition of what it is to be a parent:

```
x parent-of y if x mother-of y
x parent-of y if x father-of  y
```

We can ask questions:

English Is Philip the father of Charles?
PROLOG Does (Philip father-of Charles)
YES

This query checks for the presence of this sentence in the database.

English Who are Ann's parents?
PROLOG Which (x x parent-of Ann)
Answer is Elizabeth
Answer is Philip
No (more) answers

The 'parent-of' program tells the computer to solve the problem of who is parent of whom by finding out about who is mother of whom or who is father of whom, by searching the 'mother-of' and 'father-of' relations.

Children progress easily from using interactive programs, to querying databases, and to writing their own programs. Their first simple programs will be written in their first week of a course.

Programs written by the children have dealt with such subjects as: the periodic table of the elements, Dallas, geographical information, French translation, solar system, football, Wimbledon tennis, school timetable, calories in foods, Eighteenth-Century Poland, cars . . .

The basis of our work is logic. A correct description will run as a program, or can be used as the basis for a further program in PROLOG or another language. We are not tied to a particular generation of hardware, but our ideas form the basis for the computers that are being designed in this country, Europe and Japan for the 1990s. It is easier for children, with their flexible minds, to learn logic programming than it is for veterans of conventional languages. PROLOG has been described irreverently as 'PRObably the Language Of God'. Certainly, biblical advice to 'change and become like children' seems appropriate, in preference to the problems of 'COMAL going through the eye of a needle'.

INFORMATION RETRIEVAL

Learning history with the help of a microcomputer

USING CENSUS RETURNS

This is the first of two articles by Dr Alistair Ross on applications to history and social studies. The microcomputer is used to store, sort and retrieve original data in powerful and thought-provoking ways.

Dr Ross teaches at Fox Primary School, ILEA. He was formerly advisory teacher for history and social studies in primary schools in ILEA, and is a member of the Schools Council History Committee.

Dr Ross raises many important general issues on which we welcome comment. Are the children learning something new and previously inaccessible? Are gains in speed and motivation offset by distractions?

Fox Primary School is one of a small group of ILEA primary schools experimenting with the 380Z Research Machines Limited microcomputer. We are trying to find ways in which the computer can assist learning in traditional areas of the curriculum, rather than to introduce computer studies *per se* or devise particular new activities for children in which the micro can be used.

My particular interest is in the area of social studies and history: this article describes how my class of 9 and 10 year olds have been using the 380Z to develop their understanding in history. A subsequent article will extend this into the social studies area.

Approaches to history teaching

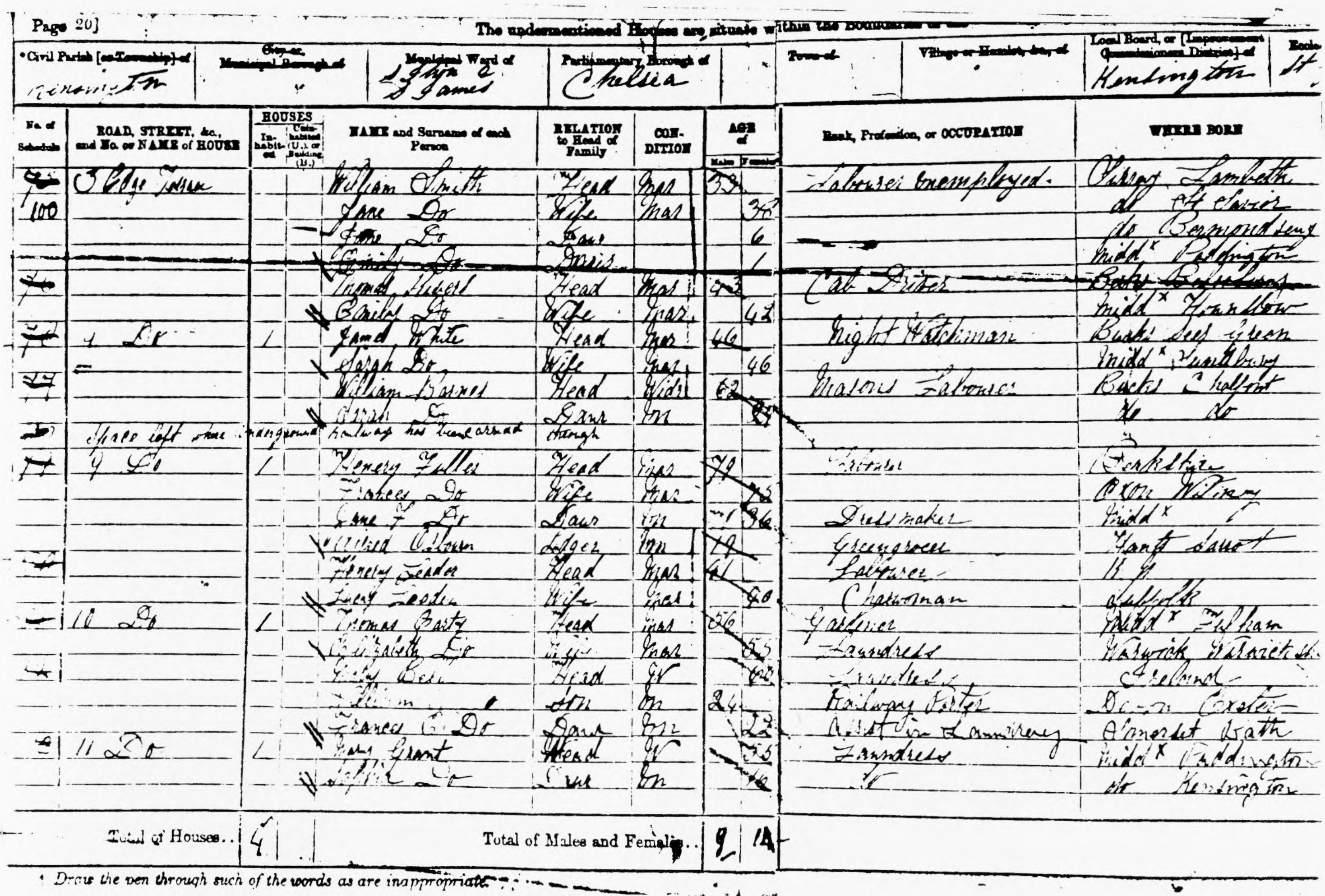
The ways in which children learn about the past have changed in recent years. The HMI Report *Primary Education in England* (1978) suggested that children should be making in-depth studies of particular periods, developments and localities, and building up understandings of historical change and cause and the nature of historical evidence (5.123 – 5.136). By concentrating on details of the social and economic life of individuals and small groups, children can effectively construct generalisations about a period: this is much more difficult for them if they are taught the other way round, by being given abstractions already made by the teacher. Groups such as the peasants or the middle classes are difficult for a young child to understand, and power figures (such as the King) are very distant. A much better sense of empathy can be developed through the study of individuals, to whom children can relate.

Another development has been to emphasise the value of historical evidence in the classroom, so that rather than learning facts that have already been synthesised, children learn skills of how to hypothesise, and ask questions about *how* we know about the past, and not just *what* we know.

Certain local records are available for many areas which allow a detailed reconstruction of social life to be made. Such exercises are particularly valuable in local history studies, when the area is limited to the few streets around the primary school which the children know well. It is much easier to understand the changes that have occurred in the last century if one can use information about named individuals to develop broad generalisations. It is even better if the houses that they lived in can still be seen, drawn and photographed.

Census material

I have found that census records are particularly useful. A census has been made in the United Kingdom every ten years since 1801 (with the exception of 1941). These records are confidential for a hundred years, and then they are available for consultation at the Public Records Office. Microfilm copies of the area covered by a local authority are usually held at the local borough's principal library for the years 1841, 1851, 1861, 1871 and, from this January, 1881. The information held for these years offers a wealth of material on individuals and their households. The precise details vary from year to year. Generally the later censuses were more detailed and more accurate. As this article is concerned with work done by children on the 1871 records, it is this example which is shown in Figure 1.



Page 20] The undermentioned Houses are situate within the Boundaries of the

Civil Parish [or Township] of Kensington | City or Municipal Borough of | Municipal Ward of Light St James | Parliamentary Borough of Chelsea | Town of | Village or Hamlet, &c., of | Local Board, or [Improvement Commissioners District] of Kensington

No. of Schedule	ROAD, STREET, &c., and No. or NAME of HOUSE	HOUSES Inhabited	HOUSES Uninhabited (U.) or Building (B.)	NAME and Surname of each Person	RELATION to Head of Family	CONDITION	AGE of Males	AGE of Females	Rank, Profession, or OCCUPATION	WHERE BORN
100	3 Edge Terrace			William Smith	Head	Mar	38		Labourer unemployed	Surrey Lambeth
				Jane Do	Wife	Mar		38		do St Saviour
				Jane Do	Daur			6		do Bermondsey
				Emily Do	Daur			1		Midd^x Paddington
				[illegible]	Head	Mar	43		Cab Driver	[illegible]
				[illegible] Do	Wife	Mar		42		Midd^x Hounslow
	4 Do	1		James White	Head	Mar	46		Night Watchman	Bucks [illegible] Green
				Sarah Do	Wife	Mar		46		Midd^x [illegible]
				William Barnes	Head	Widr	62		Masons Labourer	Bucks Chalfont
				[illegible] Do	Daur	Un		[illegible]		do do
	Space left where railway has been carried through									
	9 Do	1		Henry Fuller	Head	Mar	79		[illegible]	Berkshire
				Frances Do	Wife	Mar		78		Oxon Witney
				Jane F Do	Daur	Un		36	Dressmaker	Midd^x
				Alfred [illegible]	Lodger	Un	29		Greengrocer	Hants [illegible]
				Henry [illegible]	Head	Mar	41		Labourer	N K
				Lucy [illegible]	Wife	Mar		40	Charwoman	Suffolk
	10 Do	1		Thomas [illegible]	Head	Mar	56		Gardener	Midd^x Fulham
				Elizabeth Do	Wife	Mar		55	Laundress	Warwick [illegible]
				[illegible]	Head	W		62	Laundress	Ireland
				William [illegible]	Son	Un	24		Railway Porter	Devon Exeter
				Frances C Do	Daur	Un		22	Assistant in Laundry	Somerset Bath
	11 Do	1		Mary Grant	Head	W		55	Laundress	Midd^x Paddington
				[illegible] Do	Daur	Un		16	Do	do Kensington
	Total of Houses	4		Total of Males and Females			9	14		

* Draw the pen through such of the words as are inappropriate.

Fig. 1 From the 1871 census

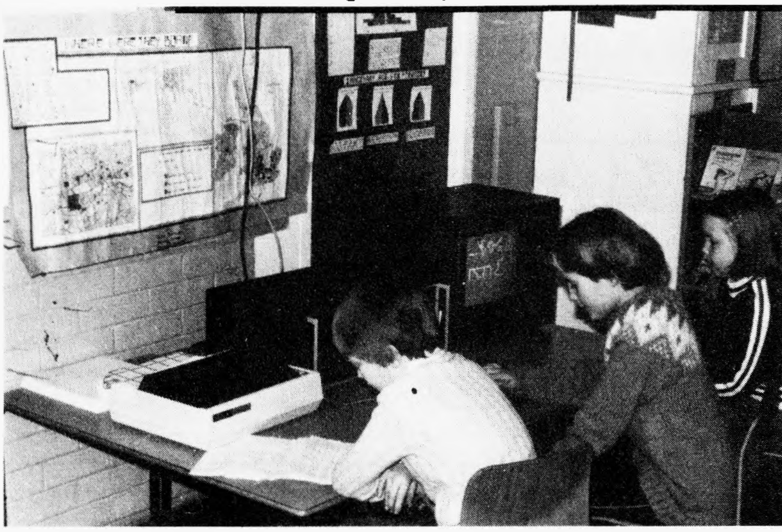
These records were compiled by the local census enumerator. Each household was given a schedule number, and each member of the household had individual information given on one horizontal line. This consisted of: name, address, condition (whether married, unmarried or widowed), relationship (to the head of the household), age and sex, occupation, and place of birth. Enumerators' handwriting varies from copperplate to barely literate scrawl, but most entries can usually be deciphered with a little practice, even by untrained junior-aged children. I have been using census records for several years in my teaching, and each time I am surprised anew by just how well children can cope!

There are problems, however. We can build up details of the locality in the past, but the work of compiling, say, a map of migrations into the city (from the 'birthplace' column) is painstaking. So is the construction of even small bodies of statistics on local occupations, or family size, or overcrowding. I usually supply my class with photocopies of census sheets mounted on board and covered with transparent film. Each card contains 20 – 25 names: there are seven cards for Edge Terrace, the street our school now stands in. It is difficult to make tally charts from cards in the classroom. Figures get mislaid; words that had once been deciphered are forgotten; cards in use by one group are needed by another. Nevertheless, we have in the past been able to do quite a lot, even though most of our time was spent bogged down in what became repetitious tasks of data collection rather than in making historical deductions of what the street must have been like.

Using the micro

The introduction of the 380Z microcomputer offered us a means of overcoming these difficulties. I already knew of Beverly Labett's pioneering work on census returns with a mainframe computer in East Anglia. The ILEA had developed a series of general purpose programs to make, maintain and search files of information called Micro LEEP (Luton Educational Enquiry Package).* I therefore constructed a file of information based on the 1871 census of

*Micro LEEP was developed by Malcolm Hall, Judy Challinor and the ILEA Educational Computing Service from the original Systime LEEP developed by John Marshall of Luton College of Higher Education. LEEP presently consists of five basic programs: LEEP (a database enquiry program), LEEPS (a shortened enquiry program for simple enquiries), LEEPDIR (a directory management program), LEEPFILE (a file management program) and LEEPSORT (a multi-key file sort program). Further details are obtainable from Microcomputer Support, ILEA Educational Computing Centre, Bethwin Road, London SE5 OPQ.



our local street, which we called FOX 1871, and the class keyed in the relevant information from our seven cards of enumerators' returns. We then used the microcomputer to sort and categorise this information in a number of different ways.

In constructing the file, I had three linked principles in mind. Firstly, although it would be possible (and more economical of computer memory) to code many of the filed names and entries, this was undesirable in that it might act as a barrier to understanding by the child user. For example, I thought it generally easier for children to read that 'William Smith' was recorded as 'Head' of household, rather than that 'W. Smith' was 'H'. Secondly, I think it inherently wrong to edit original data unnecessarily: I wanted to preserve the authenticity of the original as far as possible. But sometimes editing is necessary. For instance, the enumerator frequently used ditto marks, which are perfectly understandable when the file of data is preserved in a serial order on the page. But if one retains authenticity by recording ditto marks in a computer file, and then shuffles the entries in a sorting process, the printout of ditto marks will be at best meaningless and probably completely misleading! Obviously, therefore, some data must be modified.

My third aim was to make the computer-held files as flexible as possible to use. To do this I added fresh fields that allowed the encoding of certain data. For example, if a future file was to include, say, the 1881 census data for the same area, it would be useful to compare the occupants of the same house in 1871. But street numbering systems and street names themselves changed. Therefore two new fields were added, an x and a y coordinate, to define the location of the building (using contemporary maps). A comparison of buildings could thus be made by the input of coordinates rather than addresses. There are further and more advantageous uses of these coordinate fields. It should be noted that such accretions do not sully the authenticity of the original evidence; this is still recorded in full, and can be output in a form similar to its original state, without any of the new fields.

Nineteen fields were established: a brief description of each is given in the table below.

Field	*Length*	*Type**	*Comment*
SCHED†	3	n	Schedule number, given to each household by the enumerator. Everyone in the same household has the same schedule number.
HOUSENO†	25	a	Name of street, and number (or name) of house.
HSELOCE	3	n	x coordinate (east–west axis) for location of house.
HSELOCN	3	n	y coordinate (north–south axis) for location of house.
FORENAMES†	20	a	Forenames (and initials) of individuals. Spelt as recorded by enumerator.
SURNAME†	16	a	Surname of individual.
RELAT†	8	a	Relationship, sometimes abbreviated slightly (e.g. MOTH-I-L).
CONDIT†	1	a	Condition (married – M, unmarried – F, widowed – W).
SEX†	1	a	Male – M, female – F.
AGEY†	2	n	Age in years. Infants under 1 recorded here as 0.
AGEM†	2	n	Age in months for children under 1, all over 1 left blank.
TRADE†	25	a	'Occupation, profession or condition', recording as much as possible on the schedule, as written.
TCODE	5	n	A five-figure coding for 19th century occupations, grouped according to type of work, published in *Nineteenth Century Society*, ed. E. A. Wrigley (1972).
BIRTHT†	15	a	Town of birth (if given).
BIRTHC†	10	a	County of birth (or country).
BIRTHLOCE	3	n	x coordinate (east–west) for place of birth (see below).
BIRTHLOCN	3	n	y coordinate (north–south) for place of birth.
SPARE1	3	a	Spare fields allowing children to create temporary classifications of the file for particular purposes.
SPARE2	3	n	

* n = numerical data
a = alphabetical data
† = information direct from the enumerator's return

The BIRTHLOCE/N fields allow the place of birth to be plotted on a map. No single map would be suitable for this purpose, as a large scale is needed for the many people born in the locality and smaller scales for others born at a distance, or even in other countries. So a plan as shown in Figure 2 was used. The birthplace was found only on the most appropriate of the four maps, and coordinates found for this position. Since the same grid of coordinates covers all four maps, one pair of coordinates will also show the correct map.

The two sets of location fields (HSELOCE/N and BIRTHLOCE/N) were particularly useful in describing a given area to the computer. Without them, if one wanted to find all those born in East Anglia, one could institute a search for COUNTY = NORFOLK OR COUNTY = SUFFOLK and trust that the counties had been correctly spelt. If the enumerator had only given

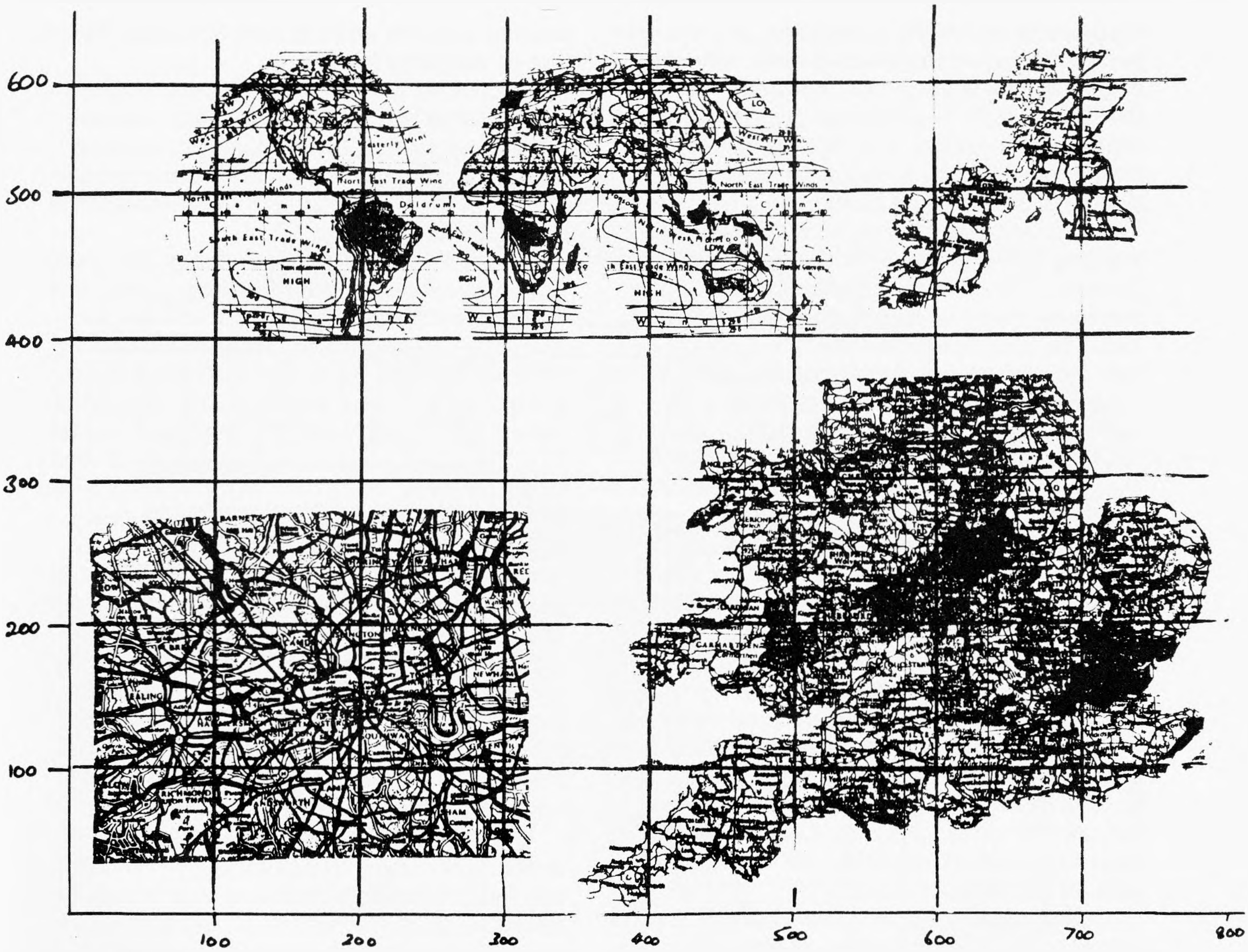


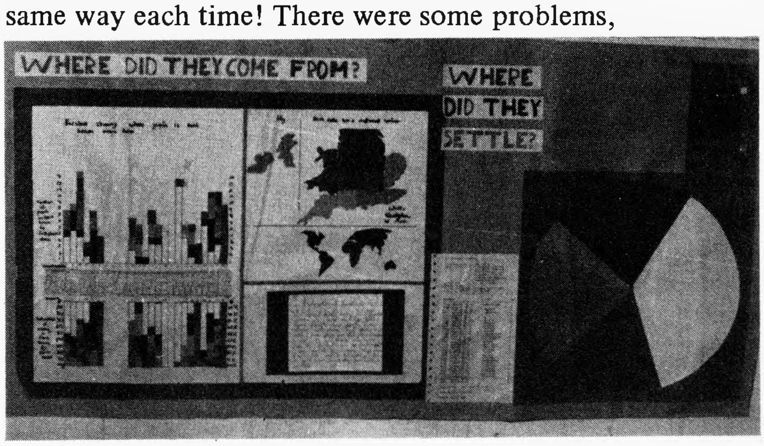
Fig. 2 Plan for locating birthplaces

the *town* of birth, say Norwich, for one individual and this information had been correctly entered in the computer-held record, then this person would not be found in the above search. Using the location system, one can instead search for BIRTHLOCE > 710 and BIRTHLOCN > 170 and < 290 to define an area.

The TCODE field can be used in a similar way. TCODE > 30000 and < 30400 will define all those in the construction industry. Simply asking for TRADE = BUILDER will not find, for example, those who described themselves as a 'bricklayer', or 'building labourer'.

The file having been constructed, the next stage was to feed in all the data relating to the street on which the school stood. Initially, this was done directly from the enumerators' returns by the children. Three or four children in a group took it in turns to key in their entries. The LEEP FILE program supplies each field name as a cue. The next group stood behind and watched the process, waiting for their turn – and at the same time discovered how a VDU worked, the delights of the DELETE key, and so on! After a day's work, all the class had had a turn, and a file of 28 names existed. I made a couple of searches to demonstrate the capabilities of the program and a discussion followed on whether we could search for all the old people, or how we would first need to determine what ages we meant by 'old'.

After the first day, we prepared data capture sheets to collect the information before keying in. This was much easier. Around the class were the necessary reference books: the TCODE index, an atlas with a gazetteer of the British Isles and an index we made as we went along of BIRTHLOCs. This latter index was useful in ensuring that each location was recorded in the same way each time! There were some problems,



particularly where the enumerator had not properly determined the place of birth: where a county only was given, we allocated BIRTHLOCE/N to a notional position in the centre of the county. The TCODE index did not have every trade, so occasionally the coding of a similar trade had to be given as an alternative.

The class completed capture sheets on the remaining 120-odd names in the street, and keyed in about 100 of them. I finished the rest. We were now ready to start using the search facilities. The LEEP program, which is under revision, has several facilities. It will sort out all those with a particular characteristic within a field, or having particular characteristics within several fields. It can then output any of the fields of the individuals so selected – which, of course, need not always be those searched for (there is no need to print condition after asking for CONDIT = M). And it can sort out the selected records into an order (alphabetical for 'a' fields, numerical for 'n' fields). The results can be displayed on the screen, or printed out. Initially, lots of enquiries were made by the children simply to discover the mode of questioning and the range of power of the microcomputer. Some children made their tests for fun – for example, finding out everyone whose name began with the same letter as their own. Others made perhaps more serious enquiries: for example, what jobs did all the old ladies do?

Only gradually did we come to appreciate the power of the sorting facility, as opposed to the enquiry facility. We then often found it useful to sort information for everyone on the street, and simply to order them. For example, we searched for SEX = M OR SEX = F (everyone; alternatively AGEY > 0 – we did not discover till later that we could simply ask for ALL!) and had as our output SURNAME and TRADE, and sorted on TCODE. This gave us a list of everyone with his job, and with the jobs grouped together conveniently in types: all those engaged in construction, followed by all those in domestic service, and so on. Those with no trade (usually housewives and children) brought up the rear (their code being 99999).

We then began a short systematic analysis of society on Edge Terrace in 1871. I grouped the class into twos and threes, and gave each group a particular facet to explore. One group looked at where people were born, and another considered if and how age was related to place of birth. Other groups looked at conditions of crowding on the street (how many families shared a house), at the range of trades, at how people on the street were different – were all the people in the same trade living in adjacent houses, or did people born in the same area stick together? Others simply used the computer facility to print data clearly, and looked at particular families. It is often possible to determine from a chain of children's birthplaces the migration pattern of the parents.

A period of intense use of the microcomputer was followed by less frequent checks and subsearches by groups. Interestingly, its novelty soon wore off and the computer came to be regarded as just another source of information in the classroom.

The photographs that accompany this article display some of the work the groups produced. The group looking at birthplace and age, for example, first determined three age groups by dividing the total list of ages into three equal groups, so that there were as many 'old' people (over 35) as there were 'young' people' (under 21). They then plotted the birthplaces of each group on maps, using both BIRTHLOCE/N and BIRTHT to help them. They observed that young people tended to be born very locally, and older people born outside the area (and indeed outside London). A hypothesis was made: Kensington, 40 years before 1871, had been very small and few people were born there. In the intervening period, people had moved in from the country and their children had been born locally. They then took this further, and plotted the distance from London of each person's birthplace against age.

The group relating trades to houses on the street discovered that groupings occurred. All the washerwomen/laundresses lived in five adjacent houses in the middle of the street. The construction workers tended to live at the ends of the street, with the plasterers in particular living in two adjacent houses. Why? Perhaps, they speculated, they had had similar jobs in the country, and moved together. Or perhaps they had passed on news of job opportunities in their own work place to their next door neighbours.

Future projects

This particular project only looked at one street of houses in a single year. We now want to progress to compare adjacent streets – some streets to the south have larger houses than our own, and were probably occupied by much more middle-class people. And, of course, we will want to move on to other years and make comparisons.

We are also trying to write a program that will plot the location coordinates for us on a map on the screen. It would be exciting and useful to plot on the screen a local map showing, for example, all the households with servants.

Meanwhile, the same LEEP package has been used in school in other projects. In history, we have made a file of the street directories for 1891–1950 of the area. And in social studies, we are creating a file of the results of a survey of people's opinions on the work that they do. A future article will describe these developments.


DATA BANK
HELP
DAVID BARLOW

Program listing: PQUERY

Roger Keeling
Newman College

In previous issues of *MICRO-SCOPE* I have described different teaching programs in mathematics and language. These fit neatly into the category of Computer Assisted Learning. Yet this is only one aspect of the influence of microcomputers on the primary school curriculum, albeit the most popular one at present. The future will see increasing emphasis on Computer Managed Learning, Control Technology and Information Retrieval.

This article shows how a class of junior children can learn to handle the ideas contained within the phrase 'Information Retrieval'. The programs used (PQUERY and PEDIT) are similar in principle to a subset of the Hertfordshire MICROQUERY, but are written to run on the RML 480Z. The program takes advantage of the fact that with BASIC in ROM and the high-resolution option board installed, the user has available 32K of memory. This enables a teacher to load on to the computer a substantial bank of information for interrogation without having to manipulate files in a cassette system.

The effective use of the program depends very much on the teacher. He must decide on a suitable data bank and know the right questions to ask in order that the pupils will acquire an appreciation of the ideas involved. Three examples are given below.

Example 1

Data bank: Geographical towns.
Information stored: Name, population, area, height above sea level, river, industry.
Typical problems: Which towns are over 250 metres above sea level?
How many of these towns are on a river?
List all the towns in descending order of population.

Example 2

Data Bank: Famous explorers.
Information stored: Name, dates, region of exploration, mode of transport, nationality.
Typical problems: Which explorers have been to the Antarctic?
How many exploration journeys lasted more than 2 years?
Where did Marco Polo travel?

Example 3

Data bank: The weather.
Information stored: (For each day of a month) cloud type, rainfall, hours of sunshine, maximum temperature, humidity.
Typical problems: On which days was there rainfall and a maximum temperature in excess of 15°C?
How common is a cumulus cloud formation?
List the wettest 5 days in order.

For the purpose of this article I want to take as a data bank the pupils themselves. The program PEDIT is used to create this file. The teacher can specify whatever field names he wishes. Then the appropriate records are stored for each pupil in the school (in this example, up to a maximum of about 140 records). An example is given here:-

Field Names	*A typical record*
1 SURNAME	:BROWN
2 FORENAMES	:KEITH
3 CLASS	:6
4 ROAD	:NORTHSIDE DRIVE
5 DOB	:74.05.09
6 HEIGHT	:133
7 WEIGHT	:26
8 EYES	:BROWN
9 HAIR	:BROWN
10 SEX	:M
11 BROTHERS	:0
12 SISTERS	:0

PQUERY is then used to analyse the data. The full list of options is given below:

Function?
1) Read data file
2) Select printer options
3) Search data
4) Sort data
5) Display/print data
6) Display field names
7) End program

The most powerful options are the facilities to SEARCH and SORT. These can operate on any field name or combination of fields. Note that SEARCH means 'find a specified subset' while SORT means 'put into a specified order'. A sort on a numeric field (e.g. HEIGHT) gives a list of all pupils in ascending order, while a sort on an alphabetic field (e.g. SURNAMES) gives a dictionary listing of all the pupils. A sort on a field name such as HAIR uses the alphabetic order of the colour-words: all the pupils with

BLONDE hair will appear first, followed by all those with BROWN hair, etc. The effect is to list the subsets of pupils according to hair colour. For example, to find the oldest girl, search the records for all the girls and then sort this subset. To find all the brown-haired children over 130 cm tall, firstly search the records for all pupils whose HEIGHT is over 130 cm and then search this subset again for all the children whose HAIR is brown. This example is given in more detail opposite (responses underlined).

In this printout the option to *invert* enables the teacher to reverse the condition. For example, 'contains' becomes 'does not contain' and so I could find all the pupils whose surname does not contain an 'e'. (This subset is the complement of the subset obtained from the original condition).

After a couple of demonstration runs in the classroom the aim is to get the pupils involved in using the program to interrogate the data bank. The important point here is to build up some idea of what we expect the correct answers to be. There is no problem with finding the date of birth of John Daines or the road in which Mary Haskins lives. There may be more debate about who is the tallest boy in the class. Can we find the names of all the blue-eyed girls who live in Watery Lane? Every time we should be comparing the information the computer finds with our expected answers based upon observations. By this time the pupils should begin to appreciate the capabilities of the program, and we can alter the problems accordingly.

Which of these are realistic requests for information?

1. All the boys in class 6 born in November. (How do you find 'born in November'?)
2. How many sets of twins are in the school?
3. Can we list the pupils in order of age?
4. What happens if we sort the information according to EYES? Is this sensible?

An imaginative teacher will use the program in several different ways. Creative writing perhaps?

'When I was in town recently I saw someone from either class 6 or class 7 out shopping with her mother and sister. She was trying on some dresses at Tammys; probably a size 10. Her hair was black because I remember the sharp contrast it made with the white dresses she was looking at. By coincidence we happened to be on the same bus back home, but she got off before me at the Lazywood Road stop.'

Q. What is her name?

Does the passage say anything useful about height or the road in which she lives? Perhaps the pupils need to get a dress size conversion chart and a map of local roads. What do I search for with respect to sisters? Is this the way the police work when narrowing down suspects? – now that leads me to an idea for a project!

```
Search number 1


What field do you want to operate on
(input the name) ? HEIGHT

What condition do you want:

1) contains
2) is identical to
3) precedes
4) succeeds
5) is less than
6) is greater than   6

Invert (Y/N) ? N
Operand ? 130

Search condition:

HEIGHT is greater than 130

Is this correct (Y/N) ?
Searching...

Searched 62
Matched 24

Do you want to list records (Y/N) ? Search number 2

Do you want to:

1) Add to the list (OR)
2) Narrow down the list (AND)
3) End the search    2

What field do you want to operate on
(input the name) ? HAIR

What condition do you want:

1) contains
2) is identical to
3) precedes
4) succeeds
5) is less than
6) is greater than   2

Invert (Y/N) ? N
Operand ? BROWN

Search condition:

HAIR is identical to BROWN

Is this correct (Y/N) ?
Searching...

Searched 62
Matched 9

Do you want to list records (Y/N) ? Search number 3

Do you want to:

1) Add to the list (OR)
2) Narrow down the list (AND)
3) End the search    3
```

```
10 REM pico query Ver 2.1  N. Osborne  24.2.82
40 PUT 31:?:?:?
50 ?"Pico-query":?"==========":?:?"Function ?":?
60 ?"1) Read data file":?"2) Select printer options
80 ?"3) Search data":?"4) Sort data
100 ?"5) Display/print data":?"6) Display field names
107 ?"7) End programme   ";
110 GOSUB9600:A=GET()-48:IF A<1 OR A>7 THEN 110
120 ON A GOTO 1000,2000,3000,4000,5000,1500
130 ?:?:?:END
999 REM read data file
1000 CLEAR3000:PUT 31:?:?:?"Name of data ? ";:ML=6:GOSUB9000:DIMST(200)
1010 ?:?:?"Rewind cassette.":?"Press 'PLAY' on recorder"
1025 ON ERROR GOTO 9500
1030 ?:OPEN£10,A$
1040 INPUT£10,NF,NR,NR:DIMD$(NF,NR):DIMF(NR),S(NF),V(NR)
1050 FOR K=0 TO NR:FOR J=1 TO NF
1060 INPUTLINE£10,D$(J,K):NEXT:NEXT
1065 FOR K=1 TO NR:V(K)=K:NEXT
1070 ?:?"Data read - switch off recorder":DR=-1:GOSUB9600:U=GET(400):GOTO 40
1500 PUT 31:?"Fields:":?:IF NOT DR THEN?"No data yet read !":U=GET(400):GOTO40
1510 FOR K=1 TO NF:?K;D$(K,0):NEXT
1520 ?:?"Press space bar to continue  ";:GOSUB9600:U=GET():GOTO 40
2000 PUT31:?:?:?"Printer options":?"===============":?:
2005 D=0:PRINT"Printer type (2-4) ? ";:GOSUB 8000
2010 ?:IFNN<2ORNN>4THEN2005ELSEPT=NN
2015 IFPT=3THENPB=0:GOTO 2035
2020 D=0:PRINT"Baud rate (0-6) ? ";:GOSUB 8000
2030 ?:IFNN>6THEN2020ELSEPB=NN
2035 ?:?:?"Max. number of characters a line? ";:GOSUB 8000:PW=NN:GOTO 40
3000 PUT 31
3002 S$="                              "
3003 ?"Search":?"======":?
3005 IFNOTDRTHEN?"No data read in !":U=GET(400):GOTO40
3010 FOR K=1 TO NR:F(K)=0:NEXT:SN=0:DS=-1
3015 SN=SN+1:PUT 31
3020 ?"Search number"SN
3025 ?:IFSN=1 THEN CN=1:GOTO3080
3030 ?"Do you want to:
3035 ?:?"1) Add to the list (OR)"
3040 ?"2) Narrow down the list (AND)  "
3045 ?"3) End the search  ";
3049 GOSUB9600
3050 A=GET(0):CN=GET()-48:IFCN<>1AND CN<>2 THEN IF CN=3 THEN 40 ELSE 3050
3080 ?:GOSUB 7000
3090 ?:?"What condition do you want:"
3100 RESTORE 7100:FORK=49 TO 54:?:READ A$:?CHR$(K)+") "+A$+"  ";:NEXT
3120 GOSUB9600
3130 A=GET(0):CC=GET()-48:IFCC<1 OR CC>6 THEN 3130
3140 ?CC:?:?"Invert (Y/N) ? ";
3145 GOSUB9600
3150 A=GET(0):A=GET():A=A+32*(A>95):IF A=89 THEN CC=-CC ELSE IF A<>78 THEN 3150
3160 PUT A:?:INPUTLINE"Operand ";O$
3170 PUT31:?:?"Search condition:":?
3180 RESTORE7100:FOR K=1 TO ABS(CC):READA$:NEXT:IF CC<0 THEN A$="not "+A$
3190 ?D$(KF,0)+" "+A$+" "+O$
3192 ?:?"Is this correct (Y/N) ? ";
3193 GOSUB9600
3194 A=GET(0):A=GET():A=A+32*(A>95):IFA=78 THEN PUT31:GOTO3020ELSE IFA<>89 THEN3194
3200 ?:?"Searching...":?
3210 M=0:FORK=1 TO NR:A$=D$(KF,V(K))
3220 ON ABS(CC) GOSUB 3500,3550,3600,3650,3700,3750
3230 R=R*SGN(CC)
3240 IFCN=1 AND R=1THENF(K)=1
3250 IFCN=2 ANDR=-1THENF(K)=0
3255 IFF(K)<>0 THEN M=M+1
3260 NEXT
3270 ?"Searched";NR:?"Matched";M:?
3280 ?"Do you want to list records (Y/N) ? ";
3283 GOSUB9600
3285 A=GET(0):A=GET():A=A+32*(A>95)
3286 IFA=78THEN3015ELSEIFA<>89THEN3285
3287 IFM=0THEN3015
3290 GRAPH:PUT31:PLOT0,57,"pico-query"
3292 PLOT24,57,"Search no."+STR$(SN)
3295 ?"Press space bar for next record "
3297 ?"Press 'esc' to abort  ";
3300 FOR K=1TONF:VH=54-K*3
3305 PLOT0,VH,D$(K,0):PLOT20,VH,":":NEXT
3310 FOR K=1TONR:IFF(K)=0THEN3400
3312 PLOT55,57,"Record:"+STR$(K)
3313 FOR J=1 TO NF:PLOT22,54-J*3,S$:NEXT
3315 FORJ=1TONF:PLOT22,54-J*3,LEFT$(D$(J,V(K)),29):NEXT
3320 GOSUB9600:A=GET():IFA=27THEN3410
3400 NEXT
3410 FOR K=1TO1:NEXT
3420 TEXT:GOTO3015
3500 R=-1:A=LEN(O$):IFA>LEN(A$) THEN RETURN
3510 FOR J=1 TO LEN(A$)-A+1
3520 IFO$=MID$(A$,J,A) THEN R=1
3530 NEXT:RETURN
```

```
3550 IF O$=A$ THEN R=1 ELSE R=-1
3560 RETURN
3600 IF A$<O$ THEN R=1 ELSE R=-1
3610 RETURN
3650 IF A$>O$ THEN R=1 ELSE R=-1
3660 RETURN
3700 IF VAL(A$)<VAL(O$) THEN R=1 ELSE R=-1
3710 RETURN
3750 IF VAL(A$)>VAL(O$) THEN R=1 ELSE R=-1
3760 RETURN
4000 PUT31:?"Sort":?"====":?
4005 IFNOTDRTHEN?:?"Data not read in yet !":U=GET(400):GOTO 40
4010 GOSUB 7000:?:?"Do you want to sort numerically or"
4020 ?"alphabetically (N/A) ? ";
4026 GOSUB 9600
4030 A=GET(0):A=GET():A=A+32*(A>95):IFA=78THENF=1ELSEIFA=65THENF=2ELSE4030
4050 ?:?:?"Sorting..."
4055 MS=0
4060 SP=0:FF=1:L=NR:GOSUB4062:GOTO40
4062 IF FF>=L THEN RETURN
4064 I=FF:J=L+1
4066 I=I+1:GOSUB4180:IFRTHEN4072
4068 IFI>=LTHEN4072ELSE4066
4072 J=J-1:GOSUB4190:IFNOTRTHEN4072
4074 IFI<JTHENT=V(I):V(I)=V(J):V(J)=T:T=F(I):F(I)=F(J):F(J)=T:GOTO4066
4076 IFFF<>JTHENT=V(FF):V(FF)=V(J):V(J)=T:T=F(FF):F(FF)=F(J):F(J)=T
4078 GOSUB4160:L=J-1:GOSUB4062
4080 GOSUB4170:FF=J+1:GOSUB4160:GOSUB4062:GOSUB4170:RETURN
4160 SP=SP+2:ST(SP-1)=J:ST(SP)=L:IFSP>MS THEN MS=SP:RETURN ELSE RETURN
4170 L=ST(SP):SP=SP-2:J=ST(SP+1):RETURN
4180 IFF=1THENR=VAL(D$(KF,V(I)))>=VAL(D$(KF,V(FF)))ELSER=D$(KF,V(I))>=D$(KF,V(FF))
4185 RETURN
4190 IFF=1THENR=VAL(D$(KF,V(J)))<=VAL(D$(KF,V(FF)))ELSER=D$(KF,V(J))<=D$(KF,V(FF))
4195 RETURN
5000 PUT 31:?"Display/print":?"=============":?
5003 IFNOTDRTHEN?"Data not read in yet !":U=GET(400):GOTO40
5005 IFPT=0THENCN=0:GOTO5040
5010 PRINT"To printer or screen (P/S) ? ";:GOSUB9600
5020 A=GET(0):A=GET():A=A+32*(A>95):IFA=83THENCN=0ELSEIFA=80THENCN=2ELSE5020
5035 PUTA:?:?IFCN=2THENPRINTERPT,PB
5040 IF NOT DS THEN CC=-1:GOTO5070
5050 ?"ALL the data, or just that selected":?"by searching (A/S) ? ";
5055 GOSUB9600
5060 A=GET(0):A=GET():A=A+32*(A>95):IFA=65THENCC=-1ELSEIFA=83THENCC=0ELSE5060
5070 FOR K=1TONF:S(K)=0:NEXT
5075 D$(0,0)="ALL"
5080 ?:?:?"Enter fields to be printed":?"(ALL for all, Exit with .)":?:J=0
5090 J=J+1:?"Field"J;:GOSUB7008
5100 IFKF=-1THEN 5120
5105 IFKF=0THENFORK=1TONF:S(K)=1:NEXT:GOTO5120
5110 S(KF)=1:GOTO5090
5120 T=0:?:FOR J=1 TO NF:IFS(J)=0 THEN 5200
5130 B=0:FORK=0TONR:IFLEN(D$(J,K))>B THEN B=LEN(D$(J,K))
5140 NEXTK:S(J)=B:T=T+B+1
5200 NEXT:D$(0,0)=""
5210 IFCN=0THENW=40ELSEW=PW
5215 IFT<W THEN F=1 ELSE F=2
5220 ?:IFCN=0THENPUT31
5230 IFF=2 THEN 5300
5240 T=0:FORK=1TONF:IFS(K)=0THEN5270
5250 PRINT£CN,TAB(T);D$(K,0);:T=T+S(K)+1
5270 NEXT K:PRINT£CN:PRINT£CN
5300 FOR K=1 TO NR
5310 IF F(K)=0 AND CC=0 THEN 5380
5320 T=0:FORJ=1TONF
5330 IFS(J)=0THEN5360
5340 IFF=1THENPRINT£CN,TAB(T);D$(J,V(K));:T=T+S(J)+1
5350 IFF=2THENPRINT£CN,D$(J,0);TAB(11);":"+LEFT$(D$(J,V(K)),W-12)
5360 NEXT J
5370 PRINT£CN
5380 NEXT K
5390 IF CN=0THEN?:?"Press space bar to return to options";:GOSUB9600:A=GET()
5400 GOTO 40
7000 ?:?"What field do you want to operate on":?"(input the name) ? ";:GOTO7008
7005 ?"Name ? ";
7008 ML=10:GOSUB9000:IFLEFT$(A$,1)="."THENKF=-1:RETURN
7010 ?:KF=-1:FOR K=0 TO NF:IF D$(K,0)=A$ THEN KF=K
7020 NEXT
7030 IF KF>=0 THEN RETURN ELSE ?:?"Field doesn't exist":GOTO 7005
7100 DATAcontains,is identical to,precedes,succeeds,is less than,is greater than
7999 REM numeric entry
8000 A$="":GOSUB9600
8005 A=GET(0):A=GET()
8010 IF (A<48ORA>57)*(A-127)*(A-13)*(A-46)THEN8005
8015 IFA<127THEN8030ELSEIFA$=""THEN8005
8020 IFRIGHT$(A$,1)="."THEND=1
8025 A$=LEFT$(A$,LEN(A$)-1):PUTA:GOTO8005
8030 IFA=13THENNN=VAL(A$):RETURN
8035 IFA=46THENIFDTHEND=0ELSE8005
8040 A$=A$+CHR$(A):PUTA:GOTO8005
8999 REM string entry
9000 A$="":GOSUB9600
9005 A=GET(0):A=GET():A=A+32*(A>95ANDA<123)
9007 IF A=27 THEN 40
9010 IF A<>127 THEN 9020
9015 IF A$=""THEN 9005 ELSE PUT A
9016 A$=LEFT$(A$,LEN(A$)-1):GOTO9005
9020 IFA=13THENIFA$=""THEN9005ELSERETURN
9025 IF(A<48ORA>57)AND(A<65ORA>90)THEN9005
9030 IFLEN(A$)=MLTHEN9005ELSEPUTA
9035 A$=A$+CHR$(A):GOTO9005
9500 IFERR<>8ANDERR<>15THEN9510
9505 ?:?:?"You have run out of memory..."
9507 U=GET(400):ONERRORGOTO9500:GOTO40
9510 ?ERL;:ERROR ERR
9600 UU=GET(0):RETURN
```

Listings of PEDIT can be otained by sending a stamped addressed envelope to Roger Keeling at Newman College.

A primary BASIC—part 6

John Fair
Newman College

In previous articles we have illustrated some ideas of DATA searches and we have seen how parts of words may be recognised. Now data can be processed in various ways – it can be *stored* or *retrieved*, *sorted* or *collated*, *updated* or *deleted*. Census returns are an obvious example. In this article we examine the ideas and techniques needed for sorting into order (numerical or alphabetical).

We start with a simple routine for rearranging the letters of 'MAPE' into alphabetical order. In Fig. 1 we first look at each pair of letters and check whether the order is alphabetical. If not, we exchange them. This is repeated for each successive pair of letters in the 'word' until the last pair. This is the first round (R=1). There are three comparisons (N=1,2,3) and two exchanges (X). MAPE becomes AMEP and the last letter alphabetically (P) has 'fallen' to the end. This last letter is now fixed and so is excluded from the next sorting round (R=2) where AME/P becomes AEM/P with two comparisons (N=1,2) and only one exchange (X). In this case the final round (R=3) makes no further alteration. Note that when R=1, 2 or 3 the number of comparisons is 3, 2 or 1 respectively.

```
(R=1)     [M     A     P     E]
      1    M  -  A     P     E
      X    A  X  M     P     E
      2    A     M  -  P     E
      3    A     M     P  -  E
      X    A     M     E  X  P
(R=2)     [A     M     E] /  P
      1    A  -  M     E  /  P
      2    A     M  -  E  /  P
      X    A     E  X  M  /  P
(R=3)     [A     E] /  M     P
      1    A  -  E  /  M     P

          [A     E     M     P]
```

Fig. 1

This sorting routine is called a bubble-sort. Though there are many steps, each step is clear-cut and conclusive. To write the program, we need to introduce just one new BASIC idea. If you have followed this series so far you will know that there are various ways of giving location names to computer stores. For numbers we have already met A, B, C, D or A1, A2, A3, A4. Instead we now propose to use A(1), A(2), A(3), A(4). This suits our purpose, to make comparisons within a whole set of 'consecutive' stores. We can modify the location name merely by altering the number N in A(N) from 1 to 4. This idea of 'store modification' is extremely useful and powerful. A(N) is called a *subscripted variable* with subscript N.

These alternatives are illustrated in Fig. 2. Lines 10, 110, and 210 are similar. But lines 310–330 perform exactly the same function as line 210. This is a far more economical way of handling lots of data.

```
10 READ  A   ,B   ,C   ,D
20 REM
110 READ A1  ,A2  ,A3  ,A4
120 REM
210 READ A(1),A(2),A(3),A(4)
220 REM
310 FOR N=1 TO 4
320 READ A(N)
330 NEXT N
```

Fig. 2

The sorting of a set of elements into a given order can be achieved by the comparison of consecutive pairs. When we use subscripted variables we compare A(N) with A(N + 1) for a repeated series of different values of N. Now look at Program 1 (Fig. 3). It prints out the greatest number from a given set: to do this, it only needs one round of a bubble-sort. Lines 410–430 READ the DATA (5, 3, 8, 2) from line 490 into the appropriate locations. The loop 440–470 determines the greatest number and 'drops' it to the end.

```
400 REM *********** PROG 1 ************
410 FOR N=1 TO 4
420 READ A(N)
430 NEXT N
440 FOR N=1 TO 3
450 IF A(N)<=A(N+1) THEN 470
455 A=A(N)
460 A(N)=A(N+1)
465 A(N+1)=A
470 NEXT N
48C PRINT A(4);"is the greatest."
490 DATA 5,3,8,2
500 END
```

Fig. 3 Program 1

The comparison is in line 450: if the numbers compared are already in numerical order (i.e. if $A(N) \leqslant A(N+1)$) then the conditional jump to 470 causes a progression to the next comparison. Notice that there are only *3* comparisons as before. However, if the condition is unfulfilled the contents of the stores are *not* in numerical order and need to be exchanged. We trace through the successive contents of two stores to demonstrate the concept of 'exchange', using A as a 'dummy' store.

Location name	A(3)	A(4)	A
contents at start	8	2	0
450 Question: A(3)<A(4)? Answer: NO			
455 A = A(3)	8	2	8
460 A(3) = A(4)	2	2	8
465 A(4) = A	2	8	8
470 contents at end	2	8	8

This process also works with letters, which are in fact stored inside the computer in numerical form. We have A<B<C ...; just like 1<2<3. Now look at Program 2 (Fig. 4) where 'string' (i.e. alphabetic) subscripted variables A$ (N) are used.

```
600 REM *********** PROG 2 ************
610 FOR N=1 TO 4
620 READ A$(N)
630 NEXT N
640 FOR R=1 TO 3
650 FOR N=1 TO 4-R
660 IF A$(N)<=A$(N+1) THEN 680
670 A$=A$(N): A$(N)=A$(N+1): A$(N+1)=A$
680 NEXT N
690 NEXT R
700 PRINT "Alphabetical order is: "
710 FOR N=1 TO 4
720 PRINT A$(N)
730 NEXT N
740 PRINT
750 DATA M,A,P,E
760 END
```

Fig. 4 Program 2

This has similarities to Program 1: line 660 compares the letters for order and the contracted line 670 (cf. 455–465) exchanges them if necessary. This time however we show how to use the variable R to give *all* the rounds of the routine. We saw above that when R=1, 2 or 3 the number of comparisons N=3, 2 or 1. So N=4-R, which explains line 650. The loop 650–680 is repeated for each round just as in Fig. 1.

Now try sorting a given list of *words*, of varying lengths, into dictionary order. This may appear a much harder task. However, there is no need to look up MID$ (A$,N,1) from *MICRO-SCOPE* 5! The BASIC command language is powerful enough to accept whole words for comparison as easily as single letters. Of course, *internally* it uses letter-by-letter comparison. This is a good example of a 'black box' in programming, whose invisible workings need only concern the committed enthusiast. Comparing Programs 2 and 3, the only modifications are because 7 items need 6 rounds.

```
800 REM *********** PROG 3 ************
810 FOR N=1 TO 7
820 READ A$(N)
830 NEXT N
840 FOR R=1 TO 6
850 FOR N=1 TO 7-R
860 IF A$(N)<=A$(N+1) THEN 880
870 A$=A$(N): A$(N)=A$(N+1): A$(N+1)=A$
880 NEXT N
890 NEXT R
900 PRINT "Alphabetical order is: "
910 FOR N=1 TO 7
920 PRINT A$(N)
930 NEXT N
940 PRINT
950 DATA FISH,SOUP,JAR,PEN,PIG,FIG,SAD
960 END
```

```
RUN
Alphabetical order is:
FIG
FISH
JAR
PEN
PIG
SAD
SOUP
```

Fig. 5

'Bubble-sort' is only one of several sorting routines available 'off-the-peg' for most machines. It is not the most efficient method, but the choice depends on the nature of the problem. The educational point is that computers are good at putting items into order.

Many subjects use familiar words in a specialised way, and computing is no exception. If you think you've finally conquered the jargon of new maths, watch out for this usage. 'SORT' to a programmer means 'put into order'. However, when we 'sort into subsets', the computing word is 'SEARCH' according to a given attribute.

We shall develop further ideas on collating and revising data in future issues.

The changes computers may bring

Bob Campbell and Margaret Wellard
Department of Education, University of York

The media have announced that the microcomputer revolution is under way. Hard-pressed teachers are being encouraged to prepare their pupils for the technological world of tomorrow and to tailor the curriculum to meet new needs. This short communication reports and comments on one outcome of an introductory study of computers carried out by a class of ten- and eleven-year-old pupils in a small rural primary school.

Becoming increasingly aware of the impact of computers and anxious to bring the importance of this to the attention of her pupils, the class teacher planned an introductory course based on the ITV series 'Living in the Future'. Although the series was aimed at a slightly younger age group, the programmes were thought suitable and, indeed, the class found them stimulating and challenging. The programmes were presented at fortnightly intervals during the summer term of 1981 and covered the use of computers in the home, at work, in medicine and in education.

Following each broadcast there was a class discussion to try and assess the likelihood of the claims being made, and to consider how the children's lives might be affected. The class talked about computer programs and, in a limited way, about how a computer operates. The climax of the course was a 'hands on' session with a PET microcomputer during which each member of the class used the computer for game-playing and computer assisted learning. Following a short introduction to programming in BASIC the pupils produced simple programs and ran them successfully on the computer. The study of the topic was consolidated by practical work, discussion and writing, and it is with one such latter activity that the remainder of this paper is concerned.

The pupils were asked to write about the changes that they thought computers may bring. The responses, although varying in detail, highlighted four particular applications which, not surprisingly, reflected the content of the TV programmes. In the health service, computers were seen to be assisting in the early diagnosis of illness and in increasing life span. Homes were envisaged as places in which all the menial tasks were done by computerised robots which also waited upon their human masters. Shops and supermarkets were described as having computerised pricing, billing and stock control systems and even robot assistants. Schools were seen to be places where computers made learning easier and provided teachers with reports on pupil progress, where text books were replaced by computers and paper by discs.

Among the other applications envisaged by this group were computerised aids for the handicapped, computerised cars which avoided traffic jams, a variety of computer games and entertainments and systems to help with homework, speed up office work and predict the outcome of horse races. One result of the predicted increase in life span was seen as the creation of greater pressure to produce more food and to exploit more resources both on earth and on other planets. Pupils felt that computers could help in this and particularly in space exploration and settlement.

Nearly all the pupils indicated that they thought that there were some aspects of their scenarios which they considered to be good and others to be bad. The good features were the provision of better services and the speed and ease of performing work tasks. All but one of the pupils noted a less desirable change taking place – an increase in unemployment resulting from the replacement of human labour by computers and robots. Individuals reported their fears in statements such as 'bosses will get rid of their workers'; 'school leavers could be on the dole for as much as ten years'; 'bosses are exchanging computers for men'; 'only ten people will work in a car factory – the rest will be robots'. Some children went as far as predicting world turmoil, violence and rioting (before the events of last summer!). The one pupil dissenting from this pessimistic view saw computers not only as creating highly-paid employment but also as generating wealth for the country.

While some pupils felt confident that computers would not rule the world, others were not so sure. Only two pupils gave an indication that they knew that computers do not always function as intended. Not one mentioned that they require to be programmed by human operators, even though this had frequently been stressed. There was no doubt in the mind of any pupil that computers will change the world. However, while parallels were drawn between computers and brains and the suggestion made that our future councillors may be advanced computers, there was some doubt cast on the overall power of computers. This was very aptly expressed by the boy who wrote, 'If the electric goes off the computer will not work'.

The picture emerging from the writings of these twenty primary school children may not necessarily reflect any widely-held view of children of this age group. Many of the ideas were taken from the TV series while others undoubtedly reflect parental viewpoints. However, there is sufficient evidence to suggest that the pupils put a great deal of careful thought into their predictions; by no means all the ideas presented in 'Living in the Future' were accepted.

We thus find it very disturbing to have to record that the only unifying feature of their predictions on the changes that computers will bring is that there will be an increase in the ranks of the unemployed. If this view is shared by other children then we wonder for how much longer we can continue to control our pupils and to organise their studies on the basis of the Protestant work ethic which persuades our pupils to work hard in the expectation of obtaining secure employment.

'So what replaced you then?'

MAPE matters

Ron Jones
Chairman, MAPE

The first 'MAPE matters' column recorded the background to the development of MAPE. I can now begin to use the column for its proper function, that of keeping members informed of news from the centre.

As you will have already noticed, we have gone 'posh' in this issue, for Heinemann have taken over production. It might be more pleasant to the eye and look more professional in its finish, but it also means meeting much earlier deadlines – which of course means that some of the news can be out of date by the time the journal reaches you.

I sincerely hope that the 'News from the Centre' approach adopted by this column will be reciprocated by members contributing regularly to a 'Regional Round-up of News'. I believe this to be very important if we are to exploit the unique position of *MICRO-SCOPE* as a means of grass-roots communication between teachers working in the primary sector. This can only develop and flourish if MAPE members are willing to use it.

Even the excellent Newsletters which are beginning to filter from the MEP Regional Information Centres are of course regionally orientated, for that's their function. *MICRO-SCOPE* provides us with our inter-regional, our national voice – let's be sure to use it effectively and efficiently.

Conference '82

This year's Exeter Conference is now a thing of the past – plans are already afoot for the Primary Micro Conference '83 although dates and venue have yet to be decided. Members are earnestly asked to send in their ideas and comments on the contents and shape they would like to see incorporated in Conference '83. Members fortunate enough to be able to attend Conference '82 appear to have gained a great deal from the experience; visitors were struck by the sheer verve and expertise displayed by many members. The educational press praised the development in maturity since the first Conference in 1981. As at that first Conference, the 'fringe' events provided useful pointers towards future developments.

I am certain that MAPE members would wish me to express their grateful thanks to all those folks who helped to make the weekend really successful. Special thanks must go to MAPE's Conference organiser, Roy Garland – the last of the stragglers to leave St Luke's College on that Sunday evening swear that they heard a great sigh of relief – that was Roy!

I am certain he would wish me to include in my thanks all those College members, including the Principal, who through their courtesy, hospitality and thoughtfulness helped to make the Conference such a success. My regret about Exeter was a personal one, for a severe bout of flu prevented my attendance. You can imagine my frustrations – not only was I unable to meet the many persons with whom I had corresponded during the year and renew old acquaintances, but also I was prevented from presenting the Chairman's report – which would have allowed me to pay tribute to the hard work of all members of the Steering Committee. If only . . . I must record my grateful thanks to Brian Weaver, MAPE's Vice-Chairman, for stepping into the breach with so little notice.

Regional development

Reports from the Conference made special mention of the success of some of the regional get-togethers. It will be interesting to see just what emerges from the relationships established. Council has promised to support local initiatives during the next year so that a strong regional structure might develop, corresponding to the MEP regions. It is not intended to establish a regional blue-print – rather it is hoped that regions will find a structure best suited to meet their own needs. Our main efforts will be geared to keeping all channels of communication open – so that as one region develops others can learn from the experiences gained. Let us know what help MAPE can offer in establishing local groups.

Booklets/information guides

The Education Guardian (4.5.82) favourably reviewed the three MAPE guides and information booklets, which should now have reached all our members. If you have not yet had your copies please contact Barry Holmes, our Secretary. They only arrived from the printers a few days before the Conference, so you can imagine the frantic efforts made by Barry and his helpers to get copies out to members.

If readers of this column have ideas which they feel they would like expanded in the form of a small booklet or guide for the benefit of other members, then please let me have details. Council will then try to find sponsors willing to help finance its production. We have MEP and CET to thank for their generous help in launching our introductory guides.

PRINT (PRimary INduction for Teachers)

You will no doubt have heard on the primary grapevine that the DoI intend to extend the Computers in Schools Scheme to the primary sector but when, what and who are still closely guarded secrets. The mail shot from Don Walton and me in Issue 5 of *MICRO-SCOPE* revealed that we have been seconded from our teaching posts with Cambridgeshire to help in the production of MEP's multi-media, distance-learning PRINT pack, which will become part of the DoI scheme once it is launched. The self-tutor Pack will have with it a software package which Don and I hope will contain trend-setting programs which commercial software houses and publishers will follow. This should ensure a steady flow of good quality software to support the introduction of micros in our schools.

We hope that the self-tutored course will be supported by a further six-hour tutored course. An article in a later issue will give further details. It is hoped that the training package will go a long way to solving the massive INSET training problem of introducing a great many of the 100 000 primary school teachers to the new skills required.

Software

From training to another serious problem. I understand the frustrations felt by many colleagues at present waiting for their BBC micro or whichever machine their LEA is willing to support. I also understand the even greater frustrations felt by teachers whose LEAs refuse to give advice at this stage. My advice is to wait patiently – once the DoI scheme is announced, things will begin to flow. In fact already the log-jam is clearing: my own school's order for the Beeb has just been delivered to our Resources Centre. Our only problem is paying for it – especially the colour monitor which we need if we are to exploit the full potential of these 2nd-generation micros.

However, it is not the hardware which will prove the problem in the months to come, it will be finding an adequate supply of educationally sound software to run on the new machines. MAPE recognises this problem and is currently validating an evaluation system devised by members of a sub-committee, Ray Haydon and David Ellingham. Once the system has been proved, and is up and running, MAPE will be in a position to list software based on this system of evaluation (see following article).

MAPE members will at least be able to know the type of program they are purchasing – but in the end of course it will be a personal choice. The key phrase in this development of software is 'educationally sound'. Teachers are the best judges of this, and MAPE keeps reminding teachers of their professionalism and expertise. There is an urgent need to marry the skills of the classroom expert with those of the programmer.

Don't let your ideas gather dust – continue to send them in to Don Walton or myself. We will ensure that those judged to have a broad application will be programmed – not by MAPE (which cannot become involved in the production of software), but by Program Centres, set up under the Microelectronics Education Programme.

Sorry to have wandered around a little – but I hope that you are somewhat wiser about the 'News from the Centre' – now please reciprocate and let me know the news from your area or region. Write to me at 25 Abbots Close, Ramsey, Cambs., PE17 1UZ, and mark the back of the envelope 'MAPE – Regional News'.

Program evaluation

David Ellingham
MAPE

A problem involved in the purchase and use of educational software is the variety in the quality of the educational aims and hence content. Often this is not discovered until the program arrives at school. The software producers, on the other hand, are not always aware of how they can meet the real needs of the primary school.

An attempt is being made to overcome these difficulties by the production of a Program Evaluation Package. This comprises:

(a) a document to be used by program producers and groups of teachers considering the design or evaluation of programs;
(b) a summary sheet which can be used by individual teachers within a school when considering the appropriateness of a program.

It is intended that summary sheets will appear regularly with this magazine in order that schools may build up a reference bank based on consistent general principles.

A check-list for teachers

Anita Straker
Mathematics Adviser for Wiltshire (County Hall, Trowbridge)

Anita Straker has sent us a useful document, 'Computer Drill and Instruction: is this what we want?' It provides a comprehensive set of considerations for teachers evaluating new software, and ideas towards development. These extracts give an indication of the scope and approach.

Many schools have purchased microcomputers and many more will do so in the near future. One of the major uses for the computer in schools will be to run programs which present educational material. The check-list described below is designed for use by advisers and teachers who may be judging the worth of educational computer programs. Such evaluation is comparable to that undertaken for any educational material. The check-list has been compiled from an educational standpoint and can be used by teachers who have virtually no knowledge of computers beyond the simple skills of switching on, loading a program and making it run.

A critical appraisal of programs is necessary partly because, in the case of commercial programs, there may be a considerable financial outlay, but mainly to ensure that programs are used to best advantage and that the time of pupils is not wasted using poor quality material. The number of programs is growing rapidly, and it is likely to be beyond the capacity of individual teachers to look at everything carefully. Ways and means will have to be found of making use of reviews carried out by others so that individual teachers spend time evaluating good material and determining where it fits in with their teaching schemes.

Ideally, every program will be accompanied by sufficient documentation to inform teachers about it, what its purpose is, who it is for, what type of program it is, who should operate it, and so on. Alternatively, or in addition, this kind of information might be contained in the program itself and might appear on the screen when the program is run. As a third possibility, information might be contained in the program in remark statements, which do not appear during a run, but which can be seen when the program is listed. For example:

```
100 REM**THIS PROGRAM IS DESIGNED FOR 4TH YEAR**
110 REM**PRIMARY PUPILS. IT GIVES PRACTICE IN**
120 REM**THE CORRECT SPELLING OF COMMON WORDS**
```

The advantage of putting information in the program in this third way is to make it accessible to teachers without bothering young children with it when they use the program.

If information is documented or contained in the program, it should be remembered that it represents only the intention of the writer of the program, not necessarily what the program will achieve in practice. Not all programmers are skilled teachers and the important judgements about program types and audiences should lie with those directly concerned with education.

Finding the answers to the questions listed below should help you to judge the merits of the program.

1. Is information given about
   – the program's name,
   – the author or publisher, or someone who may be contacted over minor queries,
   – which computer model/s it is designed for,
   – which version of BASIC or other computer language is assumed,
   – how much memory will be required?
2. Is supplementary equipment or material needed? Does use of the program require, for example,
   – a large visual display unit, e.g. a 26" TV screen,
   – a printer, sound-box or disc-drive,
   – particular textbooks or work cards,
   – particular apparatus?
3. What does the entire program package consist of? Does it contain
   – a tape or disc version of the program,
   – a program listing,
   – program notes, including warnings about its limitations,
   – notes for the teacher user, including warnings of parts which may give the pupil user difficulty, and hints or comments on questions in the pupil material,
   – notes for the pupil user, phrased in appropriate language, with adequate illustrations,
   – additional teaching material, including suggestions for extensions or increased flexibility,
   – references to other sources and/or a bibliography?
4. How much time is required
   – for loading the program from tape,
   – at the computer,
   – for the whole package?

Schools will need to catalogue their collection of programs for the teachers who will use them, and perhaps also for their pupils.

Conclusion

To sum up, the check-list should enable you to answer three major questions about any educational computer program:

1. What kind of program is it?
2. Is it in principle suitable for the general teaching scheme of which it would be part?
3. What are its merits and in what ways could it be improved?

It must be emphasised that this check-list has been concerned with only those educational programs which could be used in the classroom. Programs which are aids to administration have not been considered, though some of the questions raised here would still be applicable. Neither has the check-list concerned itself with the style of the computer language in which the program is written, what is going on 'behind the scenes', or whether the program is easily transferable to other machines. These matters are best left to those who are experts at such things. The list, however, may help those teachers who have ideas for further programs draw up a better brief for those who will produce them.

Coping with commercial software

Dr David Wharry (Department of Education) and
Dr Michael Thorne (Department of Computing Mathematics),
University College, Cardiff.

Over the last year, 30 primary schools in South Wales took part in a project financed in part by the Department of Industry. Each school was loaned, for three weeks, a 32K PET cassette system and 32 *commercially* available programs. The main aim of this work was to see how primary school teachers new to computing would cope with 'off the shelf' software of various kinds. Was the documentation the software suppliers provided adequate for non-programmer teachers? Were the programs robust enough for everyday use? These are but two examples of the questions to which we sought answers. A second aim was to see how the pupils got on with both the computer and the programs. A third was to increase the exposure of young people and their teachers to microelectronic technology.

The choice of the Commodore PET was to a certain extent Hobson's, as Commodore was the only popular microcomputer manufacturer which would assist financially. It was also very appropriate at the time the project began since then most of the *commercially* available educational software was for the PET. This situation has changed dramatically over the last year. However, the results which have emerged from this work will, we hope, have implications for commercial software from any source.

The 32 programs loaned to each school were accompanied by the suppliers' documentation, unedited in any way. A three-page leaflet which we had written explained the procedure for loading a program from a cassette tape and, except in one case, at least one teacher was taken through the tape-loading sequence when the machine and programs were delivered to the school. Often the pupils sat in on this demonstration at the teachers' request, as the children would grasp it quicker and could explain to the teachers when we had gone! Just in case, we left our telephone numbers, but only three schools ever made use of this facility. One of these calls was not due to a technical difficulty but merely reported a fault with the cassette recorder.

As this was the only 'mechanical' breakdown during the year, the PET impressed us with its reliability. Our machines were driven in the boot of a car for an average of 40 miles every three weeks and were used extensively at each of the participating schools. Over 2000 children and 80 staff had hands-on use of the machines. In school the greatest usage was during the afternoons, traditionally a time for the more arts-based subjects of the primary curriculum. Over two-thirds of the schools allowed their children access to the machine during break times. The teachers themselves seemed willing to give up their own time as well: 84% of those involved took the machine home at some point during the three weeks.

The programs

These were assembled from a variety of sources and covered a very wide range of ability and subject matter. Six were maths drill and practice programs; four more were concerned with other numerical work. Another main group contained five programs concerned with language development, the remainder being games and simulations ranging from Microchess (very popular in the school breaks) to MAXIT which one of us thought had great potential for mathematics teaching. Four of the simulation programs were much too hard for the majority of primary school children and consequently saw little classroom use.

Some suppliers offered several programs on one tape. 55% of the teachers in our schools felt this was a bad idea, presumably because of problems with loading from tape. Only one of the programs which involved multiple tape loads made the teachers' Top Ten programs. Moreover, every one of the 36 reports of difficulties with a particular piece of software concerned such programs.

The information the programs displayed on the screen was found to be helpful to the children in 88% of the cases and no one reported it to be 'meaningless or irrelevant'. Looking at some of the programs, this surprised us a little – as did the fact that about two-thirds of the teachers indicated that they felt enough opportunity was given for pupils to correct their answers (after a typing error, for example). Several of the programs gave no such opportunity at all!

About two-thirds of the teachers thought the error messages displayed were helpful and 13% found them confusing. However, according to their observations only half of the children found them helpful. This was not surprising: one of the programs offered the cryptic message 'REPOSITION TAPE AHEAD OF FILE' if you went wrong at a certain point in its use.

Most (80%) of the teachers were satisfied with the written documentation – where it was supplied – and no one made any specific suggestions for its improvement. When asked which facilities on the computer they would change given the opportunity, a few teachers expressed dissatisfaction with the cassette tape facility. There were some complaints about the positioning of the RUN/STOP key next to the RETURN key. This particular topic came up again under 'difficulties the children had with the keyboard'. Generally speaking these were few and far between – seven complaints in total. Most of them concerned the confusion which occurs if you run an old-ROM program on a new-ROM PET. (Remember, we ordered the programs for the machines which were used and passed what was supplied to the schools).

Just over half felt that one machine was adequate for a class; the rest suggested ratios of between 1:2 and 1:10 machine/pupils. Everyone felt that the effort needed to learn to use the machines had been worthwhile though only 50% would, on the evidence of the programs they had seen, have considered it worthwhile to buy a computer assuming the necessary funds were available. Nevertheless, very nearly everyone said that they would wish to try another make of machine and/or different programs if the chance arose.

The introduction of a computer into the school caused little or no disruption to normal teaching activity in the vast majority of cases. Comments received seemed to support the authors' view that in many ways the primary school curriculum is more suited than the secondary to the introduction of computers. Even so, our teachers worked hard: the average estimated amount of teachers' time involved in looking after the machines came to one hour per day.

Some conclusions

Getting an evaluation of educational computer programs from teachers generally unfamiliar with the technology is difficult. By almost any criteria *some* of the programs were poor. Yet when the dreaded SYNTAX ERROR message appeared owing to a program fault, discussions indicated that the teachers tended to blame themselves rather than the program for what had happened. This may account for the surprisingly low number of complaints about the software. Similarly, lack of objection to cassette tape systems cannot be taken as an endorsement of their classroom success. Very few of the teachers involved had any prior experience with computing at all. Consequently, as far as they were concerned, you simply had to have cassettes because other systems didn't exist or were not appropriate, so there would be little point in challenging this part of the equipment.

Because of this disinclination on the teachers' part to put the blame where it truly lay, answering even our two basic questions (Was the documentation adequate? Were the programs robust enough?) is not possible from the work so far. By the relative forcefulness of the comments, it would appear that having many programs on one tape is not a good idea for teachers as inexperienced in computing as those who took part in this project. It can also be said that the primary school teachers do rapidly move from a position of fear regarding the machine to one of acceptance. Indeed it was the very enthusiasm of the teachers for the programs which has obscured any conclusions we might otherwise have drawn.

Since, in addition, all the children were keen – according to their teachers all the pupils involved in the project were more confident with the machine after using the programs than before – primary schools would seem to offer especially fertile land in which to sow the seeds of CAL. The resources currently available to the Microelectronics Education Programme will no doubt necessitate an emphasis away from the primary schools at the present time – until, say, the DoI offers to primary schools financial aid for machine purchase similar to that presently available to secondary schools. All this seems a pity when the style of curriculum is highly appropriate and the teachers and pupils are very keen.

Primary Pilot Project—ILEA

Angela Hirst
ILEA Educational Computing Centre, Bethwin Road, London SE5

Introduction

In September the Inner London Education Authority launched a pilot project in the use of computers in primary schools. Eleven schools were selected as pilot schools and were almost fully funded in the purchase of Research Machines 380Z microcomputers with twin floppy discs, high-resolution graphics, RGB colour and an Epson printer. Some of the schools selected had a little previous experience in using computers.

The schools' equipment arrived between September and November 1981. At this time the first course for teachers in the pilot schools was under way.

Although the idea of the pilot scheme is to provide information prior to other schools going ahead, the very nature of life at the moment means that the pilot is running concurrently with other activities. Six schools had by September ordered virtually the same equipment (without colour or printer) and were taken on board with the pilot scheme.

Since the project started we have picked up five more schools who have ordered equivalent equipment. A series of four one-day sessions was provided to cater for these schools, who have also been invited to attend all current INSET for the original schools.

INSET

INSET during the first term consisted of ten Friday courses, the first five days being for one teacher from the pilots (both centrally- and self-funded schools) and the following five days for another teacher from the school. These days had the aim of teaching the course members to press the right buttons to perform certain tasks, to get familiarity with handling the equipment.

A set of 15 discs was given to the schools. These contained software at varying stages of development (mostly version Ø!) which had been said to be appropriate to children of primary school age. Part of the role of the pilot group is to specify which, if any, of the programs are worth developing. At this stage of the pilot project it was expected that much of the teachers' efforts would be directed towards their becoming familiar with the computer and material.

In January 82, following the two series of five Fridays, a three-and-a-half day residential course was attended by all the teachers involved in the project. The emphasis here was on the use of the computer with primary school children. Lectures were given on a particular program (or package) by a teacher who had used that program. In two cases children from a local primary school were brought in for the session, so that the course members could see how the computer could be used with children. The children had not used a computer before.

Until this stage of the pilot project it was not expected that the computer would be used with the children. Following the course, most members used the computer with small groups of children, outside normal lessons, to try out an idea on the use of a program. In all cases the program fitted in with work the children were currently doing.

In the latter half of the Spring term, the teachers on the pilot scheme started integrating the use of the computer into their normal lessons.

Further INSET to follow the residential course was set up to satisfy the needs of the pilots. Each course member was asked to indicate what INSET he required. Whole-day sessions were requested, for looking at a particular program or package. The format of these days was a practical session in the morning followed by a discussion on the ways it had been or could be used. A teacher with experience in using the program was invited to lead the discussion and pass on his experiences. Other members then described successes or failures they had had, and ideas they wished to try out. These groups will continue to meet in this way.

The other form of INSET required was a series of after-school meetings, perhaps two a term, where the pilots could bring any problems to the computer centre, and hopefully get solutions!

The other major form of support is via the telephone. As programmer in support of primary developments, I am available at most times to answer queries.

Programming support

A few members asked for programming courses. We were unable to arrange these for the Spring term, but will suggest courses they can attend in the near future if they still feel they wish to.

Most of the teachers will not want to program. They had a session on *reading* BASIC programs so that cosmetic changes could be made. When any of the pilots have an idea for programs, they specify what they want, how it should appear on the screen, required responses etc., and I do the programming, working closely with them.

Summary

The pilot scheme is still in its infancy. There is a whole-day conference in May for all members of the scheme to attend, at which many aspects of the project will be discussed. It should be possible to make some indication of the direction in which our schools are moving following the conference. At the moment feedback (mainly in the form of the type of INSET requested and the comments during the sessions) has indicated that the major areas of interest are in large, flexible software packages. The day sessions have been on word processing (TXED), information retrieval (LEEP), classification of information (SEEK), and LOGO.

ILEA's primary and computing Inspectorate are contributing to the running of this scheme, and the experience gained from the involvement of the pilot schools will be invaluable in extending the use of computers to other ILEA primary schools in the future.

Newman College Schools Project

Helen Smith
Project Co-ordinator, Newman College

The project got under way in the six schools just before Christmas. Each was equipped with a Research Machines Ltd 480Z (32K), a 16″ or 19″ B/W monitor, and a cassette recorder. They have just taken delivery of the high-resolution graphics boards.

Each school had to make important decisions about the siting and timetabling of the machine. The size and layout of the school have been crucial factors. In large schools it is not easy, with the single machine currently available, to facilitate a continuous micro-based scheme of work. Most of the schools are built on different levels, and moving the hardware around the school has proved difficult. Two of the six schools are, however, built on a single ground floor level and it has proved convenient to mount the micro and monitor on a trolley and wheel it around the school. At another school, staff have opted for the rigours of bodily transporting the equipment from one classroom to another each morning. Other schools have decided to site the hardware in one area. This has the advantage of enabling the area to be built up as a computer resources library. But it tends to limit the use of the micro as a teaching aid with large groups of children. Although some teachers are prepared to take groups or even the whole class along to the computer room, the micro is being used most of the time by two or three children only.

These factors also influence the teachers' choices of suitable software. There is obviously a need for programs that can be used by children working independently. Very young children or remedial groups have difficulty in selecting options from too wide a range of possibilities. Wordy instructions are also best avoided. The use of the GET command, whereby the computer waits for a specified character input, is very useful in minimising the errors that an uncertain child may make. In one school, programs have been copied on cassettes, each with data suitable for a particular level. The cassettes are colour-coded according to year groups, so that it is much easier for children to choose and load the appropriate program.

Where children are using the micro in a room on their own, there is inevitably a problem in assessing their progress. Two schools are fortunate in having a printer, which enables children to take a hard copy of their work back to class. Teachers are then able to check on the children's work. Printouts kept in folders provide a record of work. Children are, of course, often very proud of their printed results. In creative applications, it is often very useful to have printed output: one teacher has written a program which enables children to write their own data for randomised limericks. A different version of the limerick is printed at each run.

Some teachers are aware, however, of the drawbacks of using the printer routinely. Many children seem to build up a private relationship with the micro. It does not criticise them if they make a mistake. They are allowed opportunities to correct their work, leaving no trace of the original error. Over-use of the printer in a continuously testing situation would appear to undermine the trust the children have developed in the machine.

In many cases, the real signs of progress are not apparent until some later occasion, in class. It is often in odd, unexpected moments that we become aware of the progress children have made. I have come across several accounts of children being able to tackle something they had not been prepared for in class; it appears that they have learnt the necessary skill in their sessions with the micro.

I hope that more teachers will take the opportunity, where organisation permits, to use the micro as an aid in class, whether with groups or with the whole class. An important aspect of the project is to track down and distribute quality software to our schools. The more attractive the software and the easier it is to use, the greater chance there is of getting *all* our teachers involved in using the micro themselves, with their children. *Eureka*, from ITMA, is an excellent example. It inspires a high level of interest and motivation, and it is very easy for a teacher to select the self-explanatory options. The program shows the changes of water-level in a bath, whether the plug is in or out, taps on or off, or the man is in or out, in pictorial and graph form. It relates to a number of primary maths and science topics, and stimulates a variety of work to be done away from the computer.

As well as developing the use of such programs in schools, I intend to use our own information retrieval program, *P-QUERY*, to develop its possibilities as a database. As a college, we are also in a position to lend specialist support to schools interested in developing a particular aspect of the curriculum. We already have a series of excellent geography programs developed by college staff. These have attracted a great deal of interest from the Project's schools, who will now be able to use them with the long-awaited high-resolution option boards. Several schools place a high priority on remedial language support and compensatory language development, and are interested in developing programs to assist in these areas. Some teachers have indeed had success in writing their own software, specifically tailored to their children's needs. The stage is now set for future developments, both in organisation within the schools and in the long-term evaluation of the effectiveness of software. We shall be keeping you informed of these developments in future issues of *MICRO-SCOPE.*

Second thoughts on the threshold

L. McGinty

child Mum, we had a super day at school today.
mother Yes, darling, and what did you do?
child Well, we had a play on that machine that the headmaster's always taking home in his car.
mother Oh yes, and what did you play, dear?
child Well, there's this monster, see, and it's trying to eat you up.
mother Not a real monster, surely, darling?
child No, it's on the TV screen. You see, what you have to do is answer these questions and if you get them wrong the monster comes out and eats you up and then he opens his mouth and the word B-U-R-P comes out.
mother That sounds fascinating. What sort of questions did you answer?
child Times mostly, I think. Oh, and I think there were some adds, too.
mother And what did you *learn* from this machine?
child That monsters burp after their dinner.

An imaginary scenario perhaps, but the point which I am trying to make here is nevertheless valid as we stand on the threshold of the microchip revolution (or are we already over it?) The microcomputer is seen increasingly as a valuable aid to teaching in the modern primary school, but before we all go rushing off to our PTAs with our begging bowls, asking for funds to furnish our schools with one or two or three or more of these revolutionary machines, let us pause and think.

I work in a 5 to 9 first school serving an area of almost completely private housing, with children drawn from a highly articulate and professional background. Accepting that I am bound to have a large proportion of more able children, how do I evaluate the need for a microcomputer (or two) in my school?

There are two criteria – (a) the cost and (b) the benefit which *each* child would receive in educational terms from the deployment of such a machine.

Let us first consider the cost. When educational resources are rapidly dwindling and essentials like reading books are having to last longer and longer, any investment which involves an outlay of approximately £500 requires careful scrutiny. Staff are frequently crying out for equipment. The need to provide an interesting and stimulating environment for all children, including the very young, may not necessarily be found in the provision of a microcomputer. Here is a list of priorities which any teacher might consider:

> Library books,
> Science equipment,
> Mathematical apparatus,
> Carpeting for classrooms,
> Adequate display space,
> Shelving for class libraries.

Many teachers might list their own priorities based upon their own particular needs. I know my own school well enough to realise that the outlay of large sums of money could be employed in any of these areas before I would even consider a microcomputer.

The second criterion which I would employ is a far broader one than mere cost. Quite simply, in a school of the type in which I work I cannot see how a microcomputer or two could bring anything other than marginal benefits.

There are two possible uses for a microcomputer in school. The first is for children to prepare their own programs using flow charts, and gain some first-hand experience and knowledge of computer logic as well as the ability to use BASIC.

This may be acceptable in a secondary school but in the first school I am more concerned with basic numeracy and the ability to use mathematics in everyday situations. Despite the proliferation of computers and their attendant software over the last few years I still do not feel that even at the age of nine there is any great need for my children to learn computer programming.

There might be one or two extremely able children who might benefit but this brings me back to the first point regarding cost. It hardly seems justifiable to spend a vast sum of money on one or two children. What about the other two hundred?

The other use for microcomputers is, I feel, even less justifiable. This is their use as an educational aid with prepared programs for teaching say geography or mathematics. The possibilities are, I am assured, boundless. The trouble is that only one person can use a computer at a time, or perhaps two – at a pinch. So here we have all this outlay in expense and teacher time (for instance in the preparation of programs) so that one or two children can use a machine. Perhaps some of the work could be done in larger groups but this is likely to produce frustration. When I have used microcomputers I have found working even in pairs highly unsatisfactory. Ah, but am I being really fair? The children are learning – aren't they? Unfortunately, I'm not so sure. I have seen no evidence of the effectiveness of microcomputers as teaching aids – only bland assertions that we are in the twentieth century and that we must move with the times.

I have yet to be convinced that the child's experience is of any educational value. You may think that my introduction was rather facetious, but the point that I am trying to make is that quite often the child will remember the joke but not the teaching point behind it. Many of the educational programs are merely up-dated versions of games like Hangman or Battleships – games which could easily be played on paper. Perhaps while the medium still has novelty the child may be more highly motivated but it is quite possible that microcomputers will become as mundane and sterile as any ordinary textbook.

It may appear that I am totally opposed to the use of microcomputers in our schools, but this is not so. At a secondary level children could benefit considerably from the use of computers provided they were learning the processes of how to devise programs, as opposed to playing games on ready-prepared programs. This latter approach I feel is a blind alley and I believe that history will prove me right.

As for my own school, I would prefer to continue to equip it with interesting books, paper, paints and the more essential educational equipment appropriate to children of this age range. I know many people who live in badly furnished, badly decorated houses and yet they run brand new shiny motor cars. Many primary schools which have microcomputers are in a similar predicament. When our first schools are fully equipped and furnished with all the conventional necessities then I would consider microcomputers as a possible option, but until then I shall happily return to my ivory tower until somebody is able to convince me otherwise.

Bits and pieces

Newman College software

The library of available software (based on the 380Z and 480Z) continues to expand – and improve. A third disc is now available containing:

DIAGH2 SPELL3 GRIDREF2	these are rewrites of existing software
EUROPE N.AMERICA S.AMERICA AFRICA	similar to ENGLAND, but covering the areas indicated
PQUERY PEDIT	a demonstration program for teaching about information retrieval
DIRECT	practice in direction finding in geography
COGS	simulates the interaction of cogs of different sizes – mathematical or scientific; dependent upon support material being written
CODE	a code-cracking program using deductive logic
SELECT	a spelling program concentrating on letter reversals (e.g., b and d)
OPTAB	another variation on the tables theme

Before requesting these programs, please write in for the Software Summary Sheet which will explain what is available in far more detail and how to acquire the programs.

Useful addresses

MAPE (Micros and Primary Education)
Mr B. Holmes, St Helen's C.P. School, Bluntisham, Cambs. Annual membership £7.50 – strongly recommended to all schools who wish to keep up to date with advances and developments in microcomputer technology.

Machines
The past six months have seen the emergence of three very good British machines suitable for the primary sector. Further details of these can be obtained as follows:

RML 480Z: Research Machines Ltd, P.O. Box 75, Oxford.
BBC/Acorn: Acorn Ltd, Fulbourn Road, Cherry Hinton, Cambridge, CB1 4JN.
ZX Spectrum: Sinclair Research Ltd, 6 Kings Parade, Cambridge, CB21 1SN.

We would be delighted to receive comments on any of these three machines from readers who already have experience with any of them.

BBC Guidelines for Software Writers
Free copies are available from: BBC Educational Publications, 35 Marylebone High Street,

MICRO-SCOPE back numbers
We are rapidly running short of back copies of *MICRO-SCOPE* issues 1 to 4. It is therefore proposed to publish a compendium of the best articles from these issues, and hopefully this will be available from Heinemann/Ginn at the beginning of September.

Courses

Diploma in Computer Applications to Education, 5 – 13 years.
This course is intended for teachers in primary and middle schools who have an interest in the use of the microcomputer as a teaching aid in the primary curriculum. The course is one year, full time, with residence available on the campus. Applications are now being considered for entry in September 1983. Details available from: The Registrar, Newman College, Bartley Green, Birmingham, B32 3NT.

Users Groups

A 480Z Users Group based in the West Midlands has just been established in order to exchange software and experience. Further details available from: Dave Breedon, Parkside Primary School, Ballot Street, Smethwick, West Midlands.

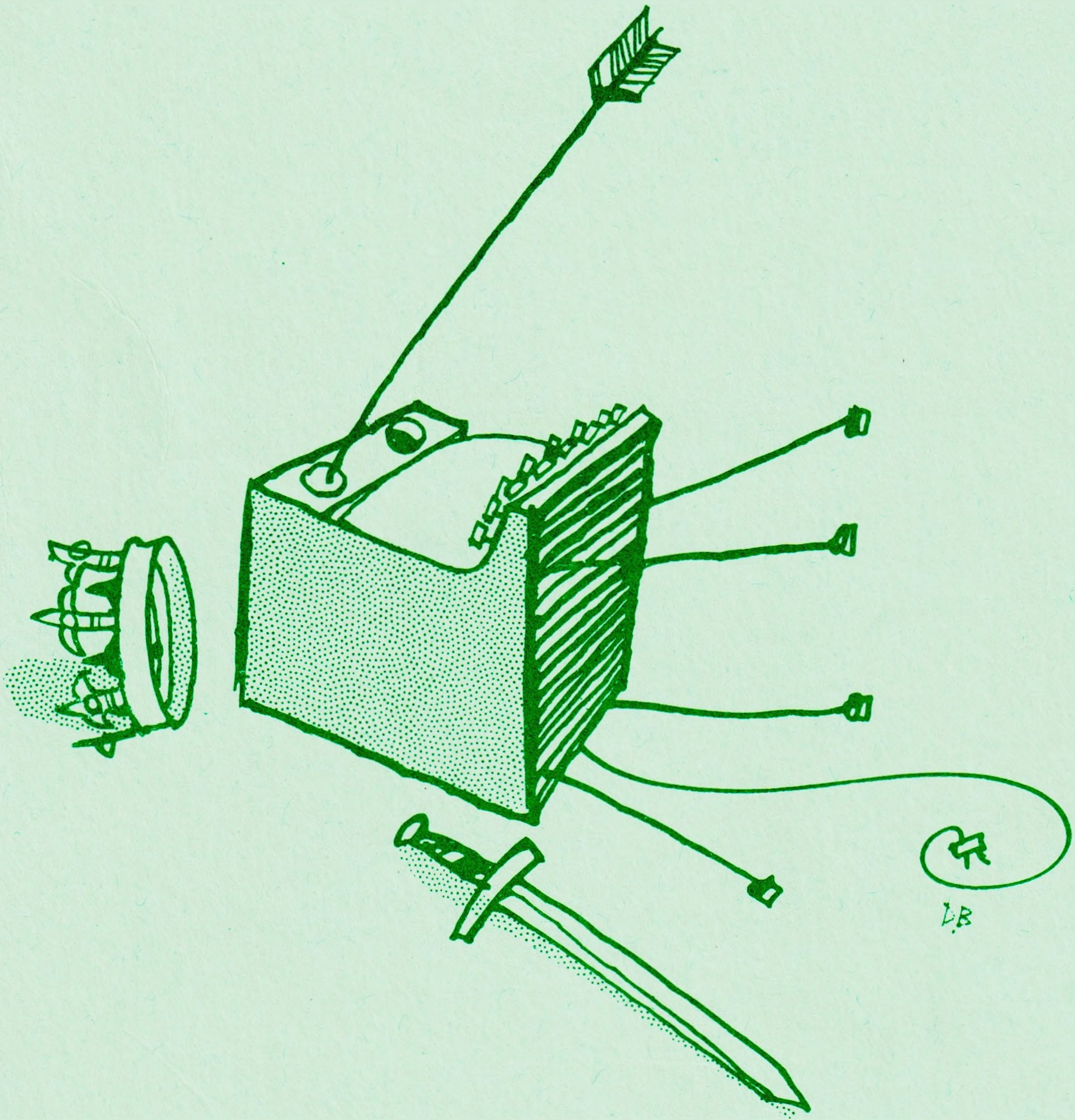
We would be pleased to hear from other such groups. Two that we know of who are interested in the primary sector are:

TRS-80: Dave Futcher, Beaconsfield First and Middle School, Southall, Middlesex.
Sinclair ZX81: Eric Deeson, Highgate School, Balsall Heath Road, Birmingham, B12 9DS.

MAPE Annual Conference 1983

Make a note of the dates now!
April 8, 9 and 10, at the University of Loughborough.

Roger Keeling



Published by **Heinemann Computers in Education Ltd**
in partnership with Ginn and Company Ltd
22 Bedford Square, London WC1B 3HH
ISBN 0 602 22613 9