

ARROW  
USER GUIDE

PN 11646

Copyright (c) 1982 by Oxfordshire County Council  
Printed in Great Britain

All rights reserved. Copies of this publication may be made by customers exclusively for their own use, but otherwise no part of it may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language without the prior written permission of Research Machines Ltd.

The policy of Research Machines Ltd., is one of continuous development and improvement of its products and services, and the right is therefore reserved to revise this document or to make changes in the computer software it describes without notice. Research Machines Ltd., endeavour to ensure the accuracy of this document and to ensure that the products described perform correctly according to their descriptions. However, Oxfordshire County Council and Research Machines Ltd., do not accept liability for the consequences of any error or omission.

Additional copies of this publication may be ordered from Research Machines Ltd., at Research Machines Limited, PO Box 75, Oxford OX2 0BW.



## CONTENTS

INTRODUCTION	1
CHAPTER 1 GETTING STARTED	3
CHAPTER 2 FURTHER POSSIBILITIES	19
CHAPTER 3 INFORMATION ABOUT EACH COMMAND	21
CHAPTER 4 TECHNICAL NOTES	29
CHAPTER 5 ANSWERS	33
INDEX	39
KEYPAD OVERLAY	41



## INTRODUCTION

ARROW is a computer language. This means that it allows you to give instructions to the computer to do a certain job. As with all computer languages you have to give your instructions in a clear and precise way. The instructions that you can give are limited - you can't tell the computer to jump up and down, or to touch its toes.

The instructions, more usually referred to as commands, that you can use with ARROW are mainly to do with drawing on the screen or, if you have one, using a turtle to draw on the floor. The language is derived from LOGO, a more comprehensive language. ARROW is an attempt to present the possibilities for screen drawing in a straightforward way. It would be quite possible for users who become proficient with ARROW to progress to using LOGO.

As with LOGO, one of the most powerful features of ARROW is the ability to invent new commands from a combination of the existing commands, and then use these new commands in exactly the same way as the existing ones. It is possible to save these new commands on a cassette or a disc and use them again at a later date.

This booklet describes a version of ARROW designed to run on the following:

1. A cassette based Research Machines 480Z with at least 32K of memory and the High Resolution Graphics Option; you will also need BASIC in ROM.
2. A disc based Research Machines microcomputer system with High Resolution Graphics board and 56K of memory; you will need also BASICSG version 5.

The differences between cassette and disc use are given, where necessary, in the text.

## HOW TO USE THIS BOOKLET

This booklet comes in five sections. Chapter 1 is a step by step introduction to most of the features of ARROW, telling you what to type and what you can expect to see. Anything you need to type has been underlined. The chapter also includes some exercises. Chapter 2 gives some ideas for further directions. The third chapter of the booklet goes through each of the commands in alphabetical order and explains their use in full. Chapter 4 is a set of technical notes and a more detailed explanation of some of the error messages. The final chapter gives possible solutions to the exercises in chapter 1.

If you are a beginner it is recommended that you work through the whole of chapter 1, trying the exercises as you go. Answers are given in chapter 5,

although you should understand that they are not the only solutions, or even necessarily the best ones - they are there to help you if you've had any problems.

## CHAPTER 1

### GETTING STARTED

#### LOADING THE PROGRAM - INTO A LINK 480Z CASSETTE SYSTEM

Begin by switching on the 480Z microcomputer; you should see an introductory message on the display screen. Start the BASIC by typing:

R

Put the tape which contains ARROW into the cassette recorder with side A uppermost, Make sure the tape is fully rewound then type:

LOAD "ARROW" RETURN

and press the PLAY button on the cassette recorder. The screen should then display each block of the program as it is read in, as follows:

ARROW	0000		
ARROW	0001	or	Loading ARROW 0000
ARROW	0002		
ARROW	0003		

The block numbers change as each section of the program is read in from the cassette tape. When the program has finished loading, you should see the message :

Ready:

Now turn to page 4 to find out how to run the program.

#### LOADING THE PROGRAM - INTO A DISC BASED SYSTEM

Begin by switching on the microcomputer - you should see a message on the display screen. You will also need a disc with the operating system, CP/M, on it, the correct version of BASIC - BASICSG Version 5, and another disc with a copy of ARROW. Put the first disc in the top, or leftmost, drive and the other disc in the other drive (if your system has only one disc drive then you will need to change these discs after you have loaded BASIC - see below). Type B and then the following sequence (the messages may be slightly different) - the parts you type are underlined:

B  
Research Machines  
56K CP/M Version 2.2

A> BASICSG RETURN

There then follows a message from BASIC followed by

Ready:

LOAD "B:ARROW" RETURN

#### RUNNING THE PROGRAM

When you have successfully loaded the program, type:

RUN RETURN

A page of information about the program will then appear, together with an arrow in the middle of the screen. This is followed by a colon (:) at the bottom left hand corner of the screen. This is called the colon prompt. You can give any instruction that ARROW understands whenever you see the colon at the bottom left hand corner of the screen.

To start with you should try out the simple movement commands, which are FORWARD, BACKWARD, LEFT and RIGHT. Each command must be followed by a number, and don't forget to leave a space between the command and the number. For example, type :

FORWARD 40 RETURN

The screen will clear of everything except the arrow. A line will then be drawn in the direction of the arrow, 40 units long. As a rough guide, the screen is about 300 units wide and 200 units high. If you go off the edge of the screen the arrow is still there, but you won't be able to see it.

Now type :

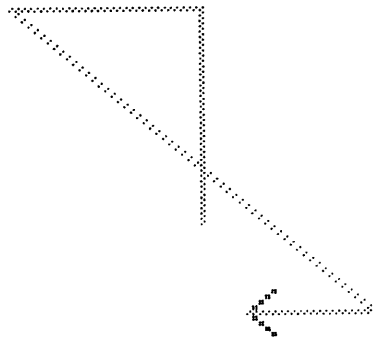
LEFT 90 RETURN

This will turn the arrow by 90 degrees (a quarter turn) to the left. Now type in the following - but remember to always wait for the colon prompt before typing the next instruction. (You should always press the RETURN key at the end of each line but it won't be shown from now.)

FORWARD 30  
RIGHT 45  
BACKWARD 80  
LEFT 45  
FORWARD 20



You should end up with the following picture :

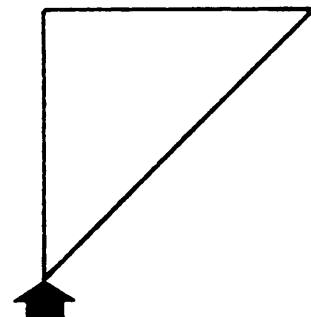
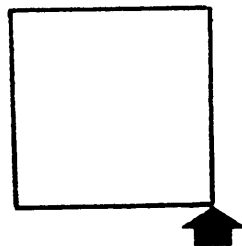
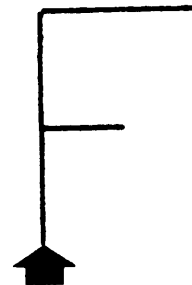
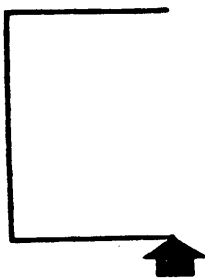


To get a clean screen and return the arrow to the centre at any time type :

FRESH

### EXERCISES

1. See if you can give the instructions to make each of the following shapes. The starting position of the arrow is shown each time. It is a good idea to write down your attempts before you actually type them into the computer.



2. Give the instructions for drawing a triangle whose sides are all 50 units long. You will find it helpful to draw a picture and measure all the angles.

3. Design your own shape, such as a dog or a hat - draw it out carefully on paper before you start writing down the instructions. Don't make it too complicated - if it has much more than ten sides, or any curved sides you may find it difficult.

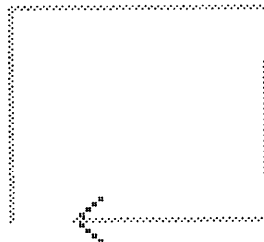
### Moving Without Drawing

So far you have only been able to draw a shape which is entirely connected together. If you want to draw two separate shapes, then you need to lift the pen off the paper and use the normal commands to get where you want to go, but without leaving a trace. You then put the pen back on the paper to carry on drawing.

Try the following :

```
FRESH  
FORWARD 40  
RIGHT 90  
FORWARD 40  
RIGHT 90  
PEN OFF  
FORWARD 10  
PEN ON  
FORWARD 30  
RIGHT 90  
FORWARD 30
```

You should now end up with the following picture :



### Repeating yourself

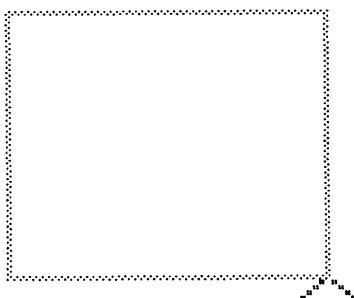
If you are drawing a square you will notice that you give the same two instructions, 4 times. An easier way of doing this is to use the REPEAT command. When you start the repeat you must say how many times you want to repeat the instructions. You then type the instructions that you want to repeat and when you have finished that list, you must then give the LIMIT instruction.

Now try the following:

```
FRESH  
REPEAT 4  
FORWARD 50  
LEFT 90  
LIMIT
```

You will notice that when you give the REPEAT instruction, it allows you to type your instructions to a slightly different way, and nothing happens until you give the LIMIT instruction. The computer then obeys the instructions in order, and shows you what it is doing at every stage.

You should end up with the following picture:



#### EXERCISES

4. Use the REPEAT instruction to draw a triangle where every side is 50 units long. You may have to experiment with the angle to decide how far to turn each time.
5. Use the REPEAT instruction to draw a rectangle which is 40 units high and 70 units long.
6. Choose one of the shapes that you drew in question 1, and use the Repeat instruction to draw five of them in a row. Remember to switch the pen off before moving to the start of the next shape.

#### Building you own instructions

So far, everything that you have done has produced a picture on the screen but as soon as you give the FRESH instruction your picture goes and the only way to get it back is to type in all the instructions again. However, there is a command that asks the computer to remember a set of instructions, and you can then use them as many times as you like, just like the other instructions which are there already. To ask the computer to remember a set of instructions you must give them a name, and it is sensible to call them something which reminds you what the instructions will do.

For example, we will ask the computer to remember the instructions to draw a square, and we will call them SQUARE. This is how to do it:

```
BUILD SQUARE  
REPEAT 4  
FORWARD 50  
LEFT 90  
LIMIT
```

You will notice that after the BUILD instruction the screen changes to be the same as when you gave a REPEAT instruction. However, this time when you type LIMIT, the computer is still waiting in case you want to give some more instructions. To tell it that you have finished you must press the

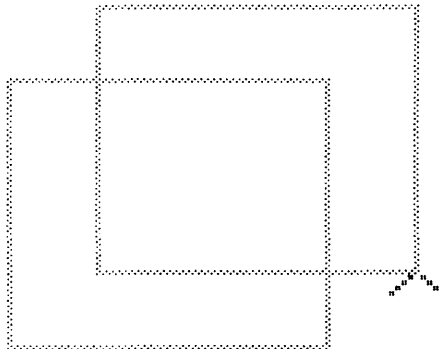
ESC key (which is usually at the top left hand corner of the keyboard)

Press the ESC key

Even then the computer will not actually draw the square until you give the SQUARE instruction. Now type the following:

```
FRESH  
SQUARE  
PEN OFF  
RIGHT 45  
FORWARD 20  
LEFT 45  
PEN ON  
SQUARE
```

You should now end up with the following picture:



A set of instructions which you build like this is called a procedure.

## EXERCISES

7. Use the BUILD instruction to store the instructions for the triangle that you drew in question 4. It will be helpful later if you call this TRIANGLE, and if it uses LEFT turns.
8. Use the BUILD instruction to store the instructions for drawing one of the shapes from question 1.
9. Use the BUILD instruction to store the instructions for drawing a row of five of the shape that you have just chosen for question 8. You are allowed to use the procedure that you built in question 8 inside your new procedure, instead of typing it in again.

## Changing things

It is possible to use your square and triangle together, to make a house. (This assumes that you answered question 7, and that you called you procedure TRIANGLE.)

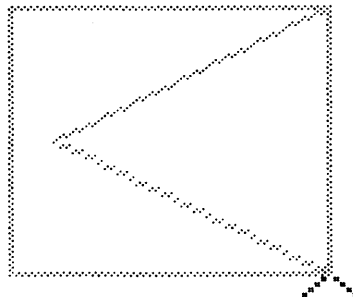
Now try typing the following:

```
BUILD HOUSE  
SQUARE  
TRIANGLE  
Press the ESC key
```

If you now type

```
FRESH  
HOUSE
```

you should end up with the following:



(your answer may be different, if you drew your triangle with LEFT instead of RIGHT turns.) Anyway, this hasn't drawn a house - it looks more like an envelope. You will need to change HOUSE, so that it draws the triangle at the top.

Now type the following:

CHANGE HOUSE

This puts your version of HOUSE on the screen, and lets you alter it. The screen should look like this:

```

                                HOUSE
_____
SQUARE
TRIANGLE
END
```

```

_____
Press U to move up, D to move down
    E to end,  R to remove a line
    S to alter stores used WITH procedure
    I to start inserting
```

You need to put in two extra lines between the SQUARE instruction and the TRIANGLE instruction. To do this you need to move the flashing mark to that position and then start inserting:

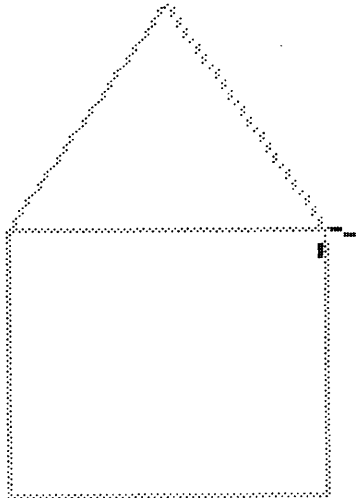
```

D to move down
I to start inserting
FORWARD 50
LEFT 30
press ESC to stop inserting
E to end the changes
```

If you now type

FRESH  
HOUSE

you should end up with the following:



#### Procedures that use numbers

So far, your procedures will always do exactly the same every time that you use them. However, we may want to make something a different size, or to have different number of sides, for example. You can do this by making your procedure behave in the same way as normal instructions. For example, you can build a curve so that it is whatever length you want, by typing the following:

BUILD CURVE WITH LENGTH, NUMBER

REPEAT NUMBER

FORWARD LENGTH

RIGHT 3

LIMIT

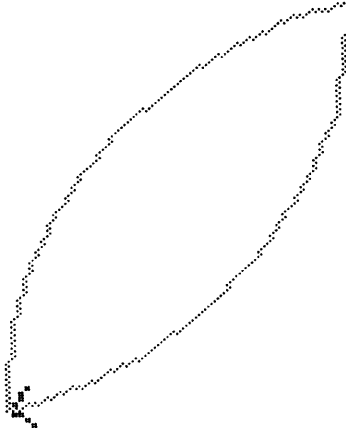
Press  to stop building

The WITH LENGTH, NUMBER part of the BUILD command means that whenever the instruction CURVE is used, two numbers will be expected. The first will be the length of each bit of the curve, and the second will be the number of these bits that make up the curve. We could have used any names for these two number, but it is sensible to use ones that remind you what they are. These are called stores, because they are each used to store a number until we want to use it again. Note that they only have an effect if you use them in the right way inside what you build.

If you now type

```
FRESH  
CURVE 4,25  
RIGHT 105  
CURVE 4,25
```

You should end up with the following:



It is also possible to change an existing procedure to use stores. For example, you can change your SQUARE procedures so that the sides can be any chosen length, by typing this:

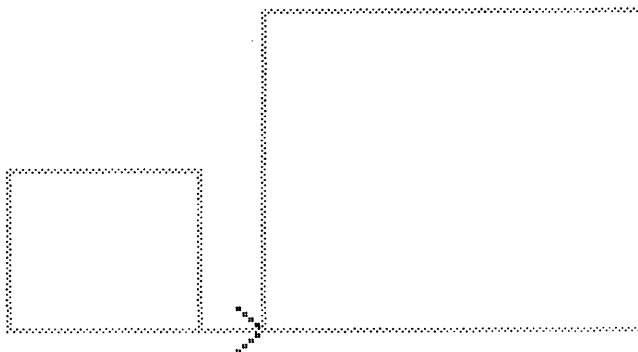
```
CHANGE SQUARE  
S to make SQUARE expect a number each time it is used.  
SIDE since we are going to call this number SIDE  
D to move down  
R to remove FORWARD 50  
I to start inserting  
FORWARD SIDE  
ESC to stop inserting  
E to end the changes
```

If you now type

```
FRESH  
SQUARE 30  
RIGHT 90  
FORWARD 10  
SQUARE 60
```



You should end up with the following:



### EXERCISES

10. Use the CHANGE command to change the instructions for the triangle that you built in question 7 so that it expects to be given the length of the side of the triangle.
11. Now change the HOUSE so that it expects to be given the length of the side of the square and the triangle. Try drawing different size houses on the screen.
12. Now build a new house, called HOME, which is made from a square of side 80 and a triangle roof of side 80. Then draw four squares of side 15 as windows inside the house. (You will need to move the pen backward before you start, or your home will not fit on the screen.)

### Doing arithmetic

It is possible to do simple arithmetic with the numbers in stores, using the MAKE command. Two examples of how to use this command now follow.

The following will draw an oblong of any height, and whose length will be 30 units more than the height.

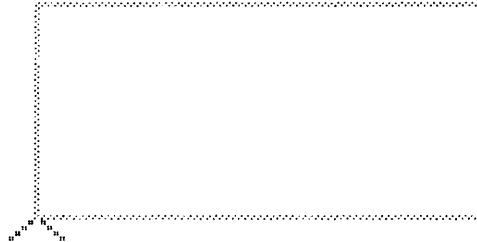
```
BUILD OBLONG WITH HEIGHT  
MAKE LENGTH BECOME HEIGHT + 30  
REPEAT 2  
FORWARDS HEIGHT  
RIGHT 90  
FORWARD LENGTH  
RIGHT 90  
LIMIT
```

Press ESC to stop building

If you now type:

```
FRESH  
OBLONG 40
```

you should end up with the following:



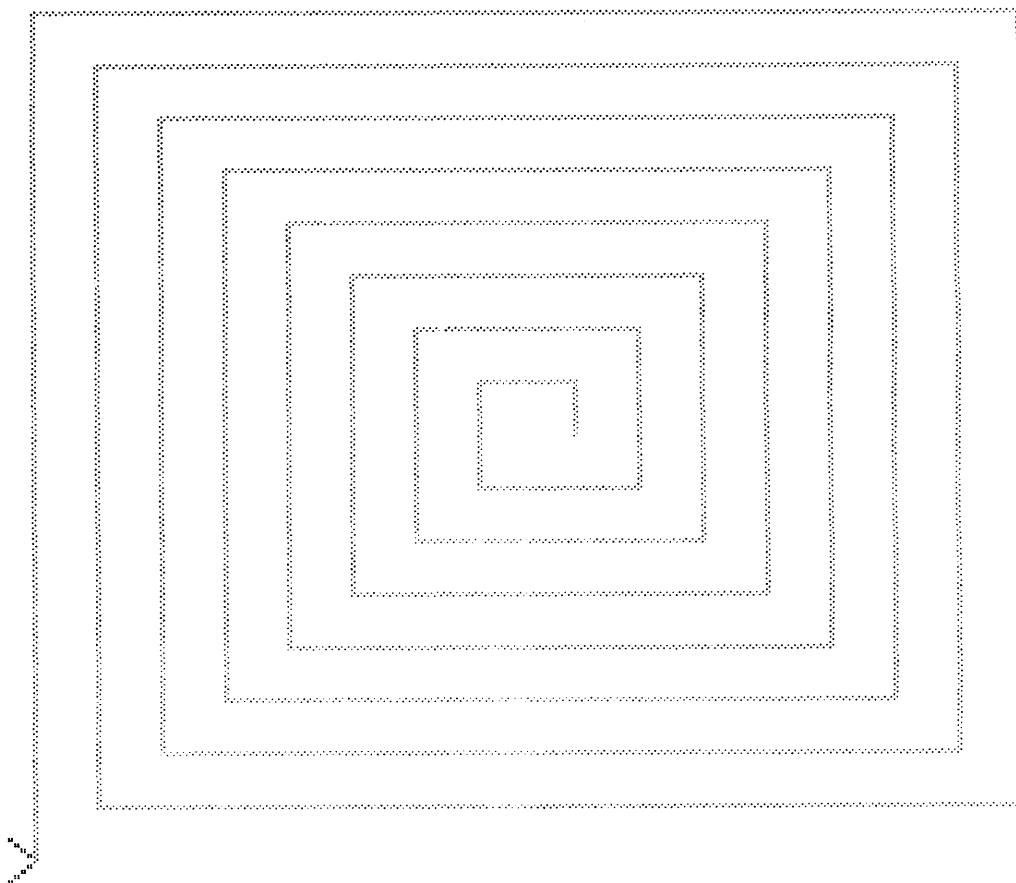
The program that follows shows another way of repeating something, using a command inside itself. This is called recursion.

```
BUILD SPIRAL WITH SIDE  
FORWARD SIDE  
LEFT 90  
MAKE SIDE BECOME 5 MORE  
SPIRAL SIDE  
Press ESC to stop building
```

If you now type:

```
FRESH  
SPIRAL 10
```

You should end up with something like the following - the actual number of sides will depend on the amount of memory in the computer.



In fact, this would never end unless you did something to stop it. However, the computer will not be able to do it for ever, so it will eventually stop and say:

Sorry, I can't do this any more.

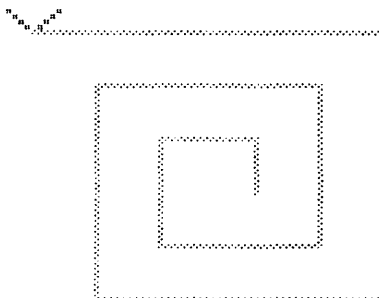
It is possible to avoid this problem by using the IF instruction. For example, to make sure that the spiral stopped just before the length of the side was 60, then type:

```
CHANGE SPIRAL
D to move down
D to move down
D to move down
R to remove SPIRAL SIDE
I to start inserting
IF SIDE < 60 THEN SPIRAL SIDE
press ESC to stop inserting
E to end the changes
```

If you now type:

FRESH  
SPIRAL 10

You should end up with the following:



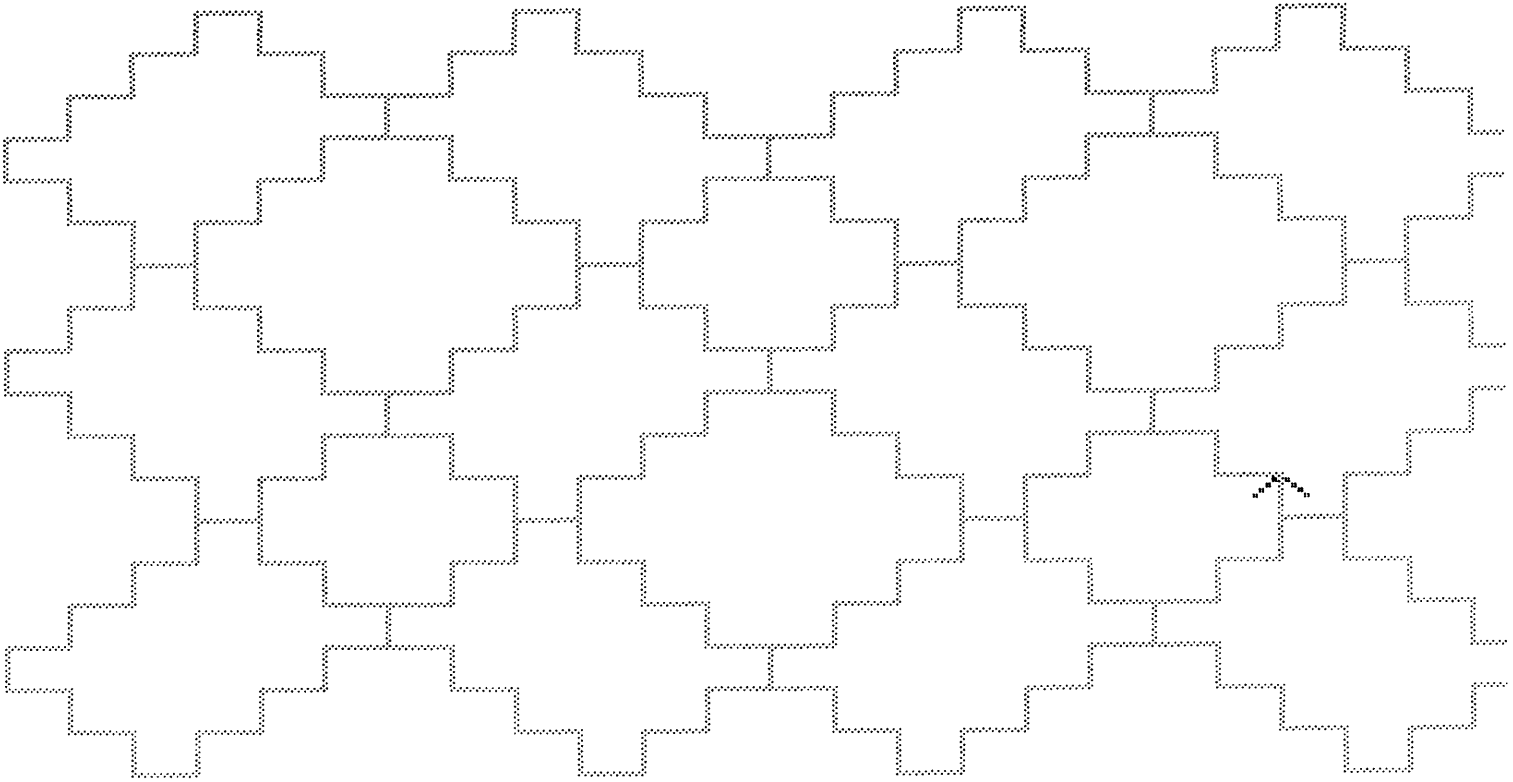
Finally, try this:

BUILD PATTERN WITH XMOVE,YMOVE  
MAKE X BECOME 0  
MAKE Y BECOME 0  
REPEAT 200  
FORWARD YMOVE  
MAKE Y BECOME YMOVE MORE  
IF Y > 70 THEN MAKE YMOVE BECOME - YMOVE  
IF Y < -70 THEN MAKE YMOVE BECOME -YMOVE  
RIGHT 90  
FORWARD XMOVE  
MAKE X BECOME XMOVE MORE  
IF X > 120 THEN MAKE XMOVE BECOME -XMOVE  
IF X < -120 THEN MAKE XMOVE BECOME -XMOVE  
LEFT 90  
LIMIT  
Press ESC to stop building

If you now type:

FRESH  
PATTERN 10,8

you should end up with the following:



### EXERCISES

13. Use the BUILD command to build a rectangle where the length is always twice the height. (Hint: to multiply, use \*).
14. Change SPIRAL so that the length of the side is halved each time round. (Hint: to divide, use /).
15. Build a pattern made up of squares. Each square should be half as big again as the one before, and placed so that it evenly surrounds it. There should be no connecting line between any of the squares.

### Saving what you do

If you want to save the procedures that you have built, then you can do so, either on cassette tape or on a disc if you have a suitable system. You can either save a particular procedure, or all the procedures. If you just type:

SAVE

then all your procedures will be saved together. On a disc system the procedures will be stored automatically on the disc you are using. On a cassette system you will be asked to put a tape into the cassette recorder, and then start it recording. Once it has started (allow a few seconds to make sure that you have passed the start of the tape), you should then press the RETURN key.

Put a tape into the cassette recorder.

Start recording

Press the RETURN key

You will be told when the recording is complete, and you can then stop the tape recorder, and remove the tape.

To get your procedures back, then type:

#### LOAD

On a disc system, the procedures will be automatically loaded from the disc you are using. On a cassette system you should put the cassette in the tape recorder, and press play. If your procedures are being loaded from cassette then you should see the following:

ALL.ARW	0000		
ALL.ARW	0001	or	Reading ALL.ARW 0000
ALL.ARW	0002		

(ALL.ARW is the name of the file containing all your procedures). When this has finished, you will be asked to wait while a list of all the procedures in the file is printed. When the colon prompt is given you will be able to use your procedures as before.

If you only want to save one procedure, then give the name of the procedure as part of the SAVE command. For example:

#### SAVE TRIANGLE

Filename shortened to TRIANG

You will notice that the name of the file can only have 6 letters at most. If your procedure has a longer name, then the filename will be shortened to the first six letters. When you load the file, then use the shortened filename:

#### LOAD TRIANG

TRIANG.ARW	0000	or	Reading TRIANG .ARW 0000
TRAIING.ARW	0001		

Once the file has been loaded you will still be able to use the full name TRIANGLE.

## CHAPTER 2

### SOME POSSIBLE FURTHER DIRECTIONS

This section contains some ideas for using ARROW. It should be clear by now that these only scratch the surface - new ideas are coming up all the time.

#### Golf

Mark out a simple golf course on a piece of plastic film, and then stick that on the television screen. It is a good idea to begin with a single 'hole' which you then have to give directions to get to, starting from the centre. See if you can do it in one shot. You can then make the course more complicated by putting in several holes, one after another, and some bunkers which you have to go round. See how few shots you take to go round the course.

#### A garden of flowers

You can use the procedure CURVE that you made in chapter 1 to build a petal. If you then put six petals together, along with a stalk, you can make a flower. You can then put several flowers together to make a garden. Try to plan your garden so that the flowers which are further away appear smaller.

#### Spirals

Try experimenting with different angles inside the spirals that you made in chapter 1. Instead of an endless spiral, try a spiral section with a chosen number of sides, a chosen angle and a chosen increase in the length of the side. If you then repeat this spiral section endlessly, it will sometimes go on for ever, and sometimes it will join back together and start all over again. Try to work out the relationship between the angle, the number of sides in the section and the amount the sides are increased by, so that repeating the spiral sections eventually join them up.

#### Hexagons

See if you can fill the whole screen with hexagons. Start by drawing a single hexagon, then a line of hexagons, and finally enough lines to fill the screen. To start with you should use a fixed size hexagon. When you have done that, try to make it work with the user able to choose the size of the hexagon. Try to experiment with other ways of filling the screen with hexagons - perhaps by drawing a spiral of hexagons?

### Letters and making words

In Exercise 1 of Chapter 1 you gave the instructions to draw some simple letters. Now try to build some more letter shapes. Make sure that they always start and end at the same place in relation to the letter. Now try to build a word from your letters.

### Tesselations

Draw a shape which tessellates - for example, a triangle, a diamond, a parallelogram, or a more irregular shape. (A shape that tessellates is one which will fill the screen without leaving any gaps - like the hexagon). Then try to fill the screen in the same way as with the hexagons.



## CHAPTER 3

### INFORMATION ABOUT EACH COMMAND

There are three types of instructions which can be given to the computer when the colon prompt (:) appears at the bottom left hand corner of the screen. The first type of instructions are called primitives and these do the actual screen drawing, etc.. Examples of primitives are FORWARD and REPEAT. The second type are called commands and do not affect the screen drawing directly. Examples of commands are HELP and CHANGE. The third type of instructions are called procedures and these only exist if you define them using the BUILD command, or if you load them from disc or cassette. An example of a procedure is SQUARE which was defined in Chapter 1. You can use primitives or other procedures inside a procedure, but you cannot use commands.

#### PRIMITIVES

**BACKWARD**      This moves the pen backward, in the opposite direction to the arrow. It must always be given with a number or a store name, which is the amount which is to be moved. The screen is approximately 300 units wide and 200 units high.

For example:              BACKWARD 30  
                              BACKWARD LENGTH

**CENTRE**        This moves the pen to the centre of the screen, without drawing and without changing the screen.

**CLEAN**        This cleans the screen, but leaves the pen in the same position as it was.

**END**            This instruction cannot be used on its own. It is assumed to be at the end of every procedure, but you should not type it in yourself. The END instruction can only be used as part of an IF instruction.

For example:              IF SIDE > 25 THEN END

**FORWARD**      This moves the pen forward, in the direction of the arrow. It must always be given with a number or a store name, which is the amount which is to be moved. The screen is approximately 300 units wide and 200 units high.

For example:              FORWARD 20  
                              FORWARD SIDE

**FRESH** This cleans the screen and moves the pen back to the centre of the screen. It is the same as the CLEAN and CENTRE instructions combined.

**IF** This allows you to make a choice of action. You can compare the value in a store with a number, or with the value in another store, using the following:

- = equal
- < less than
- > greater than
- <= less than or equal to
- >= greater than or equal to
- <> not equal

If the comparison is true, then the computer can do any primitive except REPEAT, LIMIT or another IF, or any procedure. It is also possible for the computer to do any primitive (except REPEAT, LIMIT or IF) or any procedure if the comparison is not true.

For example: IF SIDE < 25 THEN FORWARD SIDE  
IF LENGTH = 0 THEN END ELSE SPIRAL  
IF BIT <> 0 THEN LEFT BIT ELSE CENTRE  
IF SIDE > LENGTH THEN SQUARE 50

**LEFT** This changes the direction that the arrow points. It must always be given with a number or a store name, which is the amount that the arrow is to be turned left. The number is the amount the arrow turns, in degrees. Remember that a quarter turn is 90 degrees.

For example: LEFT 45  
LEFT ANGLE

**LIMIT** This instruction is only used after a REPEAT instruction. All the instructions between the REPEAT and the LIMIT will be repeated the number of times given in the REPEAT instruction.

**MAKE** This instruction is used to change the value of a store. It always starts with the name of the store whose value is to be changed. The next word must be BECOME. Following that must be one of these simple expressions:

- A number
- Another store
- Another store with a minus sign in front
- A number or a store followed by either MORE or LESS
- A number or a store followed by an operation and then another number or store. The operations can be
  - + to add, - to subtract,
  - \* to multiply and / to divide.

Note that any store mentioned after the BECOME must already have been given a value.

For example:   MAKE SIDE BECOME 30  
                  MAKE ANGLE BECOME 360 / NUMBER  
                  MAKE LENGTH BECOME BIT LESS  
                  MAKE WIDTH BECOME HEIGHT \* 2

**PEN**           This instruction must be followed by either the word ON or the word OFF. The PEN OFF instruction means that if the arrow is moved, no trace is left. Hence the normal FORWARD, BACKWARD, LEFT and RIGHT commands can be used to move the pen to any position without drawing a line. The PEN ON instruction means that from then on, moving the arrow will leave a trace as normal.

**REPEAT**       This instruction is used to repeat a set of instructions. It must be followed by a number or a store name, and the instructions after the REPEAT will be repeated this number of times. The LIMIT instruction is used to show the end of the list of instructions which are to be repeated.

For example:	REPEAT 4	REPEAT COUNT
	.	.
	.	.
	.	.
	LIMIT	.
		LIMIT

**RIGHT**       This changes the direction that the arrow points. It must always be given with a number or a store name, which is the amount that the arrow is to be turned right. The number is the amount the arrow turns, in degrees. Remember that a quarter turn is 90 degrees.

For example:       RIGHT 90  
                    RIGHT TURN

**SCALE**       Allows you to change the scale of the units on the screen. It is followed by either one or two numbers or stores. If there is one number then this applies to both the horizontal and vertical directions. If there are two numbers then the first number will affect the horizontal direction and the second will affect the vertical direction. If you want to return the scale in both directions to normal then give the instruction SCALE 1. Warning: if you give a scale of zero then you will stop all movement in that direction.

For example:       SCALE 3  
                    SCALE HEIGHT, WIDTH

## COMMANDS

### BUILD

This allows you to build a new procedure. The BUILD command should be followed by the name of the procedure. If you want to use one or more values each time you use the procedure, the next word should be WITH, followed by the names of the stores for the values you will type in each time you use the procedure. There is a maximum of 4 stores allowed with any procedure.

Once you have typed the BUILD instruction, the screen will change so that you can type in the procedure. You can use any mixture of primitives and procedures, including the procedure itself (this is called recursion), and procedures that you have not yet built. Obviously, you will have to then build those procedures before you actually use this one.

When you have finished typing in your procedure you should press ESC to end.

For example:

```
BUILD SHAPE
BUILD SQUARE WITH SIDE
BUILD SPIRAL WITH NUMBER, ANGLE, BIT
```

### CHANGE

This allows you to change a procedure that you have already built. It should be followed with the name of the procedure that you want to change. You must not include any store names which are used with the procedure.

The screen will change so that you can see the first 12 lines of your procedure. A flashing white dot will show which line you are about to change. You can use the U and D keys to move the flashing white dot up or down, to get to the place that you want to change. Once you have got to the correct place, then you can remove that line by pressing R. You can insert one or more lines (just in front of the line the white dot is pointing to), by pressing I. In that case you will be able to type in extra lines in the same way as when you build a procedure. When you have finished inserting at this point then press the ESC key, which will take you back to changing. You can then move the flashing dot to somewhere else.

If you want to change the stores which are used with the procedure, then press S. You will then be able to type in the names of stores that you want to use. If you just press RETURN without typing in a store name then when you use the procedure it will not expect any values at all.

When you have made all the changes that you want, press E to stop changing and go back to the colon prompt.

For example:

```
CHANGE SPIRAL
```

COMMANDS      This give a list of all the commands and primitives.

DELETE        This allows you to delete a procedure from the computer's memory. It should be followed by the name of the procedure that you want to delete. Once it has gone, the procedure cannot be recovered - you would have to type it in again. Deleting a procedure will not affect anything that you have already stored on tape or disc.

              For example:            DELETE SQUARE

DIR            For disc use ONLY.  
This gives a list of the procedures on the disc that you are using. The name of the file with all the procedures in will be called ALL.ARW.

FAST          This switches the drawing speed during procedures and REPEATs to be fast. The information about which instruction is being obeyed is not printed.

FINISH        This ends the program, and returns you to BASIC. If you want to use the program again, then type RUN.

HELP          This gives a brief message to remind you of what a particular command or primitive does. It should be followed by the name of the command or primitive that you want information about. If you want help with all the commands and primitives then you should type HELP ALL.

              For example:    HELP MAKE  
                                HELP TURTLE  
                                HELP ALL

LIST          This gives a list of all the current procedures. (IT does not include anything that you might have stored on tape or disc). If the command is followed by the name of a procedure then the actual procedure is listed.

              For example:    LIST  
                                LIST SPIRAL

LOAD          This allows you to load procedures that have already SAVED on tape or disc. If the command is followed by the name of a file, then the computer will look for the file containing that procedure. Filenames are always abbreviated to the first 6 letters.

              On a disc system the file will be loaded automatically.

              On a cassette system, you must put the tape containing the file in the cassette recorder and then press play. You will be told when the tape has loaded, and you can then switch off the cassette recorder.

For example:   LOAD  
                  LOAD SQUARE

PLIST           This prints a list of all the current procedures, on the printer. You should only use it if you have a printer connected. The printer option is set automatically by the program. However, your printer may use a different option to that expected by the program, and you should change it accordingly. See technical notes on page 30 for details of how to do this. If the command is followed by the name of a procedure then the actual procedure is printed.

For example:   PLIST  
                  PLIST OBLONG

RESTART        This command has the same effect as finishing the program and then running it again. It completely clears all your procedures and stores from the memory.

SAVE           This allows you to save procedures on tape or disc. If the command is given on its own, the computer will save all your procedures, in a file called ALL.ARW. If the command is followed by the name of a procedure, then the computer will just save that procedure in a file of that name, along with any other procedures that are mentioned in that procedure. If you want to save several procedures then you should give them in a list, after the command, separated by commas. In that case the file will be given the name of the first procedure in the list. Filenames are always abbreviated to the first 6 letters.

On a disc system the file will be saved automatically.

On a cassette system, you must put a blank tape in the cassette recorder and then start recording. Wait a few seconds for the tape to get past the start, and then press the RETURN key. You will be told when the procedures have been saved, and you can then switch off the cassette recorder.

For example:   SAVE  
                  SAVE HOUSE  
                  SAVE TRIANGLE, SPIRAL, CURVE

SLOW           This switches the drawing speed to SLOW. The information about which instruction is being obeyed will be printed each time, and there is a pause of at least a second between each instruction.

TURTLE         This allows you to connect a turtle. You should only use this command if you have a turtle connected. To start the turtle, then give the TURTLE ON command. To stop the turtle, give the TURTLE OFF command. While the turtle is any movement command (FORWARD, BACKWARD, LEFT and RIGHT) will appear on the screen as usual, and also be obeyed by the turtle. The turtle will also obey the PEN command. Please note that the turtle does

not obey the FRESH, CLEAN or CENTRE commands, and is not affected by the SCALE command.

#### VALUES

This gives a list of the stores that have been mentioned so far, and their current value. Note that if a store has not yet been given a value, its value will be blank, not zero.





## CHAPTER 4

### TECHNICAL NOTES

You are allowed to abbreviate any command, primitive or procedure to the first three or more letters. If you then want to follow the command with numbers or store names, then leave a space and continue as usual.

For example,   FOR 30   instead of   FORWARD 30  
                  REC 10,50   instead of   RECTANGLE 10,50  
                  BAC LENGTH   instead of   BACKWARD LENGTH  
                  VAL   instead of   VALUES

Please note that you are not allowed to abbreviate the names of stores. The store called LEN is quite different to the store called LENGTH.

#### Procedure names - Limitations

You are not allowed to use more than 12 letters for the name of a procedure. The name must only contain letters - you cannot use numbers, punctuation or spaces.

ARROW will not allow you to use certain names for procedures. If the computer tells you that a certain name is not allowed when you try to build a procedure, then don't worry - just choose a different name.

You are not allowed to use the name ALL for a procedure. The most common reason for not allowing a name is that it clashes with the first three letters of another command, primitive or procedure.

For example:   You cannot use the name LIMPET  
                  because it clashes with the primitive LIMIT  
  
                  You cannot use the name TURN  
                  because it clashes with the command TURTLE  
  
                  You cannot use the name PENTAGON  
                  because it clashes with the primitive PEN

You may have noticed that if you mention a procedure before you have built it, then you are normally given the message "Warning: I don't understand NAME yet". However, if that name is not allowed for a procedure then you will be told that the computer doesn't understand what you have just said.

#### Store names - limitations

You are not allowed to use more than 12 letters for the name of a store. The name must only contain letters - you cannot use numbers, punctuation or spaces.

The computer will not allow you to use the word ELSE as a store name.

### Changing speed when drawing

The FAST and SLOW commands set what the drawing speed will be when you the computer is obeying a procedure or a REPEAT. However, you may want to change the speed once the procedure or REPEAT has started. To do this, press the F key to make the drawing fast, or the S key to make the drawing slow. You can do this as many times as you like while the computer is actually doing the drawing. Once you get the colon prompt, then you must give the command in full.

### Stop drawing

You can stop any drawing at any time by pressing the colon (:) key. To help you remember, think about pressing the colon key to get back to the colon prompt. Once you have stopped the drawing, it is not possible to continue the procedure from where you left off - you will have to start again.

### Mentioning procedures before you build them

When you are building or changing a procedure, you are allowed to mention other procedures before you build them. IF you do, then you will have to build these before you use any procedure in which you mention them.

There is a possible confusion about the names of these procedures, because of abbreviations. When you mention the procedure before you build it, you are allowed to use abbreviations. So, if you intend to build HOUSE later on, you can still call it HOU when you first mention it, and this will be changed when you build HOUSE itself. However, the name that you build is taken to be the correct name, and cannot be an abbreviation. So, if you refer to HOUSEBOAT before you build HOUSE, the computer will not then call that procedure HOUSEBOAT. Also, you will not be able to build a procedure called HOUSEBOAT once you have built one called HOUSE. The only solution will be to change any reference to HOUSEBOAT to something else

### Printers

If you use the PLIST command you will need to have a printer connected and it is necessary also to tell the computer what type of printer you have. ARROW assumes that you have a particular type of printer and tells the computer. This is printer type 4, at speed 6 - ie, PRINTER 4,6. If you have a different printer, you will have to change your version of ARROW. (It is no use telling the computer yourself what type of printer you have because ARROW will tell the computer about its printer after you do.) To change the printer, you will need to change line 2 of the program as follows. This example supposes that you have printer type 3.

Load the program as usual. Do NOT type RUN.

Type        2 PRINTER 3

On a 480Z you will need to put a blank tape into the cassette recorder, and start recording.

SAVE "ARROW"

### Connecting a TURTLE

If you have a turtle, then it is normally connected to the SIO-4 socket, and runs at 1200 baud (speed code 3). If this is not the case, then you can change ARROW in a similar way to changing the printer speed (see above). The only difference is that you will have to change line 4 of the program, as follows - suppose that the turtle is to be connected to the SIO-2 at 300 baud (speed code 1)

Load the program as usual. Do NOT type RUN.

Type        4 PRINTER 2,1

On a 480Z you will need to put a blank tape into the cassette recorder, and start recording.

SAVE "ARROW"

If at any stage the turtle does not appear to be connected to the computer, then if the TURTLE has been switched on and a movement instruction is given, a warning message will be given and the turtle will be switched off. Please note that the computer will take about 20 seconds to decide that the turtle is not properly connected.

### Arrow and function keys

LINK 480Z only

The small keypad at the right of the 480Z can be used for some of the instructions, instead of typing them in. If you press any of the four arrow keys, then one of the words FORWARD, BACKWARD, LEFT and RIGHT are typed in for you. You will still have to type in how far you want to go. The key F1 will give the FRESH command; the key F3 will give the REPEAT instruction (you will have to type in how many times you want to repeat), and the F4 key will give the LIMIT instruction.

A cut out reminder of what each of these keys do is printed at the back of this booklet; you might like to cut it out, stick it on a piece of card, and fit it over the keypad.

### Errors

If you make a mistake you will usually be given a helpful message. Once the computer has told you of your mistake, it will then ignore what you

have just done, and allow you to try again.

Do not be afraid to make mistakes - it is a good way to learn

Sometimes the message from the computer may not make sense - this happens because the computer hasn't understood you, and has tried to guess what you mean - sometimes it will guess wrong. It may be helpful to realise that the computer starts at the beginning of a command and if it is correct works through the command until it meets something wrong. It then assumes that it is this which is wrong, even if most of the rest of the command obviously meant something else.

For example      if you type MAKE SQUARE WITH SIDE  
                         you probably meant to say BUILD instead of MAKE.

                         However, the computer will assume that MAKE  
                         is correct, and that you want to change a store  
                         called SQUARE. It will then expect to find the word  
                         BECOME and will give the message:  
                         Should say BECOME, not WITH

If the computer's message isn't helpful, then just think about what you should have said and then type that instead - don't worry too much about what the computer thought you were trying to say.

Some of the messages need a little more explanation:

Sorry, there's no room left

This can happen for several reasons. You may have too many procedures; you may have too many instructions inside your procedures; you may have too many variables. If it is either of the first two reasons, then you will find that you cannot give any primitive or procedure as an instruction. One solution may be to DELETE a procedure that you no longer need. Otherwise, you can SAVE some of the procedures, (don't SAVE them all at once or you will have the same problem when you LOAD them again.) You can then clear the memory with the RESTART command, and LOAD those procedures that you want.

Sorry, I can't do this any more

This happens if you use recursion that doesn't end, or if you use REPEAT inside REPEAT a lot of times. The computer will stop drawing and return to the colon prompt. It is caused by the computer running out of room and there is no way of overcoming the problem, except to not ask the computer to do quite as much.

## CHAPTER 5

### ANSWERS

Please note that these are not the only possible answers, or even the best answer. They are simply to give you an idea if you are stuck. It should be possible to type these answers in directly. In the case of a procedure there is usually an example of how to use it.

1.           FRESH  
              LEFT 90  
              FORWARD 30  
              RIGHT 90  
              FORWARD 40  
              RIGHT 90  
              FORWARD 30

              FRESH  
              FORWARD 40  
              LEFT 90  
              FORWARD 15  
              BACKWARD 30

              FRESH  
              FORWARD 20  
              RIGHT 90  
              FORWARD 15  
              BACKWARD 15  
              LEFT 90  
              FORWARD 20  
              RIGHT 90  
              FORWARD 30

              FRESH  
              FORWARD 40  
              LEFT 90  
              FORWARD 40  
              LEFT 90  
              FORWARD 40  
              LEFT 90  
              FORWARD 40

              FRESH  
              FORWARD 40  
              RIGHT 90  
              FORWARD 40  
              RIGHT 135  
              FORWARD 57

2.        FRESH  
          FORWARD 50  
          RIGHT 120  
          FORWARD 50  
          RIGHT 120  
          FORWARD 50
3.        FRESH  
          FORWARD 50  
          LEFT 120  
          FORWARD 30  
          RIGHT 150  
          FORWARD 62  
          RIGHT 120  
          FORWARD 62  
          RIGHT 150  
          FORWARD 30  
          LEFT 120  
          FORWARD 50  
          RIGHT 90  
          FORWARD 10
4.        FRESH  
          REPEAT 3  
          FORWARD 50  
          RIGHT 120  
          LIMIT
5.        FRESH  
          REPEAT 2  
          FORWARD 40  
          RIGHT 90  
          FORWARD 70  
          RIGHT 90  
          LIMIT
6.        FRESH  
          PEN OFF  
          LEFT 90  
          FORWARD 100  
          RIGHT 90  
          PEN ON
- REPEAT 5  
          FORWARD 40  
          LEFT 90  
          FORWARD 15  
          BACKWARD 30  
          PEN OFF  
          LEFT 90

FORWARD 40  
LEFT 90  
FORWARD30  
LEFT 90  
PEN OFF  
LIMIT

7. BUILD TRIANGLE  
REPEAT 3  
FORWARD 50  
LEFT 120  
LIMIT  
Press ESC to stop building

FRESH  
TRIANGLE

8. BUILD SHAPE  
FORWARD 20  
RIGHT 90  
FORWARD 15  
BACKWARD 15  
LEFT 90  
FORWARD 20  
RIGHT 90  
FORWARD 30  
Press ESC to stop building

FRESH  
SHAPE

9. BUILD ROW  
REPEAT5  
SHAPE  
PEN OFF  
RIGHT 90  
FORWARD 40  
LEFT 90  
FORWARD 20  
LEFT 90  
PEN ON  
LIMIT  
Press ESC to stop building

FRESH  
PEN OFF  
LEFT 90  
FORWARD 100  
RIGHT 90  
PEN ON  
ROW

10. CHANGE TRIANGLE  
S  
SIDE  
D  
R  
I  
FORWARD SIDE  
Press ESC to stop inserting  
E  
  
FRESH  
TRIANGLE 40
11. CHANGE HOUSE  
S  
SIDE  
R  
R  
I  
SQUARE SIDE  
FORWARD SIDE  
Press ESC to stop inserting  
D  
R  
I  
TRIANGLE SIDE  
Press ESC to stop inserting  
E  
  
FRESH  
HOUSE 30
12. BUILD HOME  
SQUARE 80  
FORWARD 80  
LEFT 30  
TRIANGLE 80  
LEFT 150  
FORWARD 15  
RIGHT 90  
PEN OFF  
FORWARD 15  
PEN ON  
REPEAT 4  
SQUARE 15  
PEN OFF  
FORWARD 50  
LEFT 90  
PEN ON  
LIMIT  
Press ESC to stop building



FRESH  
PEN OFF  
BACK 60  
PEN ON  
HOME

13. BUILD RECTANGLE WITH HEIGHT  
MAKE LENGTH BECOME HEIGHT \* 2  
REPPEAT 2  
FORWARD HEIGHT  
RIGHT 90  
FORWARD LENGTH  
RIGHT 90  
LIMIT  
Press ESC to stop building

FRESH  
RECTANGLE 40

14. CHANGE SPIRAL  
D  
D  
R  
I  
MAKE SIDE BECOME SIDE/2  
Press ESC to stop inserting  
R  
I  
SPIRAL SIDE  
Press ESC to stop inserting  
E

FRESH  
SPIRAL 80

15. BUILD EXPAND WITH SIDE  
MAKE MOVE BECOME SIDE/4  
SQUARE SIDE  
PEN OFF  
RIGHT 90  
FORWARD MOVE  
LEFT 90  
BACKWARD MOVE  
PEN ON  
MAKE SIDE BECOME SIDE \* 1.5  
EXPAND SIDE  
Press ESC to stop building

FRESH  
EXPAND 8



## INDEX

Abbreviations	29, 30
ALL	18, 25, 26, 29
Arithmetic	13, 22
Arrow and function keys	31
BACKWARD	21
BECOME	13, 23
BUILD	7, 24
CENTRE	21
Cassette systems	1, 3, 18, 25, 26
CHANGE	10, 24
CLEAN	21
Commands	4, 21, 24
COMMANDS	25
DELETE	25, 32
DIR	25
Disc systems	1, 3, 18, 25, 26
EDITING	7, 10, 24
END	21
FAST	25, 30
FINISH	25
FORWARD	4, 21
FRESH	5, 22
HELP	25
IF	15, 22
Instructions	1, 21
LEFT	4, 22
LIMIT	6, 22
LIST	25
Loading ARROW procedures	18, 25
LOAD	18, 25
MAKE	13, 22
PEN	6, 23
PLIST	26, 30
Primitive	21
Printer	26, 30
Procedure	8, 21, 24
Procedure names - limitations	29
Recursion	14, 24

REPEAT	6, 23
RESTART	26, 32
RIGHT	23
Room - none left	15, 32
RUN	4, 25
Saving ARROW procedures	17, 26
SAVE	17, 18, 26
SCALE	23
SLOW	26, 30
Speed of drawing	30
Stopped drawing	30
Stores	11, 13, 22
Store names - limitations	29
TURTLE	26, 31
Turtle - connecting it	26, 31
VALUES	27
WITH	24

