fun school 3

Instructions and Parent/Teacher Guide

- Atari ST
- Amiga
- PC
- Commodore 64
- Amstrad CPC/PCW
- Spectrum
- BBC micro

for the Over 7s

fun school 3

Welcome to the Fun School 3 package for children of over seven years of age. Following the outstanding success of Fun School 2, we are proud to present six new programs which will delight, entertain and, most of all, educate your children in a wide range of skills using guidelines laid down by the National Curriculum.

Fun School 3 is all about learning, so until children are sure of a program's objectives, it is important that they receive adult supervision whenever a new program is loaded. This will greatly reduce the possibility of frustration at a later date. In fact, you will be surprised by the speed at which they will understand the ideas behind each program.

Each program in this Fun School 3 package deals with a different aspect of the education of the over sevens. With a group of programs as diverse as this, it is inevitable that everyone approaches its goal in a different way.

Over 7s

This package contains six programs, four of which are educational games:

Planetary Maths presents a vast array of mathematical problems.

Sentences considers a child's spelling, grammatical and punctuation skills.

Word Search continues with the theme of spelling in an interesting and novel way.

Treasure Search helps to develop an appreciation of distances and directions.

They each feature a wide range of impressive sound effects which not only makes them more fun to play, but also serves to distinguish between right and wrong answers. They begin at the lowest level of difficulty and will progress to higher levels in response to the child's achievement. In addition to this, the level of play can be altered by the user.

The remaining two programs are important educational utilities:

Database provides children with an opportunity to examine and use one of the computer's most important work tools. This program is unique as it comes with a question and answer facility.

Robot Draw contains many of the principles in the popular educational programming language Logo.

For each of these programs, this manual offers a comprehensive tutorial. This should be followed by the children before they attempt to produce work of their own.

It is important that young users are taught the ground rules at an early stage:

- Disc drives are for discs, sockets are for plugs. Neither of these were designed to take little fingers, Ghost Busters' proton packs or the like.

- Soft drinks, sticky buns and so on are out!

Children of any one age group can have widely differing abilities. Fun School 3 has been designed to cater for as wide an ability range as possible. On lower levels some programs can be operated quite comfortably by children under eight, while others may still prove taxing to a 10 or 11-year-old, especially at higher levels.

With four of these programs you, or your child, will have the opportunity to produce and use your own data. If you wish to use this at a later date you must save it to tape or disc. It is strongly recommended that before using Fun School 3, disc users format a blank disc and tape users have a blank tape, ready to accept data. If you attempt to save onto a Fun School 3 program disc or tape, you may lose existing files.

This manual gives simple instructions for each program. These are intended to be read by children with help from parents or teachers if necessary. Towards the end are separate parent/teacher notes which serve to highlight each program's objectives as well as make suggestions for adult involvement in the learning process.

All the Fun School 3 programs can be loaded easily from the menu screen using the **Spacebar** and **Return** key, thus eliminating the need for typed commands. Leaving any of the programs and returning to the pictorial screen is achieved by pressing the **Menu** key.

This easy-access method of program selection will allow children to switch between games without adult assistance, playing each for as long as their concentration span allows. You will probably find that the children's favourite choice of program changes over a period of time as they acquire new skills.

the games keys

The key lists below show the controls for each game on all the computer formats that run Fun School 3.
To *select* an option *move* to that option and use the *confirm* key.

Amiga & ST *Change levels:* F10 *Quit:* Esc

Wordsearch, Robot Draw, Planet Maths and Database
Confirm: Left mouse button *Move:* Mouse.
Treasure Search
Confirm: Left mouse button *Move:* Mouse. *Options:* F10
Sentences
Confirm: Left mouse button *Move:* Mouse. *Go to file loading screen:* L
Go to options screen: O

Spectrum *Change levels:* Symb 0 *Quit:* Symb Q

Wordsearch, Robot Draw, Database
Confirm: Enter *Move:* Cursor keys
Planet Maths
Confirm: Enter *Move:* Space cycles through options
Treasure Search
Confirm: Enter *Move:* Cursor keys *Options:* Symb Q
Sentences
Confirm: Enter *Move:* Space cycles through words. *Enters your answer (**OK**):* K
*Asks for a **c**lue:* C *Load:* L *Options:* O

CPC Change levels: Keypad 9/F9 Quit: Esc+Esc

Wordsearch
Confirm: Return *Move:* Arrows Space cycles through options
Robot Draw, Planet Maths and Database
Confirm: Return *Move:* Arrows
Treasure Search
Confirm: Return *Move:* Arrows *Options:* O
Sentences
Confirm: Return *Move:* Arrows *Load:* L *Options:* O

PCW *Change levels:* Paste *Quit:* Alt+Stop

Wordsearch
Confirm: Space *Move:* 8 keys surrounding numeric key 2
Move between word grid and options: Exit
Robot Draw
Confirm: Return *Move:* Cursor keys
Database
Confirm: Return *Move:* Cursor keys *Move between menus:* Exit
Planet Maths
Confirm: Return *Move:* Cursor keys *Move between menus:* Exit
Move force field: Cut
Treasure Search
Confirm: Return *Move:* 8 keys surrounding numeric key 2
Re-position cursor: Exit *Options:* Paste
Sentences
Confirm: Return *Move:* Space cycles through words *Move between areas:* Exit
Load: CUT *Options:* O

PC *Change levels:* F1 *Quit:* Escape

Wordsearch
Confirm: Return *Move:* Arrows *Space cycles through options*
Robot Draw, Planet Maths and Database
Confirm: Return *Move:* Arrows
Treasure Search
Confirm: Return *Move:* Arrows *Options:* O
Sentences
Confirm: Return *Move:* Arrows *Load:* L *Options:* O

BBC *Change levels:* F9 *Quit:* Esc

Wordsearch,Robot Draw, Planet Maths and Database
Confirm: Return *Move:* Arrows
Treasure Search
Confirm: Return *Move:* Arrows *Options:* O
Sentences
Confirm: Return *Move:* Arrows *Load:* L *Options:* O

Wordsearch
Confirm: Space *Move:* Left (Z), right (X), up (;), down (?)
Planet Maths and Database
Confirm: Space *Move:* Left (Z), right (X), up (;), down (?)
Robot Draw
Confirm: Space *Move:* Left (Z), right (X), up (F5), down (F7)
Treasure Search
Confirm: Space *Move:* Left (Z), right (X), up (;), down (?) *Options:* O
Sentences
Confirm: Space *Move:* Left (Z), right (X), up (;), down (?) *Load:* L *Options:* O

the programs

Word Search

Word Search is a game that is similar to the word square puzzles you may have seen in comics and magazines. When the game has loaded you will see Robbie, an empty word grid and a list of words on the screen. These are the words that are going to be used in the game.

To begin, **select** *START*. Robbie will then make a grid of letters in which the words will be hidden. Once he has done this (it takes a few seconds) the grid appears on the screen and you are ready to go.

All the words from the list are hidden horizontally and vertically somewhere in the grid. All that you have to do is to find them! When you think you have found a word, **select** its first letter. Then **select** the last letter and the whole word will be highlighted.

If you think you have made a mistake with this word, **select** *CANCEL*. This makes the highlight disappear and lets you have another go.

If you are happy with the word you have chosen, **select** the same word in the list and Robbie will tell you whether you are right or wrong.

After correctly finding a word, a little blob appears beside it in the word list to help remind you which words you have found and which you have yet to find.

If you can't see a word in the grid, you can get some help. **Select** *CLUE* and then the word you are looking for in the word list. In the grid, a letter in this word will be highlighted for a few seconds showing you where to start your search.

You can make the game more difficult. There are three levels of play and you can change between these by pressing the **Levels** key. The level of play is the number highlighted at the top of the word grid. In level 2, words are placed diagonally as well as horizontally and vertically. Level 3 also includes words in reverse.

At any time you can change the words that make up the grid, by selecting *NEW*.

Robot Draw

At first glance, Robot Draw may appear to be just another drawing program. You can use it to produce pictures, but there the similarity to any other drawing program ends. It is, in fact, a sophisticated and powerful version of the programming language called Logo. Each line, movement or colour used in the pictures produced by Robot Draw represents a line of programming. By using this version of Logo, not only will you be able to produce amazing pieces of graphic art, but you will also be learning how computer programs are constructed.

In the middle of the screen is a turtle, but not of the Teenage Mutant Hero variety. This turtle is a small arrow-head shape that you are going to control using a number of commands. You don't have to remember these as they are all displayed on a panel down the side of the screen.

To use a command just **select** it using the pointer. If the command needs a number you will then be asked to type it into the computer. These numbers tell the turtle how far to go, how much to turn and how fast to go.

The commands on the panel are:

COLOUR When you **select** this the colour menu appears at the top of the screen. This lets you choose the colour (shade on the PCW) of the line and fill that the turtle will use. On some machines, you are not limited to the default colours: By selecting the little arrows above and below the numbers you can modify the current colour. On some versions where many colours can be used on screen at the same time, a *Rainbow* option is supplied. If you **select** the *RAINBOW* option each new line will be a different colour.
This menu may also include an *UNDO* box. Selecting this will reset the program to use the previous colour set up.

FILL Fills a shape with the currently selected colour (shade). Once you have drawn a shape, such as a circle or a triangle, just move the turtle into the middle of it and **select** *FILL*. The shape that you are trying to fill must have no gaps in it or the colour will leak out and fill the whole screen.

PEN UP/DOWN Toggles between the two options. When you select either one, the panel ***displays the other ready to be selected***. When you select *PEN UP* you can move the turtle without drawing on the screen, whereas *PEN DOWN* produces the lines.

HIDE/SHOW Changes between the two modes to let you choose whether the turtle is shown or not. Hide is normally used when you have completed a drawing and don't want the turtle to spoil the view.

SPEED Tells the turtle how fast to move. You can make it move slowly when you want to see what it is doing or fast so you don't have to wait a long time while it draws a complicated picture. When you choose this option you are asked for a number between 1 (fastest) and 50 (slowest).

EDIT This allows you to look at the effect that each command in a sequence has upon a drawing. It also lets you make changes to any part of the drawing without having to start again. This command is dealt with in detail under *Editing Instructions*.

RECORD This is used to place instructions into the memory of the computer. Details on how to use it are given under *Recording your commands*.

DISC/TAPE This is used to save or load pictures and is explained under *Saving Instructions* and *Loading Instructions*.

FORWARD This moves the turtle in the direction it is pointing. After selecting this, you are asked how far to move the turtle. The number you can enter depends upon how close to the edge of the screen you are. If you enter a number and press **Return**, or **select** the *OK* box, and nothing happens, you are probably too close to the edge. Enter a smaller number and try again.

BACKWARD Works in a similar way to the *FORWARD* command, but moves the turtle in the opposite direction to the one in which it is facing.

LEFT Turns the turtle to the left. The program asks you for a number of degrees through which you want the turtle to turn. You can enter any number between 0 and 360.

RIGHT Works in a similar way to the *LEFT* command, but turns the turtle to the right.

CLEAR Wipes your drawing from the screen. A confirmation box appears so you have the chance to change your mind. If you really want to clear the screen **select** *OK* but if you selected *CLEAR* by mistake, **select** *Cancel*.

On the following pages you will find a number of suggestions that will help you to understand how to use Robot Draw. If you follow each of these in order, trying them out for yourself, you will soon be able to design and produce your own pictures.

First enter the following series of instructions. These will produce a square in the middle of the screen:

1. **Select** *CLEAR* then *OK* to clear the screen and move the turtle to the centre.
2. **Select** *FORWARD* and enter **50** units.
3. **Select** *LEFT* and enter **90** degrees.
4. **Select** *FORWARD* and enter **50** units.
5. **Select** *LEFT* and enter **90** degrees.
6. **Select** *FORWARD* and enter **50** units.
7. **Select** *LEFT* and enter **90** degrees.
8. **Select** *FORWARD* and enter **50** units.
9. **Select** *LEFT* and enter **90** degrees.

You should now be looking at a square near the middle of the screen with the turtle facing in the same direction as when you started. To go one stage further and colour the square, enter the next five instructions:

10. **Select** *LEFT* and enter **45** degrees to point the turtle towards the middle of the square.
11. **Select** *PEN UP* to stop the turtle from drawing.
12. **Select** *FORWARD* and enter **25** units to move the turtle into the square.
13. **Select** *COLOUR* and then **select** the colour or shade you want to fill the square with.
14. **Select** *FILL* to colour the square.
15. **Select** *PEN DOWN* so that the turtle will draw next time it is given a command.

You now know how to draw and colour a square. But what if you want to draw lots of squares? One way would be to move the turtle to a new starting position with the pen up and then re-enter all the previous instructions, except for number one (you wouldn't want to keep clearing the screen). To do it this way would take a long time and you are likely to make mistakes.

Fortunately, Robot Draw has a *RECORD* command. This makes a list of all of the commands that you use while you are drawing and then lets you play them back again later – a bit like singing into a tape recorder and then listening to yourself time and time again. The best way to explain how to use it is to show you an example.

Recording your commands

Clear the screen to wipe out your first box. Now carry out the following:

- **Select** *RECORD*. When the box appears headed *Name of Macro* type **SQUARE** and press **Return**. The word *RECORD* on the command panel will now be highlighted so you know that the program is recording what you are doing. The name of your recording is displayed at the bottom of the list of commands.
- Now follow steps 2 to 14 above.
- **Select** *COLOUR* and then **select** the colour you want to continue drawing with.
- **Select** *BACKWARD* and enter **25** to move the turtle back to where it started.
- **Select** *RIGHT* and enter **45** degrees so the turtle faces its original direction.
- **Select** *PEN DOWN* so that the turtle will draw next time it is given a command.
- **Select** *RECORD* once again to finish recording.

If you want to see what you have just recorded, clear the screen and then **select** the word *SQUARE* at the bottom of the command panel. The turtle follows your commands one by one until it has re-drawn the coloured square. **Select** it several times to draw a few squares, moving the turtle between each one or they will be drawn on top of each other.

A collection of instructions, such as the ones you have recorded for drawing a square, is known as a **macro**. Let's look at how to define another, this time to draw a house.

To add the macro for this picture to the one you have already created, carry out the following instructions:

1. **Select** the right arrow (to the right of the word *SQUARE*). *Macro 2* should now be displayed.
2. **Select** *RECORD*.
3. Type **HOUSE** and press **Return**.
4. **Select** *FORWARD* and enter **30** units.
5. **Select** *LEFT* and enter **90** degrees.
6. **Select** *FORWARD* and enter **30** units.
7. **Select** *LEFT* and enter **30** degrees.
8. **Select** *FORWARD* and enter **30** units.
9. **Select** *LEFT* and enter **120** degrees.
10. **Select** *FORWARD* and enter **30** units.

11. **Select** *LEFT* and enter **30** degrees.
12. **Select** *FORWARD* and enter **30** units.
13. **Select** *LEFT* and enter **90** degrees.
14. **Select** *RECORD* to finish recording the House macro.

The screen should now be displaying a house shape.

We will now use another of the commands to save your instructions on to tape or disc so that you can use them again at any time.

Saving instructions

Note: Robot Draw files should only be saved on to a blank data disc or tape, not on to the original Fun School master.

Tape users should also refer to the section on *Using tape to load and save data,* at the end of the manual.

Any macros currently in your computer can be saved using the *DISC/TAPE* option.

To save your square and house macros do the following:

- **Select** *DISC/TAPE*
- **Select** *Macro.*
- **Select** *Save.*
- Type **SQRHSE** and press **confirm.**

A message tells you that the macros are being saved and the drawing screen is shown again. Note that all macros you have set up are saved at the same time so both your square and house macros are saved together.

You can also save finished pictures and load them again at a later date. To use this option, **select** *DISC/TAPE* but instead of *MACRO* **select** *PIC.* This indicates that you want to save a picture so enter its name.

Loading instructions

Macro instructions written and saved on a previous occasion can be loaded back into the computer in the following way:

- **Select** *DISC/TAPE*
- **Select** *Macro.*
- **Select** *Load.*
- Type the name of the macro you want and press **confirm.**

After it has loaded, the drawing screen will be shown again. You will also see the name of a macro you have just loaded, at the bottom of the command panel.

You can load several macros at a time but they must have all been saved at the same time as one file, just as you saved SQUARE and HOUSE in a single file.

You can also load finished pictures: **Select** *DISC/TAPE*, but instead of *MACRO* **select** *PIC*. This indicates that you want to load a picture and so enter its name.

Macros within macros

One of the powerful features of Robot Draw is the way that it lets you link together any macros that you have already defined in order to produce complex pictures without having to write long lists of instructions every time. In order to link macros they must all be present in the same file and have been loaded as described above. Let's look at how to draw a row of houses by mixing our original HOUSE instructions with a few extra commands.

Row of Houses

First make sure that the SQUARE macro is in your computer – load SQRHSE if you need to, as this contains both the macros SQUARE and HOUSE. Now do the following:

1. **Select** the right arrow in the macro box until the words *Macro 3* are showing. Now **select** *RECORD*.
2. Enter **HOUSEROW** and press **Return**.
3. **Select** *PEN UP*.
4. **Select** *BACKWARD* and enter **80** units.
5. **Select** *PEN DOWN*.
6. **Select** the left arrow in the macro box to display the word *HOUSE*.
7. **Select** the word *HOUSE*. The first house will now be drawn.
8. **Select** *PEN UP*.
9. **Select** *FORWARD* and enter **50** units.
10. **Select** *PEN DOWN* again.
11. **Select** the left arrow in the macro box to display the word *HOUSE*.
12. **Select** the word *HOUSE*. The second house will now be drawn.
13. **Select** *PEN UP*.
14. **Select** *FORWARD* and enter **50** units.
15. **Select** *PEN DOWN* again.
16. **Select** the left arrow in the macro box to display the word *HOUSE*.
17. **Select** the word *HOUSE*. The third house is now drawn.
18. Finally **select** *RECORD* to finish recording your instructions.

You can now save all three macros using ROW as a filename.To watch the whole thing in operation, clear the screen and then **select** *HOUSEROW* in the macro box.

Editing instructions

The Edit command allows you to change any of your macros once you have finished recording them. So, if you make a mistake while recording a macro, instead of starting all over again just pretend that you haven't made the mistake and carry on to the end of the instructions. Once you have finished the recording you can then edit it and change the incorrect instruction.

As an example we will edit SQUARE, changing the shape to a rectangle. Follow these instructions:

- **Select** the left or right arrows in the macro box until the word *SQUARE* is showing.
- **Select** *EDIT*.

The command panel has now been replaced with the following list of instructions:

Pen down
Forward 50
Left 90
** **Forward 50**
Left 90
Forward 50
Left 90
** **Forward 50**
Left 90
Left 45
Pen up
Forward 25
Pen 12 (the number will vary depending upon the fill colour used)
Fill
Pen 1
Backward 25
Right 45

You should also notice that the command **Pen down** is highlighted. This shows your current position in the macro. If you select the word *STEP* at the bottom of the editing panel, the Pen down instruction will be obeyed and the command **Forward 50** will be highlighted. **Select** *STEP* once more and the turtle will move forward 50 units and the highlight will move to **Left 90.**

What we are going to do is change the number of units in the two lines marked with the ** symbols.

- **Select** *STEP* again so the second **Forward 50** is highlighted.
- **Select** *INSERT*. The editing panel disappears as the turtle waits for you to enter a new command.
- **Select** *FORWARD* and enter **80** units. The editing panel will now reappear with the command **Forward 50** still highlighted and the new instruction following it.
- **Select** *DELETE*. This removes the **Forward 50** line and leaves the **Forward 80** line in its place.
- **Select** *STEP* until the last **Forward 50** line is highlighted and repeat the above steps to change it.
- **Select** *EXIT*. This tells the program that you have finished editing the macro and want to return to the drawing screen.

To see if your editing has been successful, clear the screen and then **select** *SQUARE* in the macro box. Instead of a square a tall rectangle should have been drawn.

Complex macros

To have a final look at macros, load up the file *Samples* provided on the Fun School disc. **Select** the right arrow in the macro box until you find a macro called *Box*. **Select** this and you will see a small square drawn on the screen. Examine how this was written by selecting *EDIT*. You can see that only seven instructions were required to draw the box.

Now have a look at the macro named *Boxes*. This draws seven boxes and uses a number of colours (shades) to fill some overlapping areas. Again, examine how this was written by selecting *EDIT*. If you move through this using *STEP* you will see that it is a complicated macro involving many instructions.

Boxes calls up the macro *Box* seven times. (If every single instruction was written down in *Boxes*, there would be seven times the instructions than there are!).

Clear the screen and call up *Box* again. Try editing this macro. You could, for example, increase the length of each side to 50 units. Once you have done this have another look at *Boxes*. Changing a macro that is called up by another macro can lead to some fairly surprising effects!

Reload *Samples* and clear the screen. Now see if you can edit *Boxes* to:

- Alter the fill colours (shades).
- Change the speed at which the macro is drawn.
- Alter the position of the macro on the screen.

If you have any problems, read the section *Editing instructions* again.

Circles

Now examine the macro called *Circle*. **Select** this and a circle will be drawn on the screen.

The circle is one of the most difficult shapes to draw using a program such as this. Stop for a moment and think why. Robot Draw can draw lines forwards and backwards and the turtle can be made to move right or left. But there is no instruction for curved drawings. In order to draw a shape like a circle, the turtle must be instructed to move a little forward, turn slightly, move another tiny amount, turn again, and so on.

Select *EDIT* and step through this macro. If you were to count the number of instructions you would find that there are lots! All these are involved in making slight turns and tiny forward movements until the circle has been completed.

The macro *Circle* is called up in *Colour*. Have a look at this. You will see that it draws 14 overlapping circles and colours 12 areas. This is the most complex macro you have seen yet. It involves a lot of instructions, and calls up *Circle* 14 times. If every instruction were written within *Colour*, line by line, how many instructions do you think would be needed? (The answer is hundreds!)

Planetary Maths

Planetary Maths is an all-action, all-excitement, altogether out-of-this-world way of testing your mathematical ability. So, all aboard the Star Ship Totaliser to destroy the enemy force fields.

Before you start the game, you must pick the type of questions that you want to answer. To the left of the screen you will see four mathematical symbols, each within a small box. When the game is first loaded there is a highlighted "+" box. **Select** any of the other boxes and they too will become highlighted.

The boxes that are highlighted show which types of sums have been chosen for the game. If you want to turn a box off, just **select** it again, and the highlight will disappear. For a game, you can pick one, two, three or all four types of sum.

There are five levels of difficulty in Planetary Maths: Level 1 is fairly easy, Level 5 is very hard! If you **select** the word *Level* you will see that the number in any highlighted box changes. This number is the level of play.

You can play this game using different levels for each type of sum. For example, suppose you find addition and subtraction fairly easy. You can play these on Level 5. If you're not quite so happy with multiplication and division, play these on Level 2.

First **select** the boxes for plus and minus so that they are highlighted. Then **select** *Level* until the numbers in these boxes rise to 5. **Select** the boxes for plus and minus again to turn them off. Now turn on the multiply and divide boxes. **Select** *Level* again and the numbers in these two boxes will go up to 2. Before you play the game, turn the boxes for plus and minus back on. (Pressing the **Levels** key has the same effect as selecting *Level*).

When you are ready to start the game, **select** *Start* or press **Return**. A sum will appear on the right of the screen. If you type in the right answer a missile will be fired to destroy the force field. If you type in the wrong answer, you will hear a warning noise. You might still have time to try another number before the sum smashes into the force field and depletes your energy reserves! If this happens, the correct answer will appear in the answer box.

After you have destroyed a number of force fields showing one type of question, the level of difficulty for that type of sum automatically increases. If you succeed on Level 5, the force field begins to move to the right. You now have far less time to solve the sums!

Your energy reserves are shown at the bottom of the screen. Once the strip disappears, your quest has been halted and the game is over.

Some of the questions in the game may involve money. If you see any like this, do not type in the pound sign, just the numbers with or without the decimal point – either will be accepted.

You can earn bonus points by answering questions quickly. Higher bonuses can be earned, the further the force field is to the right. You can control the position of this yourself by clicking on *Force*.

As the game proceeds you may get an occasional message which tells you how you are doing. At the end, if your score is high enough, you will be able to put your name on the High-Score Table. For disc users only, this table is saved after you have finished playing and press the **Quit** key to go back to Fun School 3's main menu. (The table will not be saved if the disc is write protected.)

Treasure Search

Yo ho ho, me 'earties, weigh anchor an' follow Cap'n Robbie in Fun School's Treasure Search! There's an extra 'elping o' Cap'n Rob's grog, if you's can 'elp show 'im where the treasure's hid! (Then I promises to 'ave a go at Sentences to improves me Inglish!)

Somewhere on the map is a chest full of gold and jewels. Sometimes you will see a map of an island, while at other times the treasure could be hidden elsewhere! The route to the treasure is written out in a list on the right of the screen. All you have to do is follow these directions until you find the valuables.

Your current position is shown, on the map, by a marker. When the game first starts, you may not be able to see this because above it is a cross. If you move this cross over the map you will notice that the distance that the cross has moved away from the marker is written on the screen. The cross is used to mark the next place to go. Now look at the list of directions on the right of the screen. The one highlighted is where you must take the cross to first. If, for example, the instruction:

1200m W

is highlighted, you should move the cross to the left (west) until the distance of exactly 1200m appears on the screen. There is a compass at the top of the screen to help you decide which way to move. When you have moved the cross to the correct location, press **confirm**, and the marker will move to the new position.

If you get this right, Robbie will give you the thumbs up and your new instruction will be highlighted. If you should get it wrong, you will get the thumbs down and the instruction will change to help you get to where you should have been.

When you have worked your way through every instruction on the list, Cap'n Rob burrows into the ground and re-emerges with the treasure! Good luck me 'earties, an' 'appy 'unting!

Database

Fun School 3's Database program will introduce you to one of the most important uses of computers in offices, factories and even at home!

A database is an electronic filing system. Filing systems are very important as they are used to store information.

You might have seen a set of filing cabinets in the school office. In these cabinets will be lots of folders, one for each pupil in your school, including you. Inside your folder there will be some information about you – things like your date of birth, your address and so on. It may even contain copies of your last report!

Every different workplace needs a filing system: A doctor needs to know about his patients; a shopkeeper needs to know what's in stock; and a librarian needs to know what books are in the library, which books are out and who they are out to.

Information can be written onto paper or card, and stored in a filing cabinet. It takes time to write the information out, more time to file it and yet more time to search for it when it is needed. If a piece of paper is put into the wrong place it will get lost. And there's one other problem: Filing cabinets take up a lot of space!

Databases can overcome many of these problems. They are often much quicker to use and as disc or tape is used to store files much less space is needed. Just like a filing cabinet, a database stores pieces of information – *records* – in a collection called a *file*. Files can be stored and used later.

Now let's have a look at Fun School's Database program. First, you will make a simple file of your own. Then, you can have a look at one of the files produced using this program.

Load up the program and have a look at the screen. All the functions are shown around the right and bottom of the screen. To use any of these you move the selector to the one that you want and **select** it. Don't worry about all the functions at the moment, this manual will tell you how to use each in turn.

The large area at the top left of the screen can show you a record. At the moment this is blank because there are no files and therefore no records in the database.

Look at the top of the screen and you will see the message *Record 0 of 0*. The first

number tells you which record number is on the screen, the second how many records are held in the file. When there are no records in the database, two zeros are displayed.

There are two ways of putting information into the database: The first is to ADD new records by typing them in using the keyboard; the second is to load a file from tape or disc.

Adding new records

You can add records to a file that already contains information, or to a new, empty one. When you add a new record, it is always placed at the end of the file. To practice adding records, let's try entering the names and addresses of some children using the keyboard.

Select the box labelled *ADD*. You will see a cursor appear in the thin line near to the top of the screen. This is where the first piece of data will be entered. Type in the name: **STEVEN SMITH** and press **Return.**

The cursor will then move to a second, larger area which is where you can enter Steven's address. Type in the following, pressing **Return** at the end of each line:

31 OLD LANE
PEMBLETON
MANCHESTER

The cursor will now be at the start of a line which you are going to leave blank. Press **Return** and it will disappear. Congratulations, you have just entered your first Database record.

Select *ADD* again and then enter:

ROBERT JONES

on the top line. Press **Return** and type in his address:

18 MANOR GROVE
TWEEDLE
MANCHESTER

not forgetting to press **Return** at the end of each line. **Select** *ADD*, then enter:

TONY SMITH
116 PARK GROVE
HAMBLETON
MANCHESTER

Now, add one more record:

PETER ROPER
22 WHITEHOUSE LANE
HAMBLETON
MANCHESTER

Now that you have produced a file containing four records, you can take a look at some of the things a database can do. This program is controlled using the buttons to the right and bottom of the screen. Let's examine what each of them does:

START

Look at the words at the very top of the screen. They should read *Record 4 of 4.* The first number tells you which record of the file you are currently looking at and the second tells you how many records there are in the file. **Select** *START.* The screen will display the first, or starting, record and the top of the screen will hold the message *Record 1 of 4.* Using *START* always shows the first record in a file.

END

Now **select** *END.* This works exactly like *START* except that it always shows you the last record in a file.

SEARCH

Searching is a fast way of finding the records that you want to look at. **Select** *SEARCH* and the message *What word* will appear on the screen. On this occasion type in: **ROPER** and press **Return**.

The program next asks you if you want a precise match of the upper and lower case letters that you have just entered. Generally, it is best to answer **N** for no. Do this and Peter Roper's record will appear on the screen.

Another message then asks you to press the **Spacebar.** The database then continues to look through the file for any more records which contain the word ROPER.

Try searching again, but this time look for the word SMITH. You will, first of all, see record 1, STEVEN SMITH displayed on the screen. After pressing the **Spacebar** or mouse button to continue the search, you will see record 3, TONY SMITH, displayed.

Search can be used in this way to find a certain word, or even a sequence of words, from anywhere within a file.

SORT

This is used to change the order in which records are stored in a file. At the moment, the four names in your file are stored in the following order:

Record	*Name*
1	STEVEN SMITH
2	ROBERT JONES
3	TONY SMITH
4	PETER ROPER

You can change this so that they are stored in alphabetical order. **Select** *SORT* and you will be asked if you want the names to be sorted beginning with the letter A or the letter Z. If you select A the records will be sorted, so that their top lines are in alphabetical order. If you pick Z, the records will be placed in the opposite order to this.

For now, press A. **Select** *FORWARD* and you will find that the records have been sorted into the following order:

Record	*Name*
1	PETER ROPER
2	ROBERT JONES
3	STEVEN SMITH
4	TONY SMITH

If these names were to be used in, for example, a class register, they would still not be in the correct order. Such a file should be based upon the order of surnames and read JONES, ROPER, SMITH, SMITH.

This is a very important lesson about storing information in a database: When records are sorted, the program always looks at the first letter of whatever words have been entered. In this example, the first letter found was P for Peter, and not the R of his surname, Roper.

This problem can be overcome by entering each child's last name before his first name. If the records were going to be used for something such as a class register, they should have been entered as:

ROPER PETER
JONES ROBERT
SMITH STEVEN
SMITH TONY

Now let's examine one of the files in your Fun School software. First, **select** *CLEAR*. This will delete all the records currently in the database. You will be asked if you are sure that you want to clear every record, so enter **Y**.If you had made a mistake, pressing **N** tells the computer not to *CLEAR* all the files.

To load the file, **select** *LOAD*. Disc users will see a menu which contains the names of all of the database files on the disc. Tape users should simply type in the required filename.

The file that you are going to work with is called *Shopping*. Find this in the list and **select** it (tape users must enter the filename and press **Return**). Once the file is loaded you will see the database screen again.

The top line tells you that there are 32 records in this file, and you are now looking at the first record. This is headed *Apples*. Underneath the first line are listed three different prices. The first is for apples bought at a supermarket, the second at a greengrocer's shop and the third in a market.

All the records in *Shopping* compare the price of goods bought at different places. The file has already been sorted into alphabetical order.

Select *FORWARD* and you will be able to go through all the records in the file. Selecting *START* always takes you back to the first record (Apples) while *END* will take you to the last.

Imagine that there are a few mistakes in the file and you wish to correct them:

- Use *SEARCH* to find the record *Lollipops*. The data in this is wrong.
- First, **select** *DELETE*, then enter **Y**. Notice that, unlike *CLEAR* which erases all the records in a file, *DELETE* only rubs out the record on the screen. Now **select** *ADD* and enter the following data:

Lollipops
Supermarket 47p/10
Corner Shop 6p each
Newsagent 5p each

Don't forget to press **Return** at the end of each line.

- The data for Oranges is also wrong. Search for this record and delete it. Now use *ADD* to put in the correct data:

Oranges
Supermarket 16p each
Greengrocer 20p each
Market 16p each

- Once you have done this, you will find that record 31 is now Lollipops and record 32 is Oranges. The new records have been added at the end. To get the whole file back into alphabetical order, select *SORT* and enter **A**.

SAVE

You may wish to save this data. Disc users should make sure they put a formatted disc into the drive – not your Fun School 3 Program disc. Tape users should use a blank tape (see also the section on *Using tape to load and save data* if you are unsure of the

procedure). **Select** *SAVE*. Type in a title for the file, this time call it **Shops,** then press **Return.**

PRINT

If you have a printer you may wish to practice using the database to print records. First, make sure that the printer is turned on and loaded with paper (ask an adult to help you). Then **select** *PRINT* and the program will ask you: *This record or all R/A?* To print the record on the screen only, press **R**; to print the whole file, press **A.** For now, press **R**.

The next question asks you whether to print the record horizontally or vertically. If you select horizontal the program will try to fit all the information from the record onto a single line. Selecting vertical will print the record as it appears on the screen, with six separate lines of writing. Press **H** for horizontal. Finally, the program asks you if you want the record numbers to be printed. Answer **Y** or **N** and a record will be printed.

Now have another go at printing the record on the screen, but this time print it vertically.

If you have continuous feed paper in the printer you could try printing all the records. But, as this will use up a lot of paper, why not wait until you are able to print out your own data?

TEST

This final option is, as far as we are aware, unique to the Fun School 3 database. It allows you to use the database as a test or quiz. Only certain files are suitable for use with TEST.

Load TABTEST.DAT . This file, along with some others in the package, has been designed to test your ability with tables.

If you look at the first record in the file you will see the number 4 on the top line. Below this is the first half of a sum:

2 x 2

Now look at the second record. The top line now contains the number 6. Below this is:

2 x 3 3 x 2

This time there are two sums, because both 2 x 3 and 3 x 2 make 6.

As you go through the records, you will notice that some contain large numbers of sums. The answers in any one record will always come to the number on the first line.

Now, **select** *TEST*. The program will ask you whether you want to be tested on all the file (**A**) or just a part of it (**P**). For now, press **P**. You then have to tell the program how many records you want to be tested on. Using only part of the file, you are tested from record number 1. If you type in 10, you will be tested on the first 10 records. This time, type **2** and press **Return**.

You will notice that the top line of the database has now disappeared. The large box can now be thought of as a question box. Below this, a cursor has appeared along with the words *Answer Please.*

As you are looking at the first record, the large box is displaying the question:

2 x 2

Type in **4** and press **Return**. The program will tell you that you are right. Press any key and the second question will appear. After you have answered this, you have used up the two questions that you asked for.

Try the test again but instead of entering **P** for part, type in **A** for all the file. Now the program will ask you if you want to answer the questions in order (**O**) or at random (**R**). Answering the questions in order means that you will start with the first record, then move onto the second and so on. Random means in any order.

If you enter an incorrect answer, you should try again. If you continue to answer the question incorrectly, the answer is revealed, one digit at at time, at the top of the screen. You must enter the correct response to be able to continue with the test. If you wish to leave the test, press the **Quit** key.

You should now be familiar with some of the ways in which Database can be used. There is a great deal more that you can use this program for than has, so far, been suggested in this manual. How many can you think of?

Sentences

Sentences is about the rules of writing. Fun School's Robbie will show you some sentences, many of which will have mistakes in them. You might find spelling mistakes, commas in the wrong place, wrongly used capital letters and other sorts of errors. You should carefully examine each sentence, find any mistakes and put them right.

If you find a mistake, **select** it and whatever you select will be underlined. Type in what you think the correct version should be. As you type, this will appear in the bar at the bottom of the screen. When you are happy with your version, press **Return** and it will replace the old version.

For example, the screen might show you this sentence:

I had a fight with Jamie so I went back to the hous allone.

You might realise that *hous* is wrong and should be spelt *house*. ***Select*** *hous*, then type in *house* and press **Return**. The screen will then show:

I had a fight with Jamie so I went back to the house allone.

There is one more error. To correct this, **select** *allone* and type in the correct spelling

(*alone*). After pressing **Return** the screen will show:

I had a fight with Jamie so I went back to the house alone.

Once you think that a sentence is correct, **select** *OK*. If you are right, Robbie will give you the message *That's right* and wait for you to press a key before showing you the next sentence. If there are still mistakes in the sentence, Robbie will tell you that the sentence is wrong. Take a closer look for another mistake.

If you cannot see a mistake, **select** *Clue* and an error in the sentence will be underlined. Look carefully at the underlined word and try to type in what it should be, then press **Return**. If you are wrong three times, Robbie will give you a clue automatically. Should any sentence have you totally baffled, you can press **N** to move onto a different one.

There are four different levels to Sentences: Level 1 questions are fairly easy. Questions become increasingly more difficult the higher the level becomes. You can change level by pressing the **Levels** key.

notes for parents and teachers

Fun School 3 has been designed to take full account of the new National Curriculum. This outlines much of the core material which will now be taught in schools. The National Curriculum is arranged into a number of over-lapping Key Stages. The lowest level of study is Key Stage 1 and is intended for children below the age of 7. Material in Key Stage 2 is designed for children between the ages of 7 and 11 and that in Key Stage 3 for the 11 to 14 age range.

The programs which make up this package are directed towards numerous aspects of Key Stage 2 (including topics covered in Mathematics, English and Information Technology). As it is fully accepted that some children will be able to cover material in Key Stage 2 more rapidly than others, certain elements of Key Stage 3 are introduced in these programs.

It is naturally expected that the skills covered in Fun School 3 will be reinforced in other ways (a number of suggestions or further ideas are given in this manual).

Fun School 3 is a highly versatile package. The degree of sophistication achieved with Robot Draw and Database will increase with greater maturity: Young children will not be prohibited from exploring simple ideas, those approaching their secondary education may be able to produce complex database files and be well on the way to understanding the construction of computer programs.

Each of the four games can be played on a number of levels and, as a result, these

lend themselves to a wide age and ability range. Children under seven may be able to play any of the games on their lowest levels, whereas those of nine and above may be stretched at the highest levels.

Some important words of advice: Children achieve most when they are able to learn at their own pace. Learning should be fun. Do not expect too much from any child. Give praise where it is due and try not to be critical of what may appear to be silly mistakes.

Word Search

Word Search aims to encourage children to learn correct spellings of words using the theme of a popular game – searching for 10 listed words in a jumbled grid. The same words can be used time and time again, as each occasion a game is played a new grid is created.

The program comes complete with five sets of words, each containing five lists of 10 words each. Between them, three sets contain 150 of the most commonly misspelt words. The remaining two sets cover themes of places and dates.

As with many Fun School programs, flexibility is built in. By pressing the **EDIT** key you can access an editor screen which allows you to amend existing lists, create and save new ones and load files from tape or disc.

Word Search Editor key list – (except Commodore 64)

	Edit	Load	Save	Clear	Quit
Amiga	F8	F1	F2	F3	F4
BBC	F8	F1	F2	F3	F4
C64	Editing is not applicable on this version				
CPC464/6128	8 (These keys must be accessed from the numeric keypad)	1	2	3	4
PC	F8	F1	F2	F3	F4
Amstrad PCW	Cut	F1	F3	F5	F7
Spectrum	symb 8	symb 1	symb 2	symb 3	symb 4
ST	F8	F1	F2	F3	F4

When you press the **Edit** key the Editor will be shown and the current word set is displayed on the screen. You amend an existing file using a combination of the pointer and cursor keys. The pointer moves between the Subject box and the five Group boxes.

The contents of the Subject box and the top line of each Group box provide the text

that appears in the single line window beneath the word grid when playing the game. To alter any words in a Group, **select** the relevant box. Move to the word that you wish to change. Remove the existing word using the **Delete** key and then type in the new word.

When the lists have been changed you can save them to tape or disc by pressing the **Save** key. You will be prompted for a filename. Either type this in, or (disc users only) if you wish to replace an existing file, **select** the relevant filename.

To create a completely new set of words, press the **Clear** key to wipe the whole screen. You can now enter new words from the keyboard, moving between spaces, and save them as previously described. (A word of caution: You can only save files containing complete lists without any duplicated words.)

To load a file, press the **Load** key to enter the name of the file you wish to load and press **Return**. The new words will appear in the Group boxes.

You can return to the game, using the words on the Editor screen, at any time by pressing the **Quit** key.

Editing data files on the Commodore 64

Commodore 64 owners who have a copy of Mini Office 2 can edit Word Search files using this utility. These files should then be saved in Mini Office format.

Further ideas

- Encourage children to create their own word sets for entry into the Editor using spelling lists brought home from school. This could be extended by asking them to make lists of words that they are told they often misspell (or words that are highlighted by teachers in school exercise books). Help them to enter these into the word list Editor, checking that the correctly spelt version of each word is being entered. Word sets prepared in this way can be used in the game.
- Suggest that the children try to create their own jumbled word grids from the lists that appear on the screen. They could pass these onto friends to use as puzzles. You could extend this idea to have children think about the speed at which computers operate. Have them time how long it takes to produce such a word grid manually. Then suggest that they compare this to the length of time taken by the computer to carry out the same task.
- To encourage children to think about the words contained in each list (and not just their spellings) suggest that they try to create a few sentences which contain every word in the list. The sentences must of course make sense, with as few additional words as possible. (NB: Only certain lists are suitable for this exercise. Those contained in the Dates and Places files are not suitable.)

There are three levels of difficulty:

Level	*Word direction*
1	Horizontal and vertical words
2	As level 1, plus diagonal words
3	As level 2, plus reversed words

Robot Draw

Robot Draw is the Fun School 3 version of the popular educational programming language Logo. Not only is this an excellent introductory programming language but it can also be used to explore various geometrical relationships.

Because of the use of relative spatial movement and angles of rotation, younger children may find Robot Draw a difficult program to use. If this proves to be the case, it is recommended that you first consider examining Journey, a program in the 5 - 7 year olds package of Fun School 3. This program, along with its manual introduces the ideas of relative movement and the Logo programming language. The children should then attempt Treasure Search in the present package which, among other ideas, considers angles of rotation.

Further ideas

Angles of rotation

- The ideas covered in Treasure Search can be reinforced using Robot Draw. With the Robot Draw turtle in its default position have the children enter **LEFT 90** to make it point upwards. They should then enter the following lines:

FORWARD 80
BACKWARD 80
RIGHT 45
FORWARD 80
BACKWARD 80
RIGHT 45
FORWARD 80
BACKWARD 80

Robot Draw will start to draw the points of the compass. Have the children continue this routine as the complete compass is drawn.

- Tell them that they are adding 45 degrees each time. Ask them how many degrees took the turtle back to the starting line (the magic number is, of course, 360). This process will encourage them to realise that the direction of movement depends not only upon the angle entered, but also the starting direction.

- Have the children repeat this process using smaller angles, such as 30 degrees and 20 degrees. Now have them use LEFT statements instead of RIGHT. They should see that these two statements have opposite effects. You could point out that there is a mathematical relationship linking any LEFT and RIGHT angle:

  LEFT = 360 - RIGHT RIGHT = 360 - LEFT

Geometrical Drawings

- Encourage the children to examine the differences between various types of triangles. The sequence:

  RIGHT 30
  FORWARD 50
  RIGHT 120
  FORWARD 50
  RIGHT 120
  FORWARD 50

  will produce an equilateral triangle (one with sides of identical length).
- By first making scale drawings, encourage the children to attempt to produce a similar sequence for an isosceles triangle (where two sides and two angles are identical) and another for a right angled triangle
- Squares and rectangles can be produced easily. Have the children try this sequence which produces a hexagon (six-sided figure):

  LEFT 30
  FORWARD 40
  RIGHT 60
  FORWARD 40
  RIGHT 60
  FORWARD 40
  RIGHT 60
  FORWARD 40
  RIGHT 60
  FORWARD 40
  RIGHT 60
  FORWARD 40

- Encourage the children to examine the possibilities of regular figures with different numbers of sides. (There is, in fact, a mathematical relationship between the angle of rotation and the number of sides).

Examples of the above shapes, along with slightly more complex macros based upon these, are included in the file *Samples*.

Programming

As with any computer language, precise instructions are used to produce precise results. Naturally, any child can simply use Robot Draw to produce ad hoc pictures. On the whole, the results of such an approach would be poor and the child would soon become frustrated.

First Robot Draw Programs

The principles of good programming should be encouraged from the start:

- You should first encourage the children to plan their work. Planning a Logo program is best achieved using graph or squared paper. On this the children should draw an outline sketch of the picture to be produced within Robot Draw. Outline plans should be drawn in pencil so that any mistakes can be rubbed out. The paper used should reflect the dimensions of the Robot Draw screen so that scale diagrams are produced.

Since each computer is designed differently, we have supplied you with a table giving that computer's screen dimensions:

	Screen width	Screen height	Default position(x, y)
Spectrum	184	192	92,96
Commodore 64	200	200	100,100
Amstrad CPC	200	200	100,100
BBC	200	200	100,100
Amstrad PCW	192	200	100,100
PC	200	200	100,100
Atari ST	192	200	100,100
Amiga	192	200	100,100

- After drawing a plan this should be converted into lines of programming. This is, for most children, the most difficult stage of using a Logo-type language. It is also the most essential. Encourage the children to attempt the following:

First mark in (as say, a dot) the default position at which the turtle will start on the screen (see above). Then mark in the position that the drawing starts. The first job is to get from the default position to the drawing's start position. Mark a route on the plan using only right angles where turns are required.

The children are now in a position to write down the first few lines of programming, which may look something like:

PEN UP
RIGHT 90
FORWARD 20
RIGHT 90
FORWARD 10
PEN DOWN

Notice that the first line prevents the turtle from drawing while it is moved to its new position. The last line will allow drawing to begin. Now help the children to go over the lines drawn on the diagram, converting each line in turn into a line of programming. Don't forget that each time the pen moves to a new position the *PEN UP* command must be used.

- Once the diagram has been converted to a sequence of instructions, these can be put into Robot Draw. Before starting, remind the children to **select** *RECORD*. This should be used again at the end. The children should take care in copying from their list. Any mistakes should, however, be ignored as they can be corrected later using the *EDIT* facility.
- The final stage is to make any corrections to the drawing using *EDIT*. Reference should again be made to the list produced from the diagram.

Further Programming Ideas

Once the children have become proficient at producing simple drawings within Robot Draw, it should be possible to introduce the following:

- Using angles of only 90 degrees is far too prohibitive for most uses. A simple operation such as moving the turtle from A to B will involve four operations if just right angles are used. This can be reduced to two, if a direct route is taken. Scale drawings of the children's plans become increasingly more important if additional angles are to be used. The angle and distance of movement must be carefully measured before being used in Robot Draw.
- As the children become more confident in translating plans to lines of programming, it should be possible to dispense with written instructions. The children should be able to transpose the dimensions and angles used on the diagram directly into Robot Draw.
- Encourage them to look for recurring shapes within any drawing. Each shape that is used more than once can be defined as a macro and called up within the main program. For more complex shapes macros can be nested within macros. There is, theoretically, no limit to the number of nested statements that can be used. In fact,

macros can be within a macro that is itself inside another macro. (In reality, the limit is the number of macros in a single file, and therefore a single drawing, is 20.)

Suggest the children attempt the following:

- Have another look at how *Houserow* was produced using the macro *House*. Have them think about designing macros for windows, doors and chimneys. These will just be simple squares or rectangles and will be used at least three times each.
- After deciding upon each of these macros and entering them into Robot Draw, *Houserow* could be adapted to make it more realistic (and much more complex!) To do this the children could start the whole picture again: Take the pen up, move the turtle to the top left of the screen, put the pen down and call up the macro *House*. Then move the turtle to the starting position for the first window and call up the macro for this square. Then carry on...
- Alternatively, they could decide to edit the existing HOUSEROW macro.
- Ask them to think about making the picture even more complex. How about colouring in the houses? Can another macro be added for the garden paths? Are there any other ways in which the picture can be improved?

Macros are at the heart of good Logo programming. They prevent having to use identical statements over and over again. Their equivalent in other programming languages are called subroutines/procedures. Macros save a great deal of time, space and computer memory. Remember Colour, a drawing of circles, which used the macro *Circle*? This took only a few lines of programming. Without the *Circle* macro it would have been hundreds of lines long!

Included in Fun School are a number of sample macros. These can be found in the file *Samples*. Encourage the children to examine how these macros have been constructed, adapt them and even use them as macros within their own programs.

The macros included are:

Circle	Draws a small circle in the centre of the screen.
Colour	Uses the macro *Circle* to draw a sequence of overlapping circles, and colours (shades) 14 areas.
Box	Draws a small square in the centre of the screen.
Boxes	Calls up *Box* and draws a sequence of overlapping squares and colours (shades) some areas.
Hexagon	Draws a regular, six-sided figure on the screen.
Moves	Called up to provide movement between individual hexagons in the following three macros:

Hexagons	Calls up both *Hexagon* and *Moves*. Draws a sequence of 11 overlapping hexagons.
Hex2	A refinement of *Hexagons*. This produces two sequences of hexagons of different colours (shades).
Hex3	A further refinement of Hexagons, this time using fill colours.
Pent	Draws a pentagon.
Triangle	Draws an equilateral triangle.
Pentas	Calls upon *Pent* and draws a series of overlapping pentagons and shades some areas.
Maze	Draws a single line which moves from the centre of the screen to the outside in a maze-like fashion.
Journey	A colourful journey across the screen, includes numerous coloured lines and fill colours.
Triangs1	Calls upon *Triangle* to produce a pattern of overlapping triangles (how many can you count?)
Triangs2	A refinement of *Triangs1* which includes alternating fill colours (shades).

As many of these are intended to be used in children's own pictures, they do not necessarily start with the PEN DOWN command. If you should run a macro and find that you have a blank screen, invoke CLEAR, **select** PEN DOWN and try again.

And finally...
The possibilities that Robot Draw offers are endless. You should encourage the children to be adventurous, think logically, but most importantly, have fun!

Planetary Maths

Planetary Maths brings colour and excitement to the, possibly mundane, task of mental arithmetic. In addition to this, when played with friends, the element of competition will further motivate children to get the answers right and, therefore, win!

Each type of sum can be played independently or in combination with any number of other types of sum. In addition, the level for any one type of sum can be set separately of the others. Because of this, the program can be tailored to any child's individual ability.

Encourage the children, especially when first playing the game, to make a note (from the screen) of the following:

- The level of play for each type of sum.

- Any comments that appear on the screen.
- Their final score(s).

Keeping records, such as these, will help you to identify any areas of weakness. They will also serve to indicate the progress your children are making.

Treasure Search

Treasure Search is a game that introduces the ideas of relative distances and directions of movement. The skills gained by using this game can later be built upon using Robot Draw.

A mixture of distance units has been deliberately chosen to encourage children to consider the relationship between metres and kilometres. Similarly, compass points are not just shown as N, S, E and W (along with the intervening NE, NW, SE and SW) but also as angles of rotation.

Pressing the appropriate key you can call up the Treasure Search options screen. The currently selected options will be highlighted. You can change these by selecting relevant, alternative options:

Compass On/Off — This option can be used to switch off the compass display once children are confidently able to move in the required direction without having to refer to this symbol.

Compass Type — The direction instructions are displayed in a manner corresponding to the currently selected compass type, of which three are available – four directions: N, S, E and W; eight directions: N, NE, E, SE, S, SW, W and NW; and four directions: 0, 90, 180 and 270 degrees.

Timer On/Off — Turning the timer on introduces an element of competition into the game. A small bar appears beneath the direction instructions which begins to shrink as children rush to locate and indicate the next checkpoint. The bar is reset to its original length every time a location is correctly found. If the bar shrinks to nothing the game finishes.

Map selection — A variety of maps is provided as backgrounds for Treasure Search. These are selected using the Town, School and Class icons.

When you are happy with your chosen selections, **select** the *Return to game* box to leave the options screen.

Further ideas

Because Treasure Search provides an introduction to relative distances and movements in different directions, it should be used before Robot Draw.

Try to encourage children to think about the following ideas:

- Some distances are marked in kilometres, some in metres. There is a scale indicating the relative distances on the bottom of the screen. Ask children to try and find out (using the program) how many metres there are in a kilometre. While using the game, suggest that they use a ruler to try to discover how many centimetres, on the screen, represents 1 kilometre. (This can be done by moving the pointer until the distance reads either 1 km or 1000 m and then measuring the distance of the line left behind the white cross).
- Once the scale covered by the map on the screen has been established, encourage the children to draw a scale diagram of one of the locations. On this they could attempt to map the sequence of instructions as given on the screen. If this is done in tandem with the game, the final location on the screen and the diagram should be the same.
- Encourage the children to examine the three compass types and draw them on paper. Ask them to compare the compass which shows the angles of rotation with the other two. Ask questions such as: *How many degrees is east?* Questions such as: *If you are pointing north, how many degrees to the right should you move to be pointing east?* will introduce the angles of movement, as used in Robot Draw.
- Children could play the game for real in a large garden, or on an open space such as the school field. One child should hide some treasure such as a small ball or some other toy. Using about 10 directions, the other children should then map out a route, similar to those used in the game, to get from a starting point to the treasure. For distances, they could use a metre rule, or just paces. Once they have written down their route, they should then give it to another child who will have to follow it to find the prize!
- When children have fully mastered Treasure Search and are at least acquainted with the fundamentals of Robot Draw get them to try and tie the two programs together. Working through a Treasure Search map, suggest that they copy each direction from the screen. They can then produce a scale drawing of the route on paper. Using this, encourage the children to work out a Logo routine to be entered into Robot Draw. Once this is entered, the same route as used in Treasure Search should be seen on the Robot Draw screen.

Database

The Fun School Database provides a user-friendly means of introducing the over sevens to one of the most fundamental serious uses of a computer. It comes with 10

pre-written files. As well as providing a hands-on introduction to this program, these files have each been designed as educational utilities in themselves.

Database has been produced to be as flexible as possible. You could use it to:

- Examine or search existing files for information.
- Amend existing files.
- Use existing files as tests.
- Produce new files to store information.
- Produce new files for tests.

Parent/Teacher produced tests

If you intend to make use of the Test facility with your own Database files you should bear in mind that the top field will always hold the answer and the large field beneath it, the question. A simple example could be a straightforward question and answer set:

Buckingham Palace
Where does the Queen of England live?

or:

Charles Dickens
Who wrote "Great Expectations"?

However, this approach limits the use of your file to test conditions only. It might be better to turn the large field into a statement. The file could then be used as a learning aid as well as for tests.

A second, preferred, approach is to structure the information contained in the large field. This will not only provide highly informative files, but will also introduce children to the form of structured information used in professional databases.

Of the database files on the Fun School 3 package, *Dates* provides an example of the use of statements which are not highly structured. On the other hand, *Europe* shows how highly structured data can be used. Each of these files ideally lends itself to being used as tests, even though the use of questions has been totally avoided.

Database files

Seven of the files provided with Fun School 3 deal with various properties of numbers, concentrating on the use of the 2 to 12 times tables:

Numbers

This is an introductory file. Once children have briefly looked at *Tabtest*, as described earlier in this manual, you should load and examine *Numbers*. This file shows some properties of the numbers 2 to 99. It introduces the ideas of factors and prime numbers.

(Numbers which divide into another number are said to be its factors. A number that can only be divided by itself and 1 is a prime number.) Also shown is the simple, factorial composition of numbers (as relevant to multiplication tables). For example, under 04 the entry is simply 2 x 2 = 4. Many larger numbers can be made up in different ways, for example 2 and 6 multiply together to give 12, as do 3 and 4. If children examine the file carefully, they will discover all the numbers below 100 found in the 2 to 12 times tables. (Numbers less than 10 have a zero in front of them so that they don't get out of order if re-sorted. Tell the children to ignore this leading zero.)

The concept of factors is not confined to classroom multiplication tables. In order to encourage children to consider this idea, other factors are listed which fall outside normal tables.

The file could be used by children to extend their own multiplication tables. Under 26, for example, 2 is listed as a factor. Using simple division, children may be able to work out that the remaining factor is 13, hence 13 x 2 = 26.

The factors listed within the file have been deliberately left incomplete in the hope that children can be encouraged to replace records with data that they calculate for themselves.

Once children have become familiar with the concept of factors, they may wish to use *Factors*, which is described below.

If the idea of prime numbers is a new one, you should encourage the children to examine each entry where this term is used. You could ask them to write down a selection of these numbers and try to work out what they all have in common.

Numbers is not suitable for use with the test facility.

Tables 4

This file deals with the two, three and four times tables. In the large area of each file, six multiplication sums are given, in order. These represent either the first or second half of the relevant multiplication table. The answers to each sum are, similarly, arranged in order in the strip at the top of the record. This file could be used to simply learn multiplication tables.

The full sums (questions from the bottom of the file, answers from the top) could be copied from the screen and used to form complete tables or a revision list. Alternatively, you could use it to test children's knowledge of their tables. If the test facility is used, six answers should be entered with spaces between each response.

Tables 6, Tables 8 and Table 12

These each extend the above file, going up to the six, eight and twelve times tables respectively. If children are having particular problems with one or more multiplication table, you could prepare your own files which contain only the relevant records. To do this, load a *Tables* file which contains the relevant multiplication table(s) and delete those records in which you are not interested. After this, save the amended file to a data disc or tape.

Tabtest
This is designed to test children's ability at multiplying numbers found throughout the common multiplication tables. As with *Numbers*, it contains the various factorial varieties for each product. Again, you can amend this file to the individual needs of children by deleting records and/or replacing records to show different questions.

Factors
This file is the most complex of the numerical files provided with Fun School 3. It is suggested that this be used by children only after they have worked through the remaining numerical files. At the top of each record are listed, in order, all or some of the factors of a particular number. Below this is a statement which indicates how many factors are listed and the number to which they refer.

If children work through this file they should soon realise that some numbers have far too many factors to include on the top line of a record. You could then encourage them to list (on paper) the missing numbers. The file contains 53 records. You could suggest that the children extend the file to include factors of higher numbers.

Factors is suitable for use as a test file. It provides a very taxing test: A requisite number of factors must be entered in order separated by spaces.

The remaining three files each deal with different areas of the curriculum:

Shopping
This was briefly described earlier in this manual. An initial use of the file could be accompanied by work sheets to encourage searching the file for various pieces of information. It can also be used as an example of the collation of data from different sources which you could encourage children to extend, by way of their own surveys. Both of these are considered below, under *Further ideas*.

Europe
This file contains information about the capital cities, languages, currencies and populations of various European countries. You could use it as an educational resource (the test facility is especially useful here). As time proceeds, population statistics and other data will require updating.

Dates could act as a blue-print for similar files regarding countries of other continents.

Dates
This contains a selection of historic dates and can never be regarded as complete. Indeed, one person's idea of an important date may be another's idea of total irrelevance. DATES has been deliberately designed to be added to using dates covered at school. If you follow the current format it will form an instant test and hence revision aid, thus fulfilling the Government's current thinking regarding the study of History.

Further ideas

You can use Shopping to cover two National Curriculum ideas:

- To encourage children to search a database for information. The following are examples of the type of question that will encourage such searches. You could let children use them directly from this manual or as the basis of a work sheet:

1. Where can I buy the cheapest oranges?
2. Who sells the most expensive onions?
3. What is the cheapest way of buying frozen peas?
4. Work out how much 1 litre of shampoo would cost from the four retailers listed. Whose is the most expensive?
5. Whose eggs cost the least?
6. Mrs Brown's shopping list is: 1lb apples, a tin of baked beans, a loaf of bread, 250g of butter, 12 eggs, 1lb onions, 4 bars of soap, 400ml washing up liquid.

a. If she buys all these items from the supermarket, how much will she spend?
b. Could she save any money by purchasing any of the items elsewhere?

There are, of course, many other possible permutations of questions based on this and other files.

The National Curriculum in Information Technology states that pupils should be able to "*Insert and amend information in a computer database*" and "*use software...so that data can be captured, stored and retrieved*". This might sound a virtually impossible objective for young children (it is aimed at those under 11) but using *Shopping* as a template it could become a straightforward exercise:

- Help children select a number of common shopping items to survey. Keep the total number of items small at first, say to 10. The items can be chosen from those listed in the file, but they may of course select their own. Have them list these items in an exercise book leaving about half a page for each item.
- When you next go shopping allow them to start their survey. Under each item write down the name of the shop, the quantity that the item is sold in and its price. Let them repeat this process in a few more shops.
- After collecting three or more pieces of information for each item, encourage them to re-examine *Shopping*. Using the same format they should start their own database. It is important that you discourage children from amending *Shopping* as this will lead to a mixture of the original data and that collected by them: This will only lead to confusion!

- Children should save the data to their own disc or tape. The file can be built up using different items over a period of time. Each time a new record is added, the file should be sorted into alphabetical order.
- Once data for about 30 items (more if the children's interest is maintained) has been entered into the database, they could prepare a short report about local prices. Encourage them to consider looking at the cheapest way to shop and deciding how much could be saved when compared to the most expensive. If you have a printer, help them to print the file to include in the report.

Fun School's Database can be used for a wide variety of circumstances. The following are a few which you might wish to suggest:

- ***An address and phone number file.*** Place friends' and relatives' names on the top line of each record (surname first!) and other data below this.
- ***A birthday date reminder.*** The date should be placed onto the top line (ask children to think about different ways of entering this information: What is the best?). Names should be placed below this.
- ***A catalogue of books, computer software, LPs or videos at home.*** Encourage the children to decide upon the most appropriate format for the files. Usually they will decide to place the title in the top line.
- ***Production of quizzes.*** The questions could be based upon work covered in class (and therefore be used as revision aids) or produced just for fun.
- ***Spelling tests.*** The questions could consist of single definitions, the sound of a word or a deliberate misspelling.

If you do create any interesting files please send them to Database Educational Software – they may be good enough to be published and be worth money.

Sentences

Sentences provides a novel way for children to develop and improve essential basic skills that are, in many quarters, sadly neglected. Within the program, children are presented with a range of common errors which they have to identify and correct. The file provided with Fun School 3 contains 100 sentences, each with three or four errors. There are 25 sentences for each level of difficulty. During play, errors are selected at random. Because of this, each time that Sentences is loaded, apparently identical sentences will appear to contain different errors.

Children should be aware that, as errors are selected randomly, they will, on a few occasions, be presented with sentences devoid of mistakes.

Error types

Sentences categorises errors into four main groups:

- Punctuation (including the use of capital letters).
- Similar sounds.
- Spelling.
- Grammar.

The degree to which each of these have been used in the Sentence file reflects the frequency that such errors crop up in the written work of children in the 7 to 11 age range:

Punctuation

The most common punctuation errors include:

- Misuse of capital letters (eg: my friend john, London zoo, next saturday).
- Misuse of inverted commas (eg: do'nt go, Sams House, my brothers' birthday).
- Misuse of commas and full stops.

Other, less common, errors (including misuse of speech marks) are also included in the file. The majority of the questions in the file contain at least one punctuation error.

Spelling

Common spelling errors fall into a number of over-lapping categories, the most common of which are:

- Phonetic spelling (iland instead of island, vew instead of view).
- Swapping a/e (graet, seperate).
- Swapping of other vowels (mirrar, centery instead of century).
- Misuse of double consonants (droped instead of dropped, scarred instead of scared).

Less common errors are, again, included in the file.The majority of questions contain at least one spelling error.

Grammar

A number of common grammatical errors are included in the file and about 40% of the questions contain one or more.

Similar sounds

Children sometimes write down a word that sounds more or less the same as the word that they meant to write, for example: We live in Grate Britain. Many errors that appear

to fall into this category might also include other (spelling) mistakes. Because of this, only a few examples of such errors are included in the file, with about 10% of the questions containing such mistakes.

By pressing the options key you can display the error type menu. Normally, all four error types are switched on (this is indicated by a tick beside each name). They can, however, be turned off and on by selecting the appropriate option. When you have set the options you require, **select** OK to return to the game.

This option is particularly useful if you wish to confine the errors that children examine to spelling, punctuation or grammatical mistakes. As similar sounds are encountered infrequently, it is suggested that this option is avoided for use alone.

Creating new files

New Sentence files can be created using any word processor capable of producing Ascii files. The following is an example of the format which must be followed:

3: Stop at once<!/?/,> <Your/your> back [tyre/tire] (has/have/haves) gone {flat/flatt/falt}.

The 3 is the sentence level (ranging from 1 to 4). This must be followed by a colon. The remainder of the sentence is constructed from normal and bracketed words. The bracketed words represent the alternatives, the first being the correct version. The type of bracket may be used to indicate the type of error involved:

< > Punctuation [ ] Similar sounds { } Spelling ( ) Grammar

You may include as many (or as few) mistakes as you wish. There are no limits to the combinations of errors that can be included in any one sentence, or upon the number of alternatives for any one error. You should, however, be driven by common sense not to overload any one sentence and make it impossible to correct.

The maximum length for any sentence is 32 words, bracketed alternatives counting as a single word. Each sentence must be entered without any carriage returns in it. Because of this, most word processors will limit the total number of characters to 160. Each sentence should be separated by a carriage return. Files must be saved in pure Ascii format and have an extension .DOC if your computer filing system supports extensions.

Loading new files

Files can be loaded by pressing the load key from within the program. Enter the name of the file and press **Return**.

There are four levels of play in Sentences. In the file supplied, each level is successively more demanding, containing more complex errors.

Further ideas

- Encourage children to make a note of sentence errors which they cannot locate themselves, but have to use the Clue facility to find. Suggest that they compose their own sentences containing the corrected version.
- List spelling errors that children find problems and use these as the basis for a Word Search file.
- If you have a word processor, suggest that children help you to construct a series of sentences for use in the program. Some of the errors could be mistakes that have occurred in their school exercise books. Involve children as much as possible in constructing the sentences, giving you correct versions of each word or character as well as alternative incorrect versions.

Using tape to save and load data files

If you are unsure of the correct procedure to successfully load or save a data file when using tape, first consult the manual that came with your computer.

Loading files

Unlike the disc version, the tape version of Fun School 3 does not list the files when you wish to load one. Instead you will be prompted to enter the name of the file.

Saving files

Once again Fun School 3 will not list the files already on the tape. You will be prompted to enter a filename under which to save your data.

If you are unsure of the correct procedure to successfully save a data file to tape, first consult the manual that came with your computer. In addition you might like to bear in mind the following words of advice:

- Always use good quality tape – either specially manufactured for the purpose, or C30s. Avoid using C90 tapes as the thinner magnetic tape is not as reliable.
- Only save one data file on each tape. This may seem rather extravagant but in the long run it will save you time and hassle.
- Always make sure the tape is wound on past the 'leader' section before commencing the recording. Many a data file has been lost due to this one.
- Label the tape clearly with the data filename and the date of the recording.

Tape counter

For the convenience of tape users we have included the following chart to help you pinpoint the exact location of each game on your Fun School software. To use it, rewind each tape to the beginning and reset the counter on your cassette player to zero. As you come to the start of each program, make a note of the counter reading so that on a future occasion you can fast forward to it.

Program	*Counter*	*Side A/B*
Word Search **Robot Draw** **Planetary Maths** **Treasure Search** **Database** **Sentences**		
Data files		
RDMACROS *DBEUROPE* *DBFACTOR* *DBNUMBER* *DBSHOP* *DB4TAB* *DB6TAB* *DB8TAB* *DB12TAB* *DBTEST* *DBDATES* *WSDATES* *WSPLACES* *WSSPELL2* *WSSPELL3* *WSSPELL1*		

Note for Commodore Amiga users

If you have an unexpanded A500 you'll find that the Amiga will need resetting after quitting from any of the games. This is due to the way that the Amiga uses its memory and can be solved by purchasing a memory expansion board which slots easily into the underside of your computer.

Credits

Project Management team: Chris Payne, Peter Lee and Richard Vanner.
Atari ST version: Programmed in STOS Basic and compacted using STOS Squasher.
Programming: Darryl Kiley (Word Search, Planetary Maths), William Cochrane (Robot Draw, Database), David Hitchin (Treasure Search), Darren Ithell (Sentences), Nick Harper(Robot Draw).
BBC version: Converted by Silicon Genetics.
C64 version: Converted by John Kendal and Silicon Genetics.
CPC version: Converted by Joe Garner and Silicon Genetics.
PC version: Converted by Silicon Genetics.
PCW version: Converted by Chris Sadler.
Spectrum version: Converted by Mick Garlic, Stephen Nunn, Peter Farrington, Neil English, Martin King.
Amiga version: Converted from STOS to AMOS by William Cochrane and Peter Hickman.
Program Design: Peter Davidson and Chris Payne.
Packaging Design: Wendy Mellor.
Paste-up and typography: Visual Eyes, Stockport.
Instructions: Jon Revis, David Redpath, Alan McLachlan, Nick Harper.
Testing: Alex Blagg, Lee Fahy, Peter Lee, Richard Vanner,Nick Harper, Lance Concannon, Sally Meller and the children at Bebington House School in Chislehurst.