Jonathan Inglis

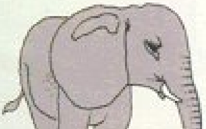
BEGINNERS' MICRO GUIDES

Commodore 16

BEGINNERS' MICRO GUIDE

Commodore 16

By Jonathan Inglis
Illustrations by Mike Atkinson and Janos Marffy
Photography by Mark Gatehouse



GRANADA

Published by Granada Publishing 1985
Granada Publishing
8 Grafton Street, London W1X 3LA

British Library Cataloguing in Publication Data
Inglis, Jonathan
Commodore 16. – (Beginners' micro guides 5)
1. Commodore C16 (Computer)
I. Title   II.Series
001.64'04   QA76.8.C64

ISBN 0-246-12680-9

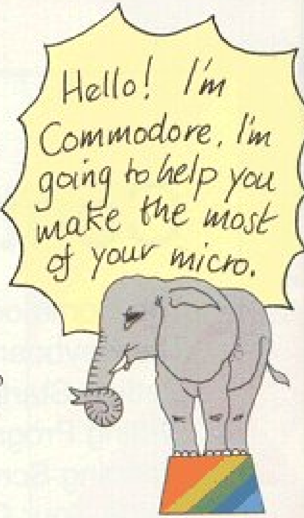
Printed in Italy by
New Interlitho, Milan

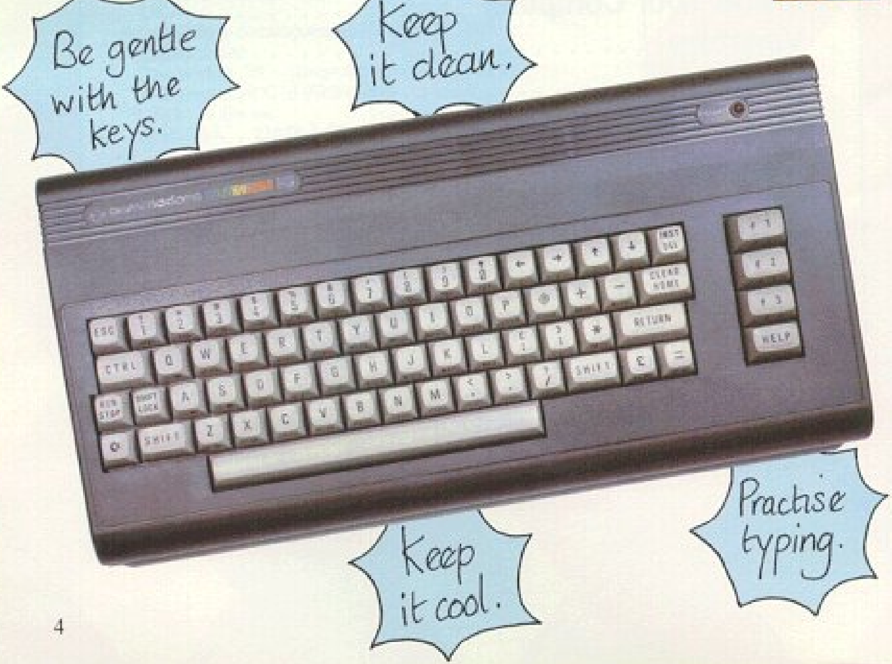
Contents

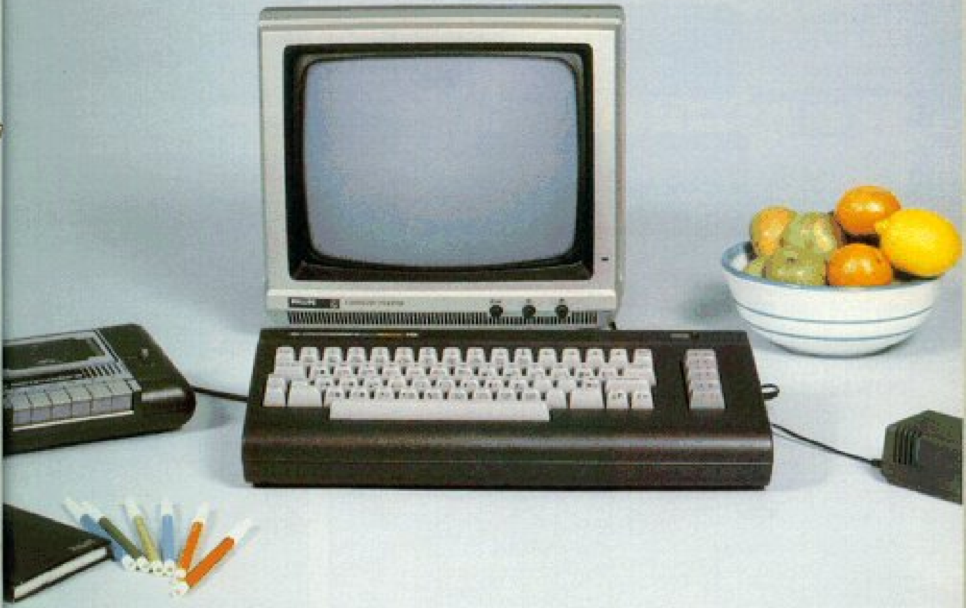
The Commodore 16

The Commodore 16 is an exciting machine. You can play games on it, or use it like a notebook to store information. It can draw multi-coloured patterns and play music. This book will teach you how to make your Commodore 16 do many jobs. There is a lot to learn; you may find it slow going at first, but don't be in too much of a hurry. A computer is a complicated machine. It takes skill and patience to use it properly.

It also needs care. The computer is delicate, so treat it carefully. Don't hammer the keys, and don't be tempted to take the lid off just to look inside! Later in this book we'll show you what's inside.


Hello! I'm Commodore. I'm going to help you make the most of your micro.





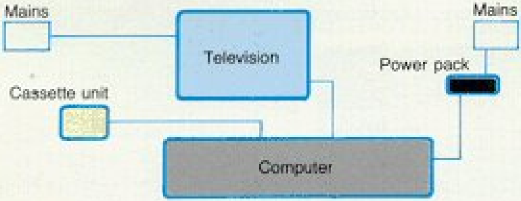
The computer system

The equipment in a computer system is called the HARDWARE. The picture shows three pieces of hardware.

1. The MICROCOMPUTER itself, which is usually just called the COMPUTER, and its power pack.
2. A television, which is sometimes called a VDU or VISUAL DISPLAY UNIT. You can use either a TV or a specially designed MONITOR.
3. A CASSETTE RECORDER, which stores any programs you may want to keep. This recorder is specially designed for using with the Commodore 16.

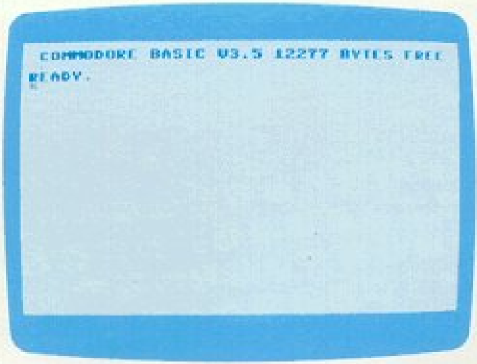
You can add other items of hardware, too, such as a printer.

Setting up



Your Commodore 16 comes supplied with a separate power pack. Get an adult to put a plug on the power pack lead. Connect the plug to the mains. Plug the other end of the lead into the socket marked POWER, on the right hand side of the computer. Next to this socket is a switch. Switch it to the ON position. The red light marked POWER, on top of the computer, should now light up.

Plug your television into the mains. Now connect the smaller end of the aerial lead to the socket marked RF on the computer, and the larger end to the aerial socket on your television. If you have a monitor, the monitor lead should be connected to the socket marked VIDEO. Switch on your television and adjust the tuning control until you can see this message:

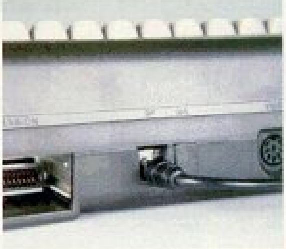


```
 COMMODORE BASIC V3.5 12277 BYTES FREE
READY.
```

Adjust the colour and contrast controls so that the message is clear and sharp. Remember, you are going to spend a lot of time staring at the screen and it isn't good for your eyes if the picture is too bright.

Power pack

The POWER light tells you if the computer is on.



Aerial lead

Don't worry if it takes time to tune your TV. You'll get there in the end.

What is a computer?

MORFAX

The message on the screen tells you what version of BASIC your Commodore 16 uses (Commodore Basic V3.5). It also tells you how much space there is for your programs in the computer's memory. With the Commodore 16 this is 12277 Bytes, or just under 12K of memory.

Your computer is more than just an electronic typewriter. It works by obeying sets of instructions. These can be written in BASIC, a special language which the computer obeys, but which is also easy enough for beginners to understand.

A set of BASIC instructions is called a PROGRAM. Putting programs into your computer is called PROGRAMMING.

Inside your computer are all sorts of microchips and electronic circuits. Some of these translate the instructions you give into coded electronic signals. Each word or instruction in a program has its own code. These are stored in chips which form the computer's MEMORY. Once a program is in the memory, it can be used over and over again.

```
10 LET Z=2.25
20 LET Y=7.575
30 PRINT Z
40 PRINT Y
50 PRINT Z+Y
```

A computer program is a list of instructions placed in numbered lines.

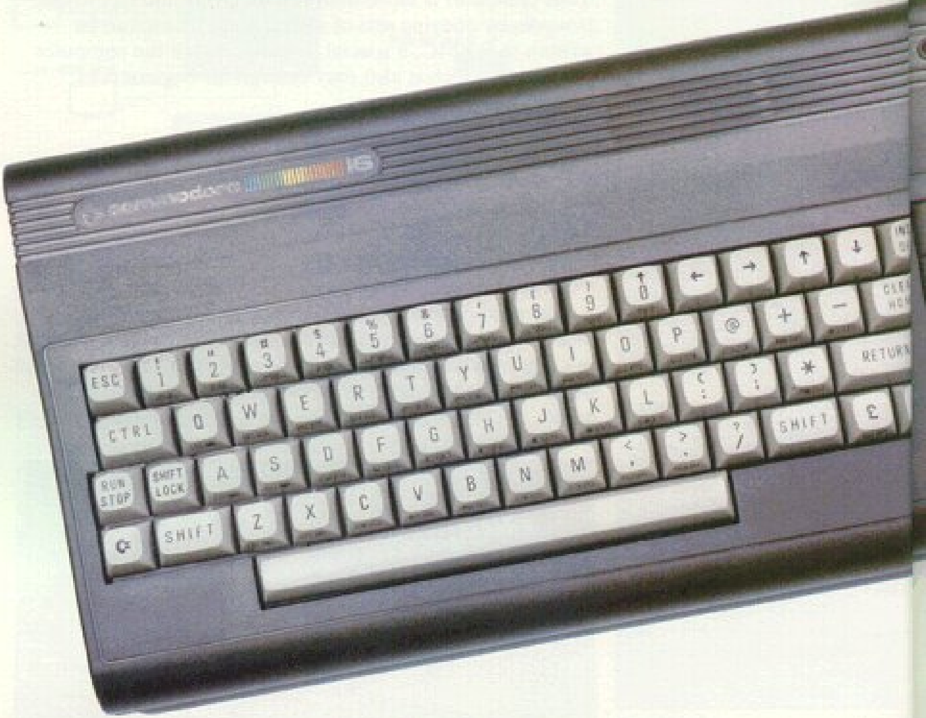
MORFAX

You can buy ready-made programs in a number of different forms. The most common are on cassette tapes, but if you have a disk drive you can buy software on floppy disks too.

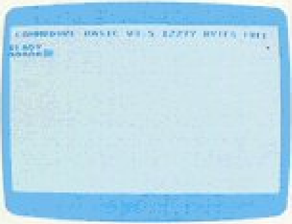
When you first switch on your computer, it cannot yet do very much. It needs programming. You can buy ready-made programs on tape, and feed them in using the cassette recorder. These tapes and the programs on them are known as SOFTWARE.

Using ready-made software is fun, but not as interesting as writing your own programs. To do this you type the programs in directly, using the keyboard of your computer.

The Keyboard



Your Commodore 16 has four rows of keys, and a long key at the bottom. This is called the KEYBOARD.



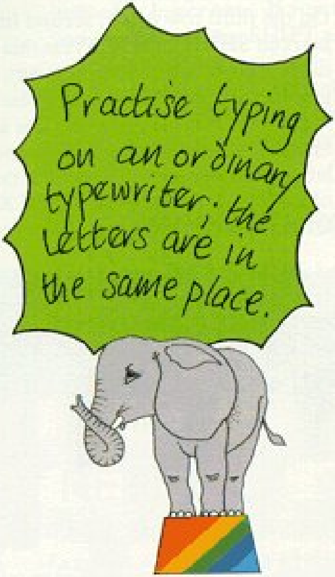
```
COMMODORE BASIC V3.5 12277 BYTES FREE
READY.
```

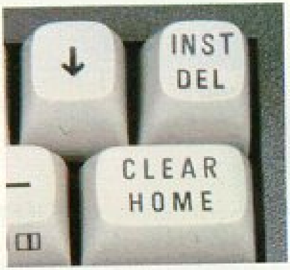
Now look at the screen. On it you will see a flashing block. This is known as the CURSOR. Anything you type in will be printed where the cursor is. Press one of the letter or number keys. That letter or number will be printed on the screen. If you keep your finger on the key, it will be printed again and again until you take your finger off.

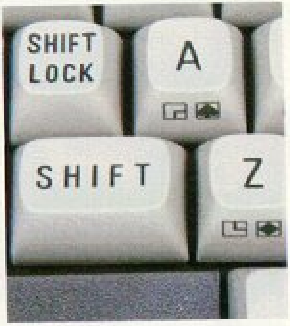
Type a few letters. If you want to leave a space between two letters then press the SPACE BAR (the long key at the bottom of the keyboard).

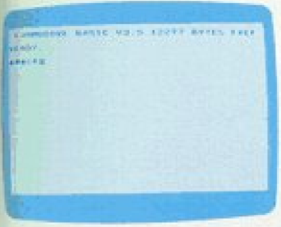
Rubbing out

At the top right hand corner of the keyboard is a key marked INST DEL. When you press this it rubs out or 'deletes' the letter to the left of the cursor. Keep your finger on this key and it rubs out everything you have typed in, though you cannot delete the cursor.


Practise typing on an ordinary typewriter; the letters are in the same place.





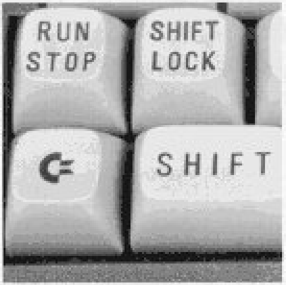


```
COMMODORE BASIC V3.5 12277 BYTES FREE
READY.
[illegible]
```

Keyboard control

The SHIFT keys

On the bottom row of the keyboard are two keys marked SHIFT. Press one of these. Keeping your finger on it, press one of the letter keys. On your screen you will see a GRAPHICS SHAPE or GRAPHICS CHARACTER. The shape is the right hand one printed on the front of the key you have pressed. Try this with several letter keys. If you press SHIFT and a number key, the symbol printed above that number will appear on the screen.



The Commodore key
In the bottom left hand corner of the keyboard is a key with the Commodore trademark printed on it. Press this key and, keeping your finger on it, press any of the letter keys. This prints the left hand graphics shape shown on that key.

The SHIFT LOCK key
On the third row down, second from the left is a key marked SHIFT LOCK. Press this down and it locks (stays down). Now, when you press any of the letter keys, the graphics shapes are printed. Press SHIFT LOCK again and it unlocks. It is easy to press this key by mistake. If you find that you are typing shapes and symbols instead of letters and numbers, check that the SHIFT LOCK key is off. Also, when the SHIFT LOCK key is on, the INST DEL key won't work. So don't leave it on by mistake.

REMEMBER! The INST DEL key won't work if the SHIFT LOCK key is on.

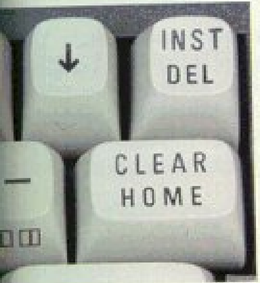
Little and LARGE letters

Press the Commodore and SHIFT keys at the same time. You will see that this changes all the capital letters on the screen to little (lower case) letters. It also changes graphics shapes to capital letters. Numbers and symbols stay the same.

When you type in more letters, they will automatically be printed in lower case. If you want to print a capital letter, you must now press SHIFT and the letter you want.

For the moment, just use capital letters. Press the Commodore and SHIFT keys at the same time, to make all the small letters capitals again.

Clearing the screen

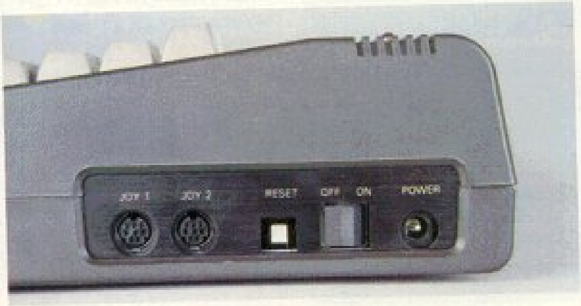


Just below the INST DEL key is a key marked CLEAR HOME. To clear the screen, press this key and the SHIFT key at the same time.

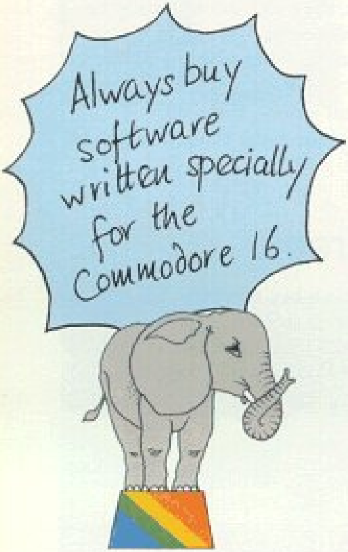
The screen will clear, and the cursor can then be seen in the top left hand corner.

RESET

If you find that your Commodore 16 isn't behaving as you expect, you may need to clear its memory. To do this, press in the white button marked RESET (next to the ON/OFF switch on the side of the computer). Always do this before loading in a game.



Getting Started



Now that you can use the keyboard, you can begin to give your Commodore 16 instructions. But you can't just type in anything. What you type in must make sense to the computer.

Your Commodore 16 has been programmed to recognize a language called BASIC. If you want to learn how to use your computer, you will have to learn about BASIC too.

BASIC is made up of words, numbers and symbols. You must be very careful how you write in BASIC. Computers need precise instructions and there are strict rules that you must follow, called 'syntax'. A mistake which breaks these rules is called a SYNTAX ERROR. Typing in a wrong letter or symbol can stop a program from working.

Basic buying

Most home computers use BASIC. Unfortunately they don't all use the same version of BASIC. You might think that a program which works on other computers will work on the Commodore 16.

BE WARNED: IT MAY NOT!

If you are copying out a program, first make sure that it will work on the Commodore 16. If you don't, you could be in for a disappointment.

If you buy tapes, make sure that they are for the Commodore 16. Atari, BBC Micro, Sinclair Spectrum and even Commodore 64 software will not work. Software can only work on the computers for which it has been written.

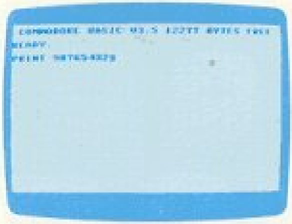
MORFAX

Commodore 16 BASIC is known as Commodore BASIC 3.5. It offers many new graphics and sound commands which were not in the earlier versions of BASIC used by the VIC 20 and Commodore 64 machines. These commands have been added to make the Commodore 16 simpler for beginners.

PRINT

BASIC uses many words that you will recognize. The first BASIC word to look at is PRINT. To start with, type in:

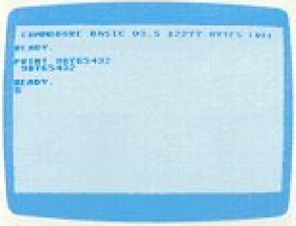
```
    PRINT 98765432
```



```
 COMMODORE BASIC V3.5 12277 BYTES FREE
READY.
PRINT 98765432
```

This instructs your Commodore 16 to print the number 98765432 on the screen. Of course you can already see that number on your screen, but that's because you've typed it in. What you want is to see your Commodore 16 printing the number.

To do this you have to use another key, the RETURN key. This is on the right hand side of the keyboard. Press this now and you will see the number 98765432 printed on the screen.

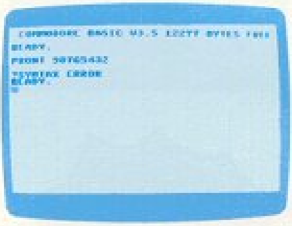


```
 COMMODORE BASIC V3.5 12277 BYTES FREE
READY.
PRINT 98765432
 98765432

READY.
```

Report messages

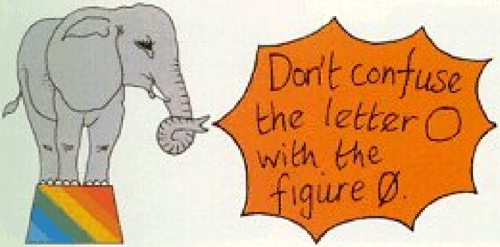
If your Commodore 16 has carried out its instructions successfully, you will see the word READY on your screen. This is shown as a REPORT.

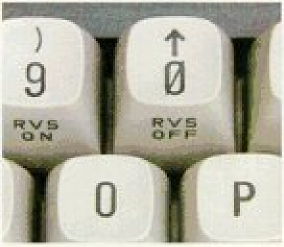
There are all sorts of report messages. They can tell you if your Commodore 16 has not understood an instruction. Problems can be caused by simple errors in typing. Type in the word PRONT instead of PRINT, and you will be told that you have made' a syntax error.



```
 COMMODORE BASIC V3.5 12277 BYTES FREE
READY.
PRONT 98765432

?SYNTAX ERROR
READY.
```

Try using PRINT to print some more numbers on the screen. Remember to press the RETURN key every time you want your Commodore 16 to carry out its instructions.



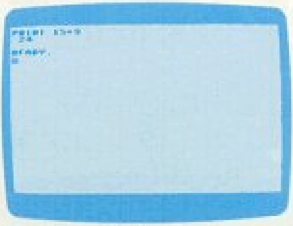


Symbols and sums

Adding and subtracting

You can also PRINT numbers and mathematical symbols. Some of these you will know already. For instance, type this in:

```
PRINT 15+9
```



```
PRINT 15+9
 24

READY.
```

Now press the RETURN key and, hey presto!, your Commodore 16 prints the answer to your sum.

Try this with some other numbers. You can add up more than two numbers. Remember to press the RETURN key when you want to see the answer.

You can also subtract, using the minus sign. You may use the minus and plus signs together on one line.

Multiplying and dividing

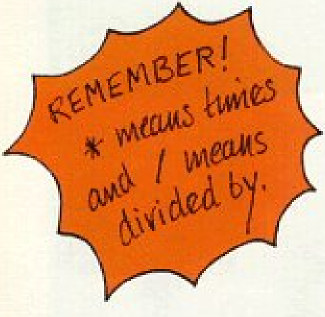
In BASIC you do not use the × or ÷ signs. Instead you use the * sign for multiplication, and for division you use the / sign. The / sign is next to the right hand SHIFT key.

To see how these work, try typing in:

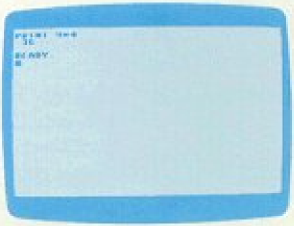
```
PRINT 9*4
```

Find out the answer and then type in:

```
PRINT 36/3
```

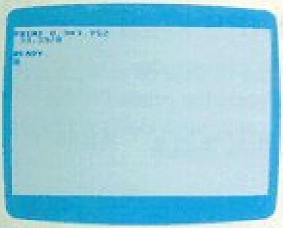


REMEMBER! * means times and / means divided by.



```
PRINT 9*4
 36

READY.
```

```
PRINT 36/3
 12

READY.
```



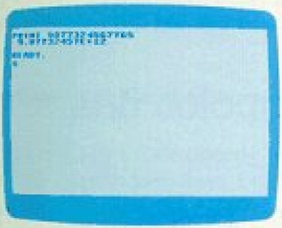
```
PRINT 8.9*3.752
 33.3928

READY.
```

Decimal points

You can use decimal numbers with your Commodore 16. For the decimal point you use the full stop sign. Type in:

```
PRINT 8.9*3.752
```

and press RETURN to see the answer.



```
PRINT 9877324567765
 9.87732457E+12

READY.
```

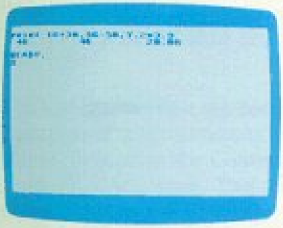
Big numbers

Have you noticed what happens when you try to use very large numbers? See what happens when you type in:

```
PRINT 9877324567765
```

and press the RETURN key.

The Commodore 16 cannot store very large numbers or fractions in their usual form. So it shortens them or expresses them in a different way.



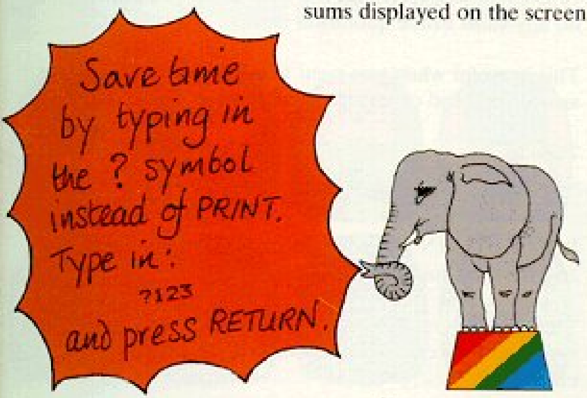
```
PRINT 18+30,96-50,7.2*3.9
 48        46        28.08

READY.
```

Separating sums

You can put more than one sum on one line, if you separate each one with a comma like this:

```
PRINT 18+30,96-50,7.2*3.9
```

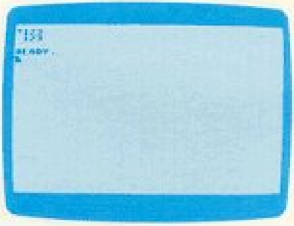
Press RETURN and you will see the answer to all three sums displayed on the screen.



Save time by typing in the ? symbol instead of PRINT. Type in:

```
?123
```

and press RETURN.



```
?123
 123
READY.
```

Letter strings

Words, shapes, even numbers can be printed on the screen using quotation marks. Any information printed in this way is known as a STRING, because several characters are strung together inside the quotation marks.

Your Commodore 16 can also print words. If you want to print words, you have to use quotation marks (press SHIFT and key 2). To see how this works, type in:

```
      PRINT "COMMODORE"
```

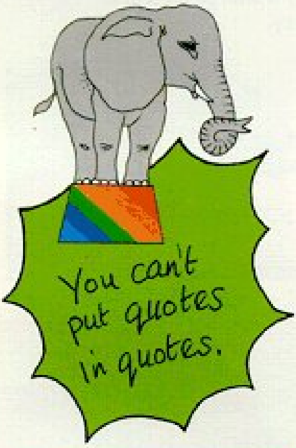
```
PRINT "COMMODORE"
COMMODORE

READY.
```

Press RETURN, and your Commodore 16 will carry out its instructions. Now type in:

```
      PRINT "BASIC
```

```
PRINT "BASIC
BASIC

READY.
```

and press RETURN. Leaving out the second set of quotation marks makes no difference. Some computers do not let you do this. The Commodore 16 is quite happy with just one set of quotation marks, in front of the words you want to print.



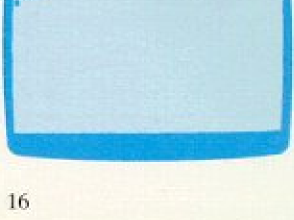
Number strings

You can print numbers inside quotation marks too. When you print numbers and symbols inside quotation marks your Commodore 16 will not treat them as a sum. It won't do any sums inside quotation marks.

This is useful when you want to print both a sum and its answer, instead of just the answer on its own.

Type in:

```
      PRINT "15*3=";15*3
```



```
PRINT "15*3=";15*3
15*3= 45

READY.
```

Press RETURN and what do you see? Your Commodore 16 prints the sum and its answer.

The semi-colon ; symbol is very useful. Use it when you want certain items of information to be printed next to each other.

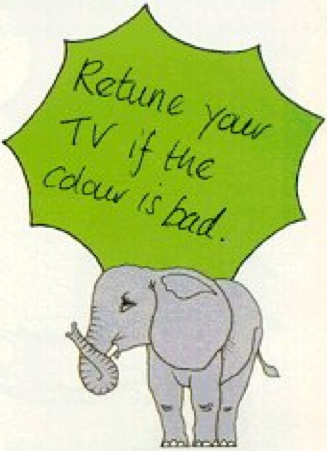
Shapes . . .

You can also print graphics shapes in quotation marks. Try these ones.

```
PRINT "◆ ▚ ◤ ╳ ♠"
```

```
PRINT "[illegible]"
[illegible]

READY.
```

Type in PRINT, open the quotation marks and then type in some shapes. Remember to press SHIFT and the key you want for the shape printed on the right of a key. Press the Commodore key and that key for the shape on the left.

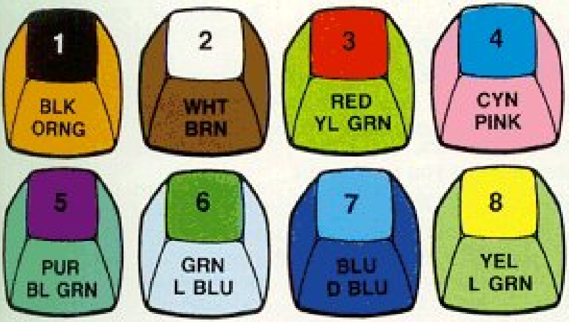


Retune your TV if the colour is bad.

. . . and colours

You can add colour directly. Printed on the front of each of the number keys from 1 to 8 are the names of two colours. Pressing the CTRL key and one of these keys will give you the top colour. Pressing the Commodore key and that key gives you the bottom colour. For example, pressing the Commodore key and key 7 gives you a dark blue cursor.

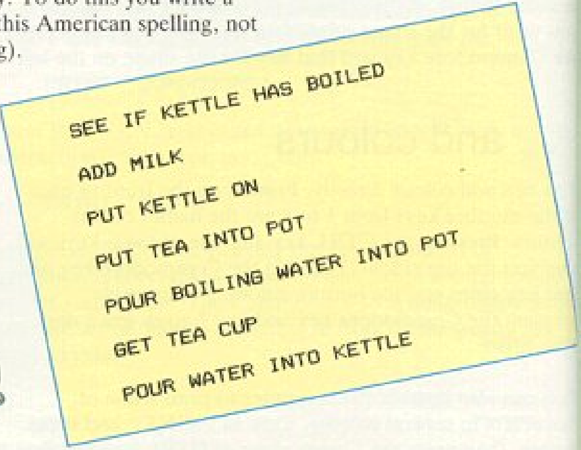
You can also instruct the computer to print a line of characters in several colours. Type in PRINT " and some shapes, then press the Commodore or CTRL key and the number key you want. This doesn't change the colour of the cursor now, but it prints a graphics shape known as a CONTROL CODE. The control code for red is a pound note sign. This doesn't get printed when you press RETURN, but it does change the colour of the shapes after it.



```
PRINT "[illegible]"
```

Writing Programs

So far you've looked at how to give your Commodore 16 simple commands which it has carried out directly. But you can also make it carry out whole lists of instructions, which it is able to store in its memory. To do this you write a PROGRAM (you always use this American spelling, not PROGRAMME, in computing).



```
SEE IF KETTLE HAS BOILED

ADD MILK

PUT KETTLE ON

PUT TEA INTO POT

POUR BOILING WATER INTO POT

GET TEA CUP

POUR WATER INTO KETTLE
```

What is a program? Look at Commodore's instructions. They are a type of PROGRAM, not for a computer but for you, which describes how you might make a cup of tea.

Each line is an instruction, but somehow they have got jumbled up. How would you sort them into the right order?

If you get the instructions in the right order, you can make a cup of tea. But if you get them in the wrong order you may have to go thirsty.

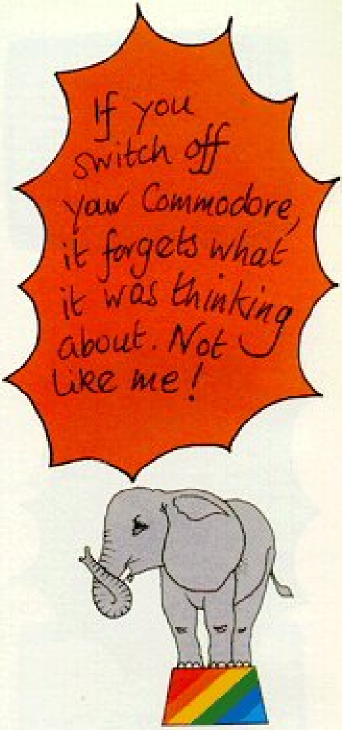
Of course you know how to make a cup of tea. You know that you need to use boiling water and that you need to get a cup ready. Computers aren't so clever. To program them you must give them very clear and precise instructions, in the right order. If you don't, your computer will be in as much of a mess as you would be if you tried to make a cup of tea without putting any water in the kettle first!

```
1. POUR WATER INTO
   KETTLE

2. PUT KETTLE ON

3. SEE IF KETTLE
   HAS BOILED

4. PUT TEA INTO POT

5. POUR BOILING
   WATER INTO POT

6. GET TEA CUP

7. ADD MILK

8. POUR TEA INTO
   CUP
```

Line numbers

A program is a list of instructions written in BASIC. So that the computer knows in what order to carry out the instructions, each line of instructions is given a number. Before typing in any instruction you put a line number so that that line can be stored in the computer's memory.

When it carries out the program you've typed in, your Commodore 16 looks for the lowest line number and starts there.


If you switch off your Commodore, it forgets what it was thinking about. Not like me!

Your first program

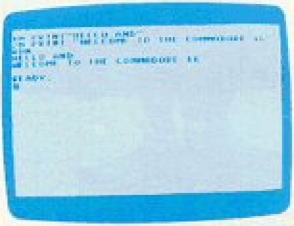
To see how a simple program works, type in this line:

```
    10 PRINT "HELLO AND"
```

When you've done this press RETURN and type in:

```
    20 PRINT "WELCOME TO THE COMMODORE 16"
```

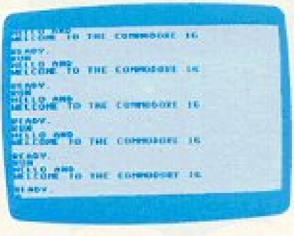
Press RETURN again. You've just written your first computer program. Congratulations! Now you have to tell your Commodore 16 to carry out the program.



```
10 PRINT "HELLO AND"
20 PRINT "WELCOME TO THE COMMODORE 16"
RUN
HELLO AND
WELCOME TO THE COMMODORE 16

READY.
```

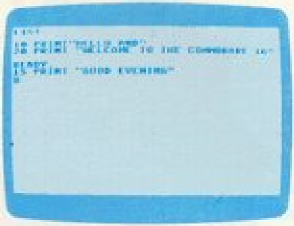
RUN

Type in the BASIC word RUN and press RETURN. You should now see your message on the screen. Your Commodore 16 has carried out the instructions in your program.



```
HELLO AND
WELCOME TO THE COMMODORE 16

READY.
RUN
HELLO AND
WELCOME TO THE COMMODORE 16

READY.
RUN
HELLO AND
WELCOME TO THE COMMODORE 16

READY.
RUN
HELLO AND
WELCOME TO THE COMMODORE 16

READY.
RUN
HELLO AND
WELCOME TO THE COMMODORE 16

READY.
```

Once you have written a program, you can RUN it as often as you like. You don't have to type the program in again. Try running your program a few times. When you reach the bottom of the screen, the screen display moves up. This is known as SCROLLING.



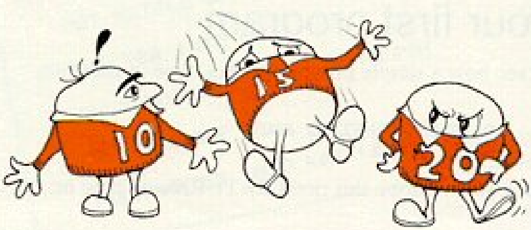
```
LIST
10 PRINT"HELLO AMY"
20 PRINT "WELCOME" TO THE COMMODORE 16"
READY
15 PRINT "GOOD EVENING"
```

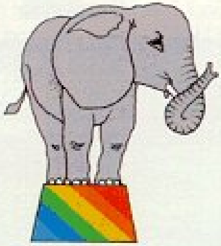
Listing your program

The list of instructions that make up your program is called a PROGRAM LISTING. If you run the program often enough, the listing scrolls off the top of the screen. Don't worry! It isn't lost. It is still stored in your Commodore 16's memory.

To get your listing back, you use the word LIST. Type in the word LIST now and press the RETURN key. Your listing will appear on the screen. If you want to clear the screen first, press the SHIFT and CLEAR HOME keys at the same time and then list the program.

Don't forget to leave room for latecomers in your program listings.





Leaving space

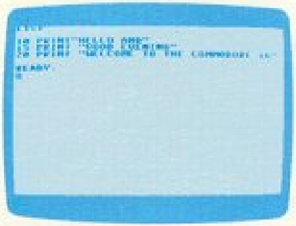
Look at those line numbers again. Why is there a gap of ten between each one? Why not just start at line 1 and go on to line 2?

Well, nobody's perfect. You may want to put some more instructions in between your first and second lines. A gap of ten leaves enough room for you to do this. Try typing in a new line such as:

```
     15 PRINT "GOOD EVENING."
```

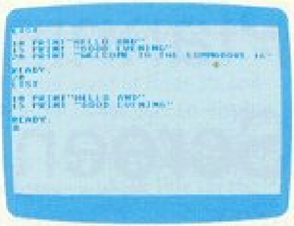
Don't forget to press RETURN. Now type in the instruction LIST and press RETURN again. As you can see, line 15 has jumped into place between lines 10 and 20.

RUN the program and you will see what difference line 15 makes to your screen display.



```
LIST
10 PRINT"HELLO AMY"
15 PRINT "GOOD EVENING"
20 PRINT "WELCOME TO THE COMMODORE 16"
READY.
```

Chopping and changing



```
LIST
10 PRINT"HELLO AND"
15 PRINT "GOOD EVENING"
20 PRINT "WELCOME TO THE COMMODORE 16"

READY.
20
LIST
10 PRINT"HELLO AND"
15 PRINT "GOOD EVENING"

READY.
```

Removing lines

As well as putting new lines into your programs you can take lines out. List the program that you have just written. What if you want to get rid of line 20? All you have to do is type in:

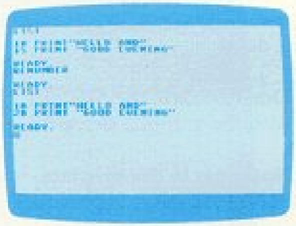
```
20
```

Press RETURN. Now list the program. Line 20 has disappeared from your program listing.

If you want to get rid of a line in a program, just type in its line number and press RETURN. Make sure that it's the right one though!

Changing a line

If you want to change the instructions in one of your lines, write the line again and press RETURN. The new version of your line replaces the old one.



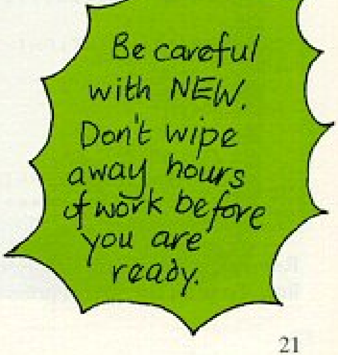
```
LIST
10 PRINT"HELLO AND"
15 PRINT "GOOD EVENING"

READY.
RENUMBER

READY.
LIST
10 PRINT"HELLO AND"
20 PRINT "GOOD EVENING"

READY.
```

Renumbering lines

Programs always look neater when there's a gap of ten between each line number. If you want to tidy up a program which contains different gaps, type in:

```
RENUMBER
```

Press RETURN and then list the program. This renumbers the lines in your program and of course means that you can add some more lines in between again!



NEW

Sometimes you may want to get rid of a whole program. To do this you type in the BASIC word NEW and press the RETURN key. NEW removes the program from your Commodore 16's memory. Now you can type in a new program.

Planning Screen Displays

Don't worry that your typing will go off the screen. Your Commodore 16 will move it to the next line for you.

One way to improve displays is to clear the screen before the program runs, by putting in the SHIFT/CLEAR instruction at the beginning of your program.

To do this type in:

```
    10 PRINT "
```

Now press the SHIFT and CLEAR HOME keys at the same time. This prints the heart shaped control code which clears the screen. Line 10 should now look like this:

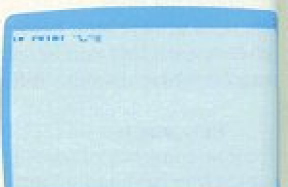
```
    10 PRINT "♥
```

You will see this line at the beginning of many programs so make sure that you understand what it looks like and what it does.

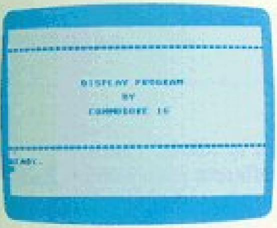
Don't confuse the SHIFT/CLEAR HOME control code with the graphics shape on key S.

Now type in this program, which as you can see uses that SHIFT/CLEAR HOME sign at line 10:

```
10 PRINT "♥"
20 PRINT:PRINT
30 PRINT "******************************
**********"
40 PRINT:PRINT:PRINT:PRINT
50 PRINT "                DISPLAY PROGRAM"
60 PRINT
70 PRINT "                        BY"
80 PRINT
90 PRINT "                   COMMODORE 16"
100 PRINT:PRINT:PRINT:PRINT
110 PRINT "*****************************
***********"
```



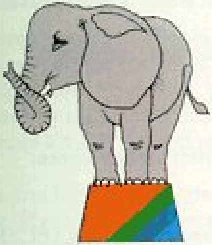
Remember to press RETURN before going on to a new line. To print the spaces press the SPACE BAR.



```
****************************************

            DISPLAY PROGRAM
                  BY
             COMMODORE 16

****************************************
READY.
```

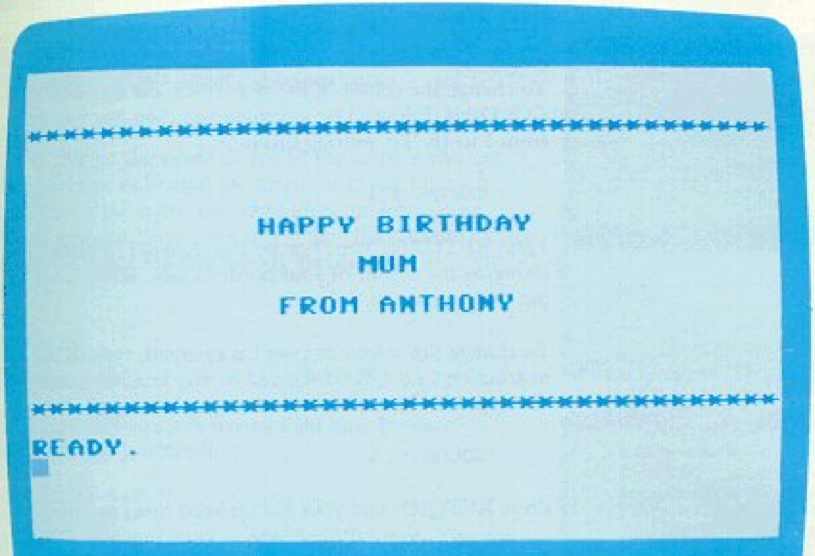
Now RUN the program. If you would like to experiment with the spacing to make your screen display look neat, list the program and change the lines you want to rewrite. You can type in any graphics shapes you want at lines 30 and 110, to decorate the top and bottom of the screen display.

On some lines you will see the word PRINT on its own without any instructions. This leaves a blank line on the screen and helps to space out any words or messages you use.



At line 90, where it says COMMODORE 16, why not put your own name?

You could change this program in all sorts of ways, to make a title page for a program or just a friendly greeting for a friend.



```
****************************************



             HAPPY BIRTHDAY
                  MUM
              FROM ANTHONY



****************************************
READY.
```

Adding colour

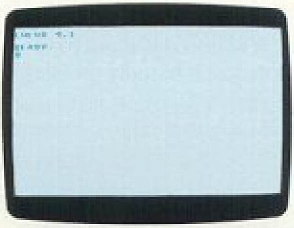
Screen displays look even better in colour. The Commodore 16 offers you 16 different main colours. Each can be in any of eight shades, so altogether you have a choice of 128 colours.

These can be used in a number of ways. You can have:

1. A coloured frame or border for your displays.

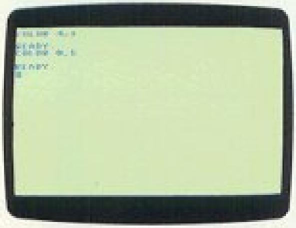
2. A coloured background inside the border.

3. Coloured letters or shapes printed on the screen.



```
COLOR 4,1
READY.
█
```

COLOR
To get these effects you use the instruction COLOR. (Note the American spelling here.) This may be followed by three numbers.



```
COLOR 4,1
READY.
COLOR 0,6
READY.
█
```

To change the colour of the border use the instruction COLOR 4, followed by a comma and then any number from 1 to 16. Try putting this in:

```
    COLOR 4,1
```

Press RETURN. Your border has changed to black. Try changing the colour of your border again. Remember to put that comma in.

To change the colour of your background, type in the instruction COLOR 0 followed by any number from 1 to 16. Type this in:

```
    COLOR 0,6
```

Press RETURN and your background turns to green.

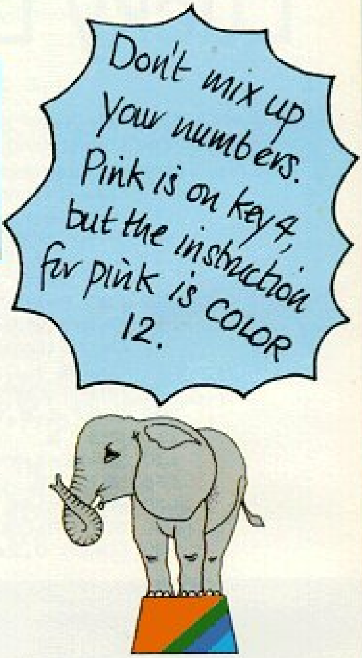
Try some more combinations of background and border colours. If you get confused, just press the RESET button. This restores the normal screen display colours.

'MORFAX'

To get back to the normal screen colours and your program listing, press RUN STOP and RESET at the same time. Don't worry about what appears on the screen, just type in X, press RETURN and then LIST the program as normal.

Each colour has its own code number. The table below lists what these are.

1	BLACK	9	ORANGE
2	WHITE	10	BROWN
3	RED	11	YELLOW-GREEN
4	CYAN	12	PINK
5	PURPLE	13	BLUE-GREEN
6	GREEN	14	LIGHT BLUE
7	BLUE	15	DARK GREEN
8	YELLOW	16	LIGHT GREEN


Don't mix up your numbers. Pink is on key 4, but the instruction for pink is COLOR 12.

Coloured words and shapes

You can alter the colour of any words or shapes printed on the screen too. To do this, use the instruction COLOR 1 followed by the code number of the colour you want. For instance COLOR 1,3 will give you red letters.

When you try these out, they can be too pale and difficult to see. If you want to return to black letters, just press the CTRL key and key 1. This changes the cursor to black again.

If you want to use more than one colour for the words in your programs, don't forget the colour codes.

Light and dark

You can change the shade of any of the colours you use. So far you have only used the brightest shade for each colour. If you put a third number after the instruction COLOR you can choose the shade too. Each shade has its own code number ranging from 0, the darkest, to 7, the lightest.

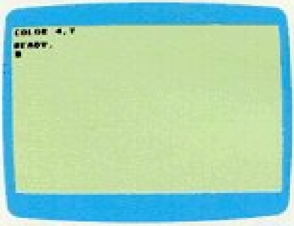
Type this in:

```
COLOR 4,7
```

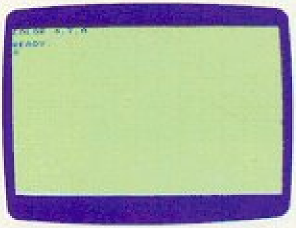
Press RETURN and you have a light blue border. However, now type this in:

```
COLOR 4,7,0
```

Press RETURN and you can see the difference. You can change the shade of any colour used for the border, background and words or shapes printed on your screen. Try a few changes now.



```
COLOR 4,7
READY.
■
```



```
COLOR 4,7,0
READY.
■
```

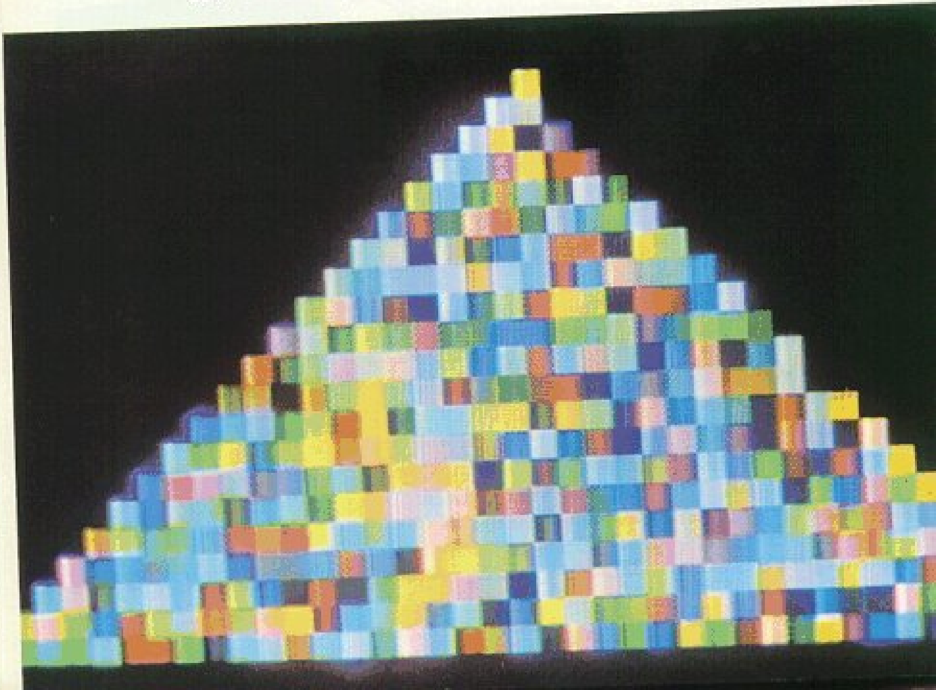
Lucky Dip

Here are some programs for you to type in and try. Don't worry if you don't understand everything in them. Just have fun. This is the program for the picture below.

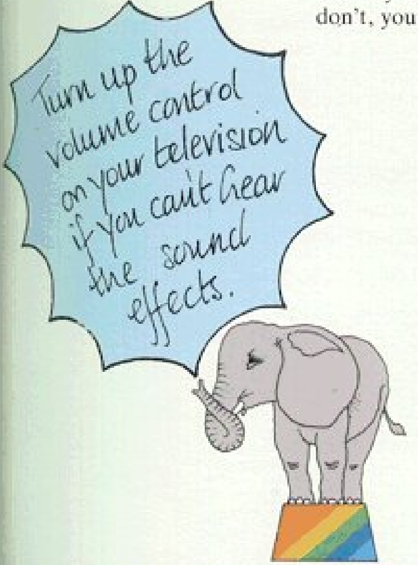
```
 10 SCNCLR
 20 XE=39:Y=22
 30 COLOR 4,1:COLOR 0,1
 40 FOR XS=0 TO 20
 50 FOR B=XS TO XE
 60 S=INT(RND(1)*4)+3
 70 C=INT(RND(1)*15)+2
 80 COLOR 1,C,S
 90 PRINT CHR$(18);
100 CHAR 1,B,Y,CHR$(160)
110 NEXT B
120 XE=XE-1:Y=Y-1
130 NEXT XS
140 GETKEY A$
150 SCNCLR:COLOR 4,2
160 COLOR 0,2:COLOR 1,1
```

"MORFAX"

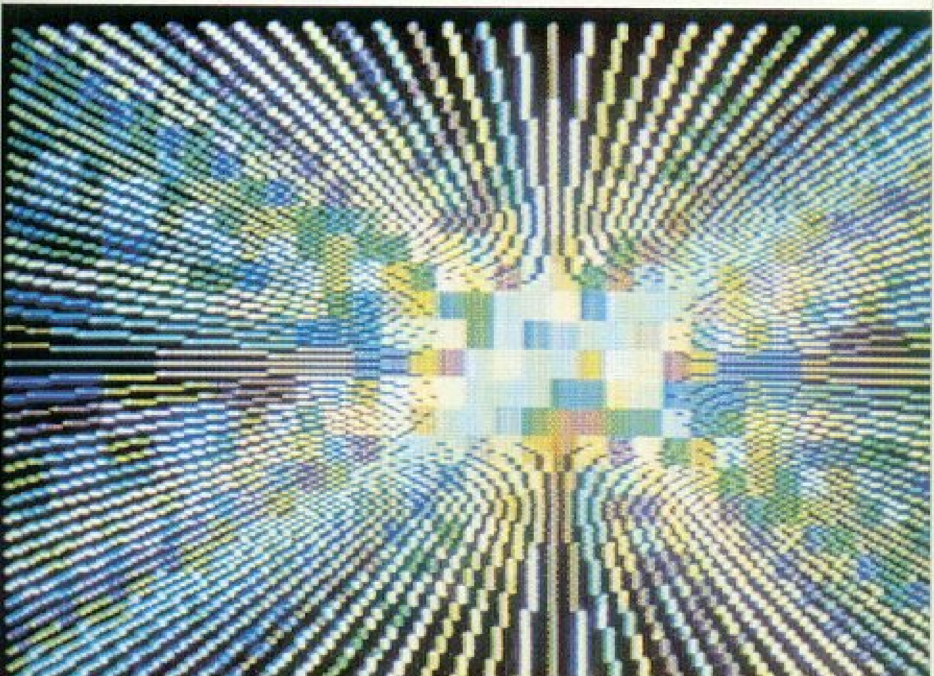
When you RUN this program, you won't see the word READY on the screen. This is because of lines 140 to 160. To end the program and clear the screen press any key. You can add these lines to any program.



Now try this one. Make sure you type it in correctly. If you don't, your Commodore 16 will soon let you know.

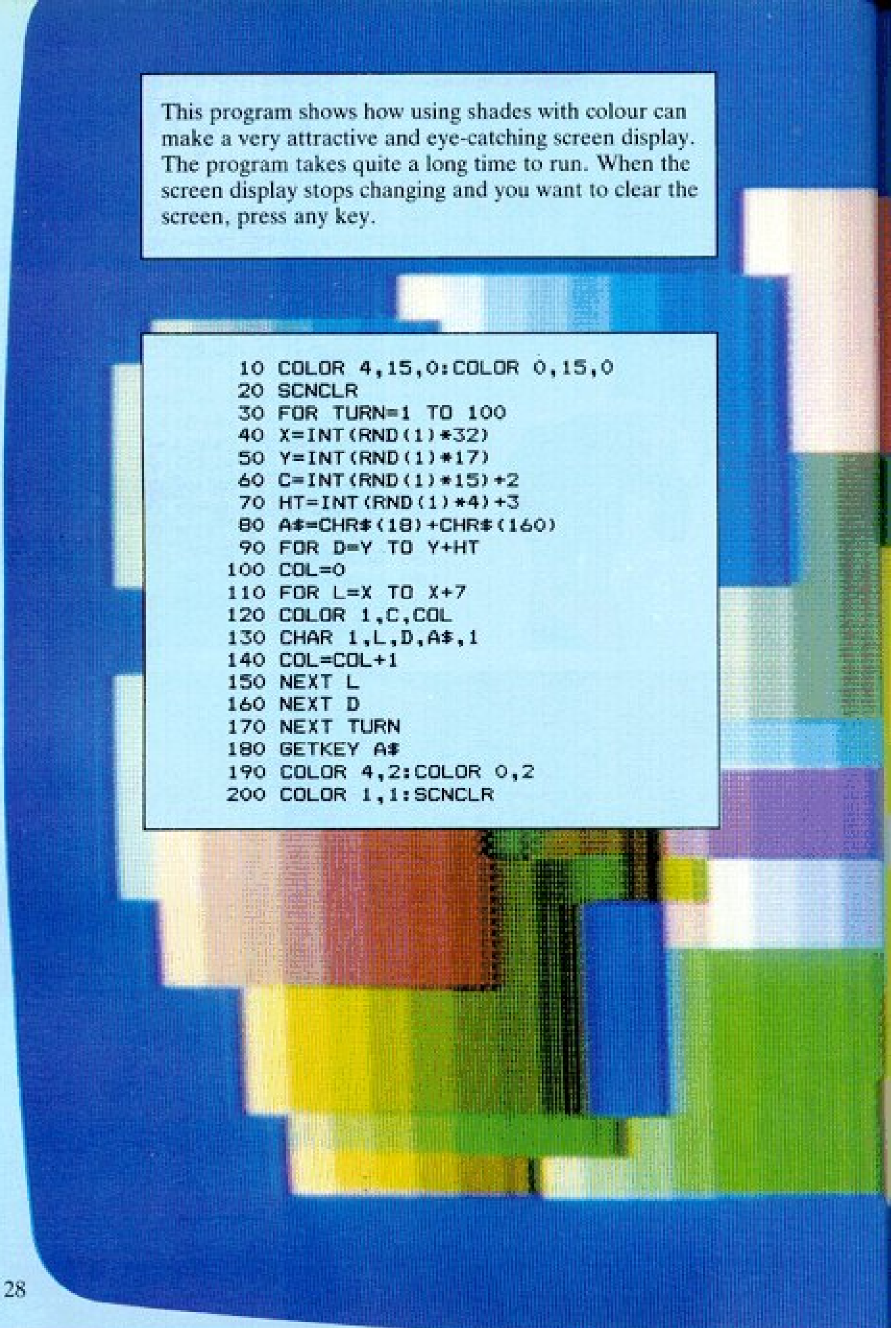

Turn up the volume control on your television if you can't hear the sound effects.

```
 10 GRAPHIC 3,1
 20 VOL 8
 30 COLOR 0,1:COLOR 4,1
 40 FOR X=0 TO 160 STEP 5
 50 C=INT(RND(1)*15)+2
 60 B=INT(RND(1)*4)+3
 70 COLOR 1,C,B
 80 DRAW 1,X,0 TO 160-X,200
 90 SOUND 1,X*5,5
100 NEXT X
110 FOR Y=0 TO 200 STEP 5
120 C=INT(RND(1)*15)+2
130 B=INT(RND(1)*4)+3
140 COLOR 1,C,B
150 DRAW 1,0,Y TO 160,200-Y
160 SOUND 1,Y*5,5
170 NEXT Y
180 GETKEY A$
190 GRAPHIC 0
200 COLOR 4,7: COLOR 0,2
210 COLOR 1,1
```



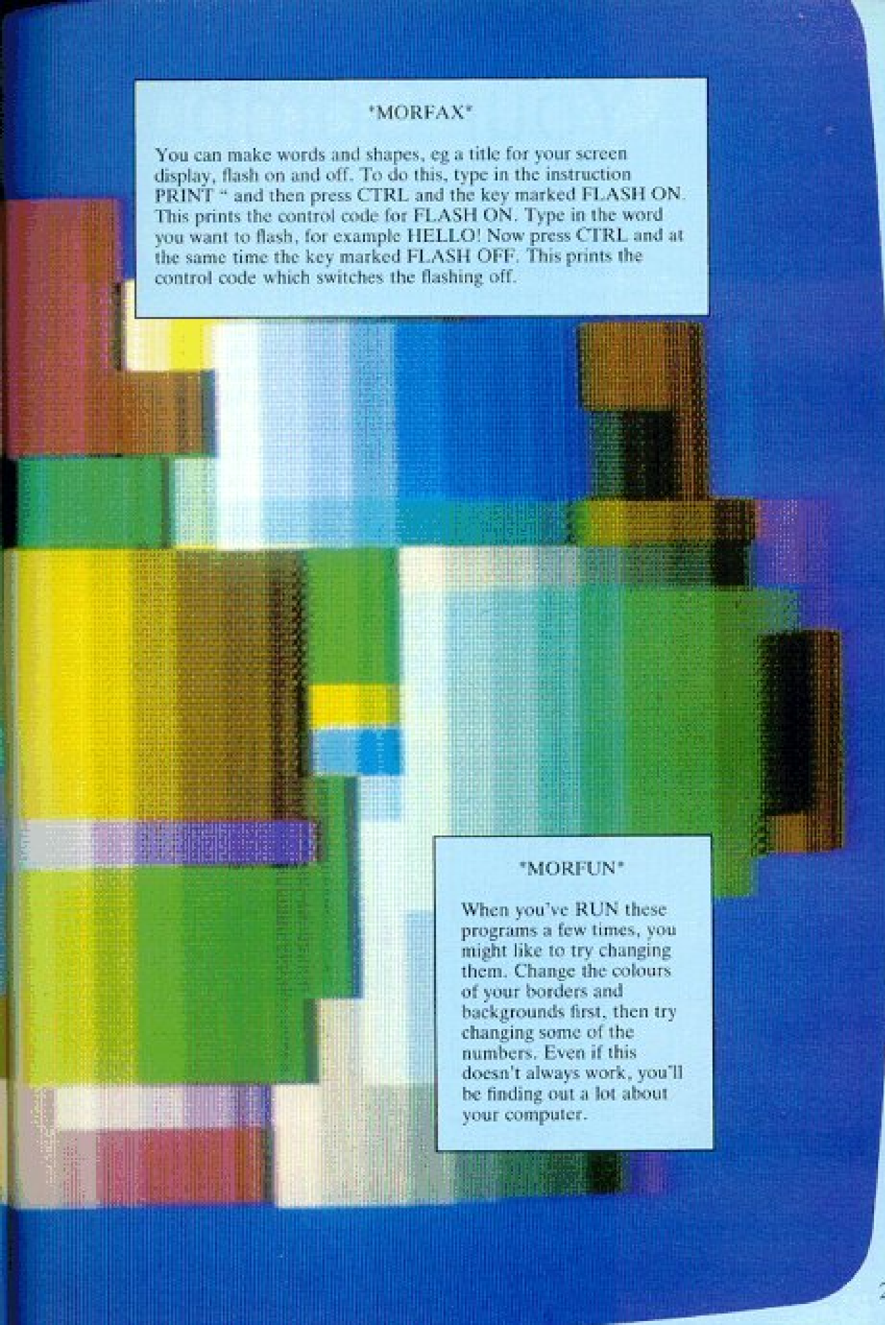
This program shows how using shades with colour can make a very attractive and eye-catching screen display. The program takes quite a long time to run. When the screen display stops changing and you want to clear the screen, press any key.

```
 10 COLOR 4,15,0:COLOR 0,15,0
 20 SCNCLR
 30 FOR TURN=1 TO 100
 40 X=INT(RND(1)*32)
 50 Y=INT(RND(1)*17)
 60 C=INT(RND(1)*15)+2
 70 HT=INT(RND(1)*4)+3
 80 A$=CHR$(18)+CHR$(160)
 90 FOR D=Y TO Y+HT
100 COL=0
110 FOR L=X TO X+7
120 COLOR 1,C,COL
130 CHAR 1,L,D,A$,1
140 COL=COL+1
150 NEXT L
160 NEXT D
170 NEXT TURN
180 GETKEY A$
190 COLOR 4,2:COLOR 0,2
200 COLOR 1,1:SCNCLR
```



MORFAX

You can make words and shapes, eg a title for your screen display, flash on and off. To do this, type in the instruction PRINT " and then press CTRL and the key marked FLASH ON. This prints the control code for FLASH ON. Type in the word you want to flash, for example HELLO! Now press CTRL and at the same time the key marked FLASH OFF. This prints the control code which switches the flashing off.



MORFUN

When you've RUN these programs a few times, you might like to try changing them. Change the colours of your borders and backgrounds first, then try changing some of the numbers. Even if this doesn't always work, you'll be finding out a lot about your computer.

Inside Your Computer

Inside your computer are a number of chips mounted on a circuit board. While it is working, your computer is busy sending electronic signals to and from these chips. These signals travel along the metal tracks printed on the board. Each set of chips has a particular job to do.

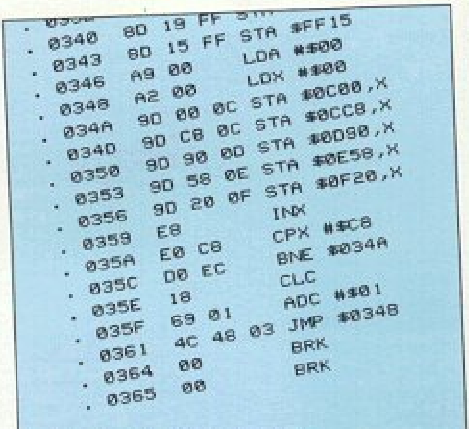
The most important chip is the CPU (Central Processing Unit). It is the 'general manager' of the computer and controls all the other chips. The CPU for your Commodore 16 is sometimes called the TED chip. The TED chip also handles graphics and sound effects.

When it is at work your computer does not use BASIC. It uses a special language known as MACHINE CODE. You can write programs in machine code. But it is very complicated, so most people use BASIC.

The ROM chips translate the BASIC words you type in into machine code so that your computer understands them. (ROM stands for Read Only Memory.)

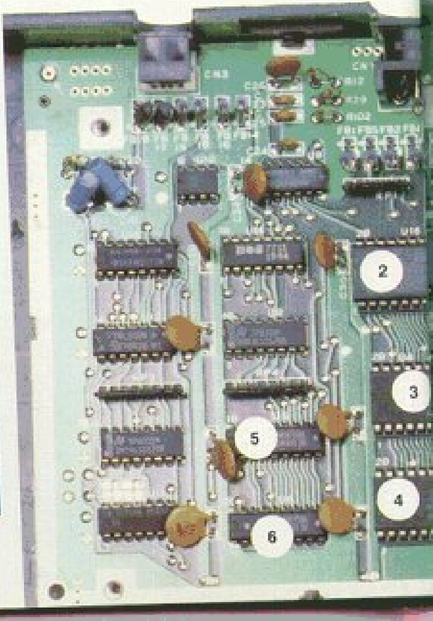
'MORFAX'

Machine code runs faster than BASIC! Most arcade games are programmed in machine code to take advantage of this.



```
. 0340  8D 19 FF  ST
. 0343  8D 15 FF  STA $FF15
. 0346  A9 00     LDA #$00
. 0348  A2 00     LDX #$00
. 034A  9D 00 0C  STA $0C00,X
. 034D  9D C8 0C  STA $0CC8,X
. 0350  9D 90 0D  STA $0D90,X
. 0353  9D 58 0E  STA $0E58,X
. 0356  9D 20 0F  STA $0F20,X
. 0359  E8        INX
. 035A  E0 C8     CPX #$C8
. 035C  D0 EC     BNE $034A
. 035E  18        CLC
. 035F  69 01     ADC #$01
. 0361  4C 48 03  JMP $0348
. 0364  00        BRK
. 0365  00        BRK
```

This machine code routine puts characters on the screen and changes them very quickly.



Inside the computer:

1. The TED chip

2, 3, 4. The ROM chips

5, 6. The RAM chips

The memory store

REMEMBER!
Memory space
is important.
Don't waste it.

The programs that you feed into your Commodore 16 are stored in the RAM chips. RAM stands for Random Access Memory. The RAM chips store your programs only while the computer is switched on.

Programs take up space in the RAM. The longer and more complicated the program, the more space they take up.

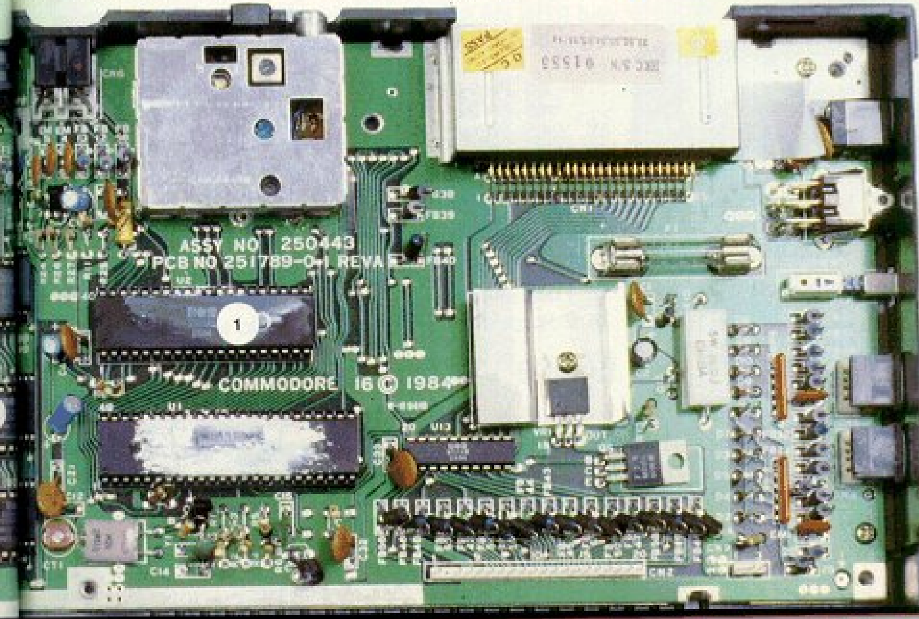
The amount of space available to store progams is measured in kilobytes or K. One kilobyte of memory can store about five hundred BASIC words and instructions. This may sound like a lot but long programs can and often do fill nearly all the memory storage space in RAM.

Because of this, it is most important that programs don't take up unnecessary space in the computer's memory. One reason why BASIC is more like a code than a language (such as English or French), is to save memory space.

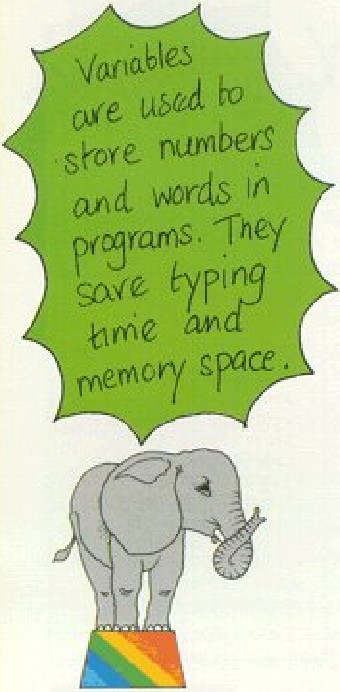
Short programs like the ones in this book aren't likely to fill the memory space of your Commodore 16. All the same, saving memory space is an important part of BASIC programming. As you will see, it can save typing time too.

"MORFAX"

A kilobyte (K) is made up of one thousand and twenty-four BYTES. A byte is a binary number made of eight BITS (0 or 1) eg 11000101. The Commodore 16 has 16K of memory. Some of this is used by the computer, leaving 12277 bytes (just under 12K) free for program storage.



Variables


Variables are used to store numbers and words in programs. They save typing time and memory space.

LET A stand for . . .

Some numbers and words may be used more than once in a program. But you don't always need to type the same information in over and over again. Instead you can store it with the BASIC word LET.

To see how this works, type in:

```
10 LET A=9.872
```

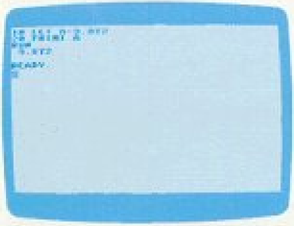
What you have done here is call the number 9.872 by the name 'A'. Your Commodore 16 now stores this number in its memory as A. Every time you want to use 9.872 now, don't say 9.872, just say A. Add this next line:

```
20 PRINT A
```

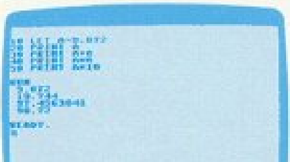
and RUN the program to see how this works. You can use A as often as you like. Now add these lines:

```
30 PRINT A+A
40 PRINT A*A
50 PRINT A*10
```

and RUN the program. Typing A is easier than typing out the number 9.872. Although the number 9.872 is used six times in this program, it is only stored once in your Commodore 16's memory. This helps to save memory space.



```
10 LET A=9.872
20 PRINT A
RUN
 9.872

READY.
```



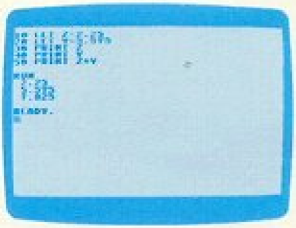
```
10 LET A=9.872
20 PRINT A
30 PRINT A+A
40 PRINT A*A
50 PRINT A*10
RUN
 9.872
 19.744
 97.4563841
 98.72

READY.
```

More variable names

The letter A was used to stand for 9.872 in the last program. You can use any letter of the alphabet that you like, as long as you say what number it stands for at the beginning of your program.

You can also use more than one letter at a time. Type in:

```
10 LET Z=2.25
20 LET Y=7.575
30 PRINT Z
40 PRINT Y
50 PRINT Z+Y
```



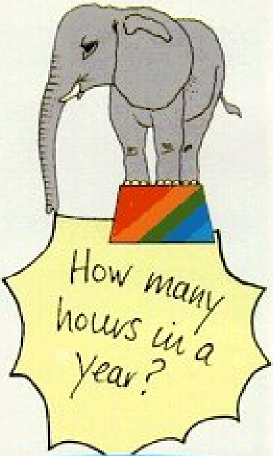
and run it. Here again, why not add some lines of your own using Z and Y?

The letters that take the place of the numbers in your programs are called VARIABLES. Z and Y are examples of VARIABLE NAMES.

The word 'variable' means something which can be changed. Look at that last program which you typed in. Change the first two lines like this:

```
10 LET Z=5000
20 LET Y=256
```

Now when you run your program, your Commodore 16 stores different numbers as Z and Y. You don't need to change the rest of the program to get a different result. This makes variables very useful.



Longer variable names

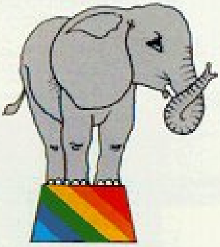
Letters like A and B or X and Y don't mean very much on their own. It helps to use longer names that describe what the variable stands for. When you read your listings, you should then be able to see what each variable stands for.

This next program shows you how to use longer variable names. It works out how many hours there are in two weeks. The number of days is multiplied by the number of hours in each day. Type in:

```
10 LET DAYS=14
20 LET HOURS=24
30 PRINT DAYS*HOURS
```

Run the program.

Variable names can be either UPPER CASE or lower case.



'MORFAX'

When your Commodore 16 looks at variable names, it only checks the first two letters in each. You must make sure that the first two letters used in a variable are not the same as the first two letters of any other variable in your program; for example, you should not use both MILES and MINUTES.

Microshopping

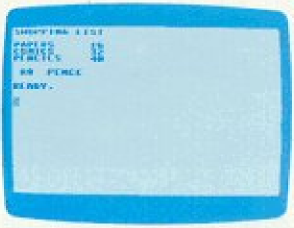
The program below adds up a list of groceries. The variables store the price of each item. Type it in and RUN it:

```
10 LET PAPERS=16
20 LET COMICS=32
30 LET PENCILS=40
40 PRINT PAPERS+COMICS+PENCILS
```

The screen display for this program is not very 'friendly'. It just prints a number. Add the following lines and see the improvement:

```
 5 PRINT "SHOPPING LIST": PRINT
15 PRINT "PAPERS ",PAPERS
25 PRINT "COMICS ",COMICS
35 PRINT "PENCILS ",PENCILS
40 PRINT:PRINT PAPERS+COMICS+PENCILS;
" PENCE"
```

```
5 PRINT "SHOPPING LIST": PRINT
15 PRINT "PAPERS ",PAPERS
25 PRINT "COMICS ",COMICS
35 PRINT "PENCILS ",PENCILS
40 PRINT:PRINT PAPERS+COMICS+PENCILS;" 
PENCE"
```



```
SHOPPING LIST

PAPERS      16
COMICS      32
PENCILS     40

 88  PENCE
READY.
```

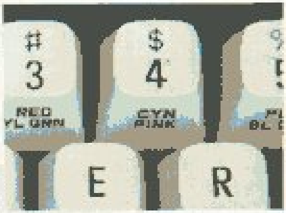
The number of hours in the day stays the same, but the price of goods can change. Here it is helpful to be able to change numbers stored as variables, without changing the rest of the program. Only the numbers need to be changed when the prices change. The rest of the program can stay the same.

Storing words

Variables can be used to store words, sentences, or just parts of words, using the BASIC word LET. Type in:

```
10 LET W$="COMMODORE 16"
20 PRINT W$
```

Run the program. That dollar sign ($) after the variable name is very important. It tells your computer that you want to store words or strings. Words or strings to be stored as variables must also be put in quotation marks, or your Commodore 16 will tell you that you've made a syntax error.



Now that it is stored in the memory, every time you want to use the words COMMODORE 16, don't say COMMODORE 16, say W$! Add these lines to your program now to see this working:

```
   20 PRINT "I AM PROGRAMMING MY ";W$
   30 PRINT "THE ";W$;"HAS A 16K MEMORY"
   40 PRINT "I AM LEARNING HOW TO PROGRAM
 MY ";W$
```

Longer variable names can also be used with words. They help to describe the kind of information stored. For example, type in:

```
     10 LET NAME$=" MIKE ROE"
     20 LET ADDRESS$="16 K MEMORY LANE"
     30 LET PLACE$="CHIPPENHAM"
     40 PRINT "MY NAME IS ";NAME$
     50 PRINT "I LIVE AT ";ADDRESS$
     60 PRINT "MY HOME TOWN IS ";PLACE$
```

and run the pogram.

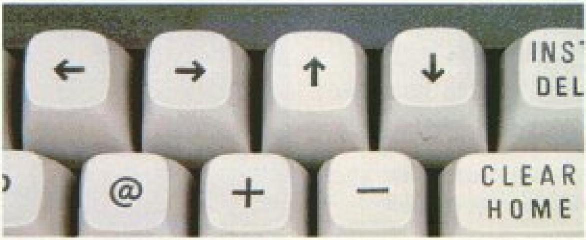
'MORFAX'

Your computer will not accept BASIC words as variables. You cannot for instance say LET RUN=1 or LET GOTO$="HI"

Editing Programs

When you want to change a line you can type in a completely new version, or you can EDIT the line already there. Type in:

```
    10 PRINT "I AM 12 YEARS OLD"
    20 PRINT "MY NAME IS COMMODORE"
```



Press the key marked ↑. The cursor moves up one line. Do this again so that the cursor is on line 10. Now press the key marked →. This moves the cursor along the line. To move the cursor backwards, press the key marked ←. When the cursor is over the 2, type in 3.

Press the RETURN key so that the cursor moves to the next line in your progam. Now press the key marked ↓ until it is clear of the rest of the program lines. List your program and you will find that it has the 'edited' version of line 10.

Now move the cursor up to line 20 and move it along the line until it is in between the words IS and COMMODORE. Press the SHIFT and INST DEL keys. Do this four times. This opens up a space in your line.

Press the SPACE BAR once, then type in the word THE. Your line should now read:

```
    20 PRINT "MY NAME IS THE COMMODORE"
```

"MORFAX"

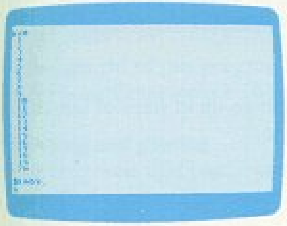
When you press RETURN after editing, you may see the message:

```
    ?OUT OF
    DATA ERROR.
```

Don't worry about this, it won't affect your programs.

Press RETURN again, move the cursor clear of the rest of the program and list it. You now have a new version of line 20.

If you have to type in a line with the same instructions as one that you've already typed in, move the cursor up to that line. Type in a new line number, press RETURN and move the cursor clear. This copies the old line and gives it a new line number.



Loops

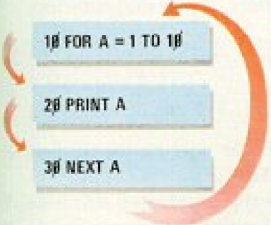
So far you have only used variables to stand for one number at a time. However, you can also make a variable stand for several different numbers in one program.

The program below shows you how a variable can stand for the numbers 1 to 10. Type in the program and run it:

```
10 FOR A=1 TO 10
20 PRINT A
30 NEXT A
```

The program prints the numbers 1 to 10. In this program then, the variable A stands FOR the numbers 1 to 10. Now change line 10 to this:

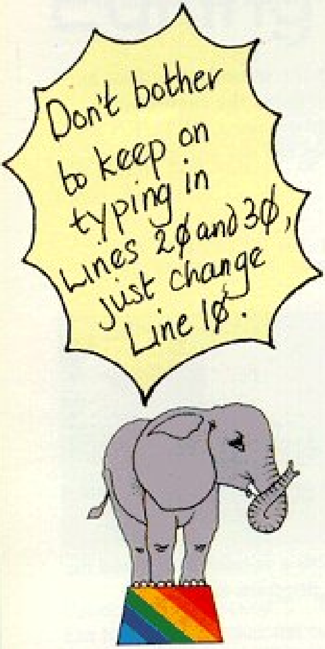
```
10 FOR A=1 TO 20
```



Run the program. Now A stands FOR the numbers 1 TO 20. This is called a LOOP. Your Commodore 16 begins by printing A as 1. This is the first value of A shown at line 10. It prints this value, then moves to line 30 which says NEXT A. This tells it to take the NEXT value of A, so it returns to line 10 to see what this is. In this program it is 2, the 'next' number after 1. The program keeps on going round and round like this until A is equal to 20, the last number mentioned at line 10.

FOR . . . NEXT loops

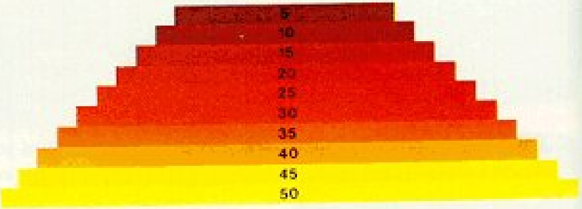
Because this kind of loop uses the BASIC words FOR and NEXT, it is known as a FOR . . . NEXT loop. You cannot use the word FOR unless it is followed by the word NEXT later on in your program. FOR . . . NEXT loops have many uses.


Don't bother to keep on typing in Lines 2ϕ and 3ϕ, just change Line 1ϕ.

STEP
You can use a FOR . . . NEXT loop to count up in steps, using the word STEP. Type in:

```
10 FOR A=5 TO 50 STEP 5
20 PRINT A
30 NEXT A
```

Run the program now. A now goes from 5 to 50 in steps of 5.

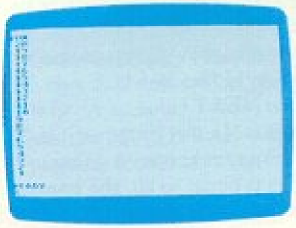


```
10 FOR A=0 TO 5 STEP .5
20 PRINT A
30 NEXT A
RUN
[illegible]
READY.
```

Another way of using STEP is to go up in steps of less than one, that is, in fractions. Type in:

```
10 FOR A=0 TO 5 STEP .5
20 PRINT A
30 NEXT A
```

Run the program and you will see that your program now prints A not just as whole numbers, but as decimals too.



```
[illegible]
READY.
```

STEP can also help you to count down as well as up. To do this you need to use a minus sign after STEP and start your loop with the highest number first. Try this to see how it works:

```
10 FOR A=20 TO 1 STEP -1
20 PRINT A
30 NEXT A
```

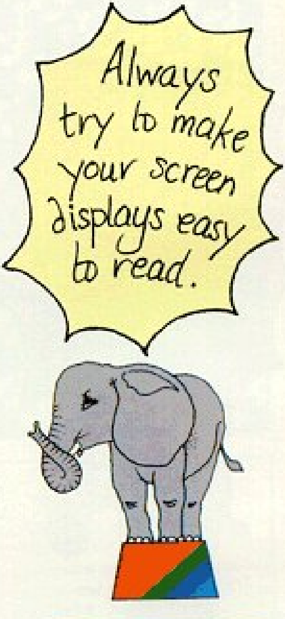
Have a go at changing the loop yourself now, using STEP with different numbers.

Doing your tables

FOR . . . NEXT loops can print out tables very quickly. This next program uses a loop to print out the 9 times table.

Instead of just printing out the numbers on their own, which looks rather dull, the program uses PRINT with words and symbols. This makes the screen display clearer and more informative. Type in:

```
10 SCNCLR
20 LET NU=9
30 PRINT "THE";NU;"TIMES TABLE"
40 PRINT
50 FOR N=1 TO 12
60 PRINT NU;"*";N;"=";NU*N
70 NEXT N
```

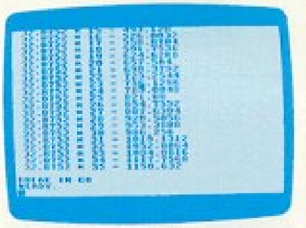


This is quite good fun but it does not really test your Commodore 16 to the full. The 9 times table is really rather easy. How well do you know the 32.8752 times table? Why not get your Commodore 16 to do it for you? Just change the 9 at line 20 and put 32.8752 instead. You'll be surprised at how quickly you get the answers!

Don't get rid of this program yet. Now we're going to make your Commodore 16 work even harder!

Scrolling and pausing
The programs that you've written so far have only taken up a small amount of the screen display space. Change line 50 to this:

```
50 FOR N=1 TO 60
```



Run the program and see just how quickly your Commodore 16 can work. Unfortunately it moves almost too quickly. Once the screen has filled up with information, it starts to SCROLL, ie the display rolls up and off the top of your screen.

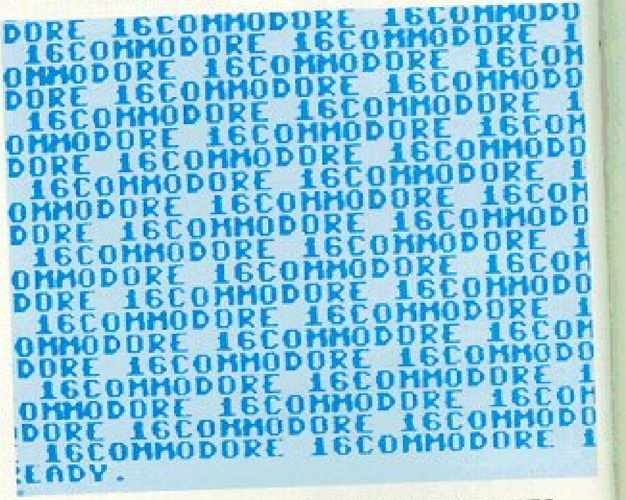
To stop this happening, press the RUN STOP key. This stops the program running and lets you see what is on the screen at the time. To continue with the program type in the command CONT (for Continue) and press RETURN.

Words and patterns

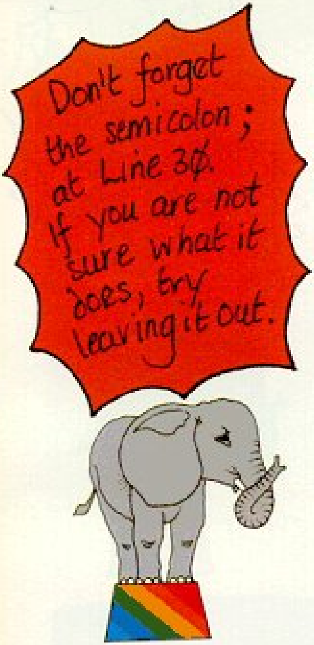
Loops can be used with words too. They can tell your Commodore 16 to print a word a certain number of times.Try this:

```
10 LET W$="COMMODORE 16"
20 FOR L=1 TO 80
30 PRINT W$;
40 NEXT L
```

This makes a pattern across the screen. Why not add some colour to this program, or use your own name to see what kind of pattern that makes? You can spend hours playing with this kind of loop.



```
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
OMMODORE 16COMMODORE 16COM
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
OMMODORE 16COMMODORE 16COM
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
OMMODORE 16COMMODORE 16COM
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
OMMODORE 16COMMODORE 16COM
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
OMMODORE 16COMMODORE 16COM
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
OMMODORE 16COMMODORE 16COM
DORE 16COMMODORE 16COMMODO
 16COMMODORE 16COMMODORE 1
READY.
```


Don't forget the semicolon ; at Line 3∅. If you are not sure what it does, try leaving it out.

MORFAX

A FOR . . . NEXT loop which does nothing can be used to slow down the screen display. Put one in at line 35 like this:

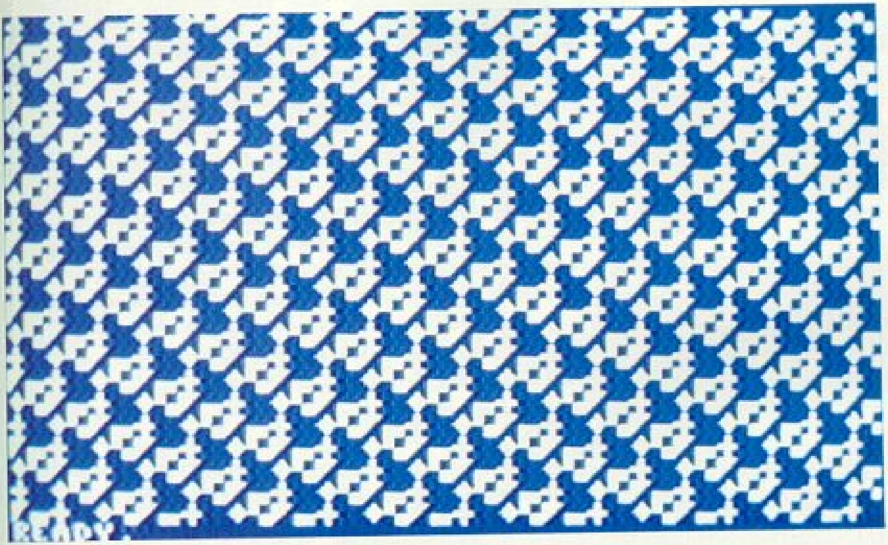
```
  35 FOR X=1 TO
500: NEXT X
```

and run the program again to see how it is affected.

Patterns can also be made using GRAPHICS SHAPES instead of letters, like this:

```
10 SCNCLR
20 COLOR 4,15,1
30 COLOR 0,15,1:COLOR 1,8,7
40 W$=" ◆ ▚ ◤ ▬ + "
50 FOR N=1 TO 200
60 PRINT W$;
70 NEXT
```

Run the program.



```
READY
```

Can you see any difference between this program and the last one? To save space and time you can leave out the word LET from your programs.

Try this with a different number of symbols strung together at line 40. What happens when you use an odd number instead of an even number?

MORFAX

That HELP key can be very useful, if your program contains a syntax error and you can't seem to find it. Just press the HELP key, and the line with the mistake will flash on and off!

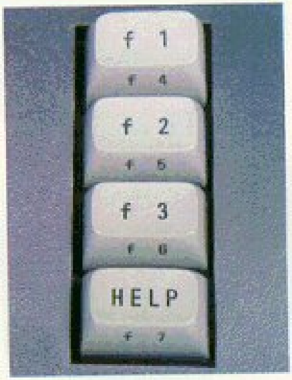
MORFAX

To the right of your keyboard are four large keys known as the FUNCTION KEYS.

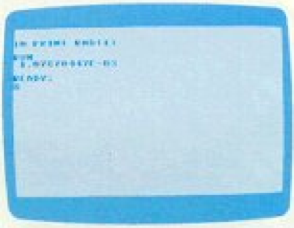
Try pressing the SHIFT key and at the same time the key marked f3. This has the same effect as typing in RUN and pressing RETURN.

Now do the same but press SHIFT and the key marked HELP. This has the same effect as typing in LIST and then pressing RETURN.

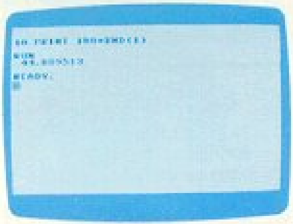
Pressing SHIFT and f1 clears the screen.



Chance



```
10 PRINT RND(1)
RUN
 [illegible]
READY.
```



```
10 PRINT 100*RND(1)
RUN
 [illegible]
READY.
```

```
10 PRINT INT(100*RND(1))
RUN
 [illegible]
READY.
```

```
READY.
```

So far you've chosen all the numbers in your programs. Using the BASIC word RND you can let your Commodore 16 pick the numbers. RND stands for random. Random means 'by chance'. Let's see how this works. Type in:

```
10 PRINT RND(1)
```

and run the program.

Your Commodore 16 will pick at random a number between 0 and 1. This means that it will be a very long decimal fraction.

Now change line 10 like this:

```
10 PRINT 100*RND(1)
```

and run this program a few times. Each time you will see a number from 0 to 100, chosen at random.

Now change line 10 to this:

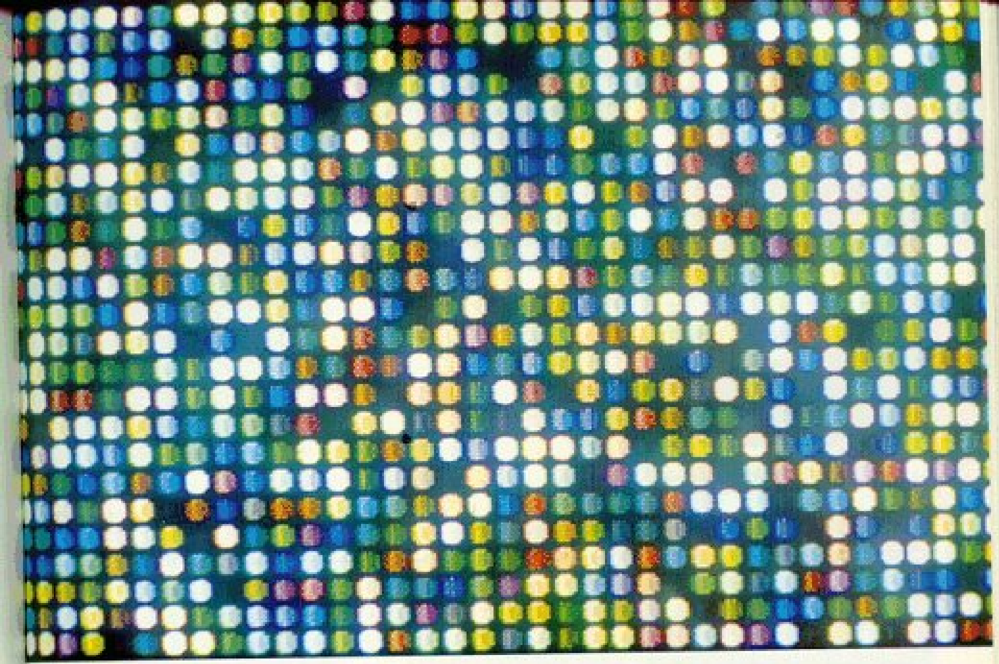
```
10 PRINT INT(100*RND(1))
```

(Make sure you put the brackets in the right place!)

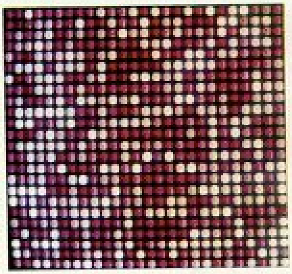
Run the program a few times. Can you see what the word INT does? It rounds the numbers down so that only whole numbers get printed.

To print out a column of random whole numbers, add these lines to your program:

```
 5 FOR R=1 TO 20
20 NEXT
```



Random colours

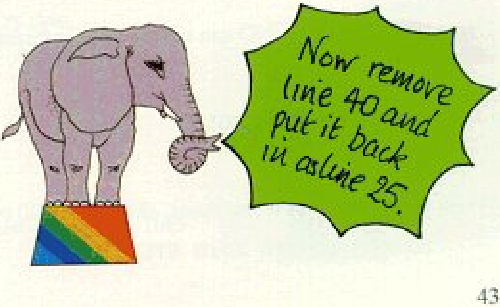


You can use RND to display colours at random. The next program shows you how this can be done. It chooses, at random, the colour and shade of the shape being printed.

Line 40 chooses a number between 0 and 16. This will choose the colour of the shape. At the end of this line you will see that 1 has been added. Since INT rounds numbers down it is possible to end with 0 as your colour, so we add one just in case. Line 50 sets the shade of the colour to be any number between 0 and 7. Type in:

```
10 SCNCLR
20 COLOR 4,1:COLOR 0,1
30 FOR N=1 TO 1000
40 C=INT(16*RND(1))+1
50 S=INT(8*RND(1))
60 COLOR 1,C,S
70 PRINT "●";
80 NEXT
```

Run the program.



Sound Effects

Your Commodore 16 can be used to play music and create sound effects. To do this you use the BASIC word SOUND. Just as each colour has its own code number, so do notes, and sounds.

The first thing to do is to set the volume or loudness of the sounds to be played. Type in:

```
VOL 8
```

The smaller
the number
after VOL,
the quieter
your notes
will be.

Now type in:

```
SOUND 1,300,60
```

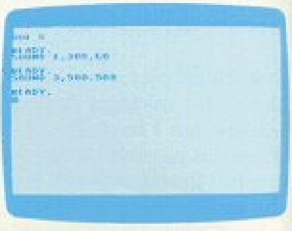
```
VOL 8

READY.
SOUND 1,300,60

READY.
```

You should hear a note played for a second. The Commodore 16 plays sound through the loudspeaker of your television. If you can't hear anything, check the volume control.

The first number following the word SOUND tells the computer what 'voice' or 'channel' to use. The second number refers to the note itself. The third number tells the computer how long the note should be played for.

Try typing this in now:

```
SOUND 3,500,500
```



```
VOL 8

READY.
SOUND 1,300,60

READY.
SOUND 3,500,500

READY.
```

The effect is quite different. The 3 here stands for the 'noise' channel. This is very useful for games and sound effects.

Loopy sounds
Using loops with SOUND can create some interesting effects. Try this program:

```
10 VOL 8
20 FOR N=0 TO 1015
30 SOUND 1,N,.5
40 NEXT
```

```
10 VOL 8
20 FOR N=0 TO 1015
30 SOUND 1,N,.5
40 NEXT
```

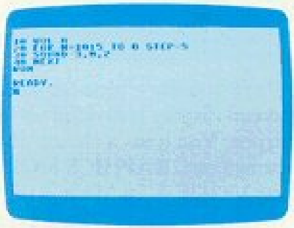
When you've run this a few times, change line 20 to this:

```
20 FOR N=0 TO 1015 STEP 5
```

Up and down

You can make your sounds go down by using a minus loop. Try this, it sounds like a rocket taking off:

```
10 VOL 8
20 FOR N=1015 TO 0 STEP-5
30 SOUND 3,N,2
40 NEXT N
```

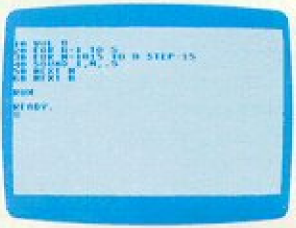


```
10 VOL 8
20 FOR N=1015 TO 0 STEP-5
30 SOUND 3,N,2
40 NEXT
RUN

READY.
```

Now add these lines:

```
50 FOR N=0 TO 1015 STEP 10
60 SOUND 3,N,1
70 NEXT N
```

As you can hear, loops can be used to help produce some startling sound effects.



```
10 VOL 8
20 FOR B=1 TO 5
30 FOR N=1015 TO 0 STEP-15
40 SOUND 1,N,.5
50 NEXT N
60 NEXT B
RUN

READY.
```

Looping the loop

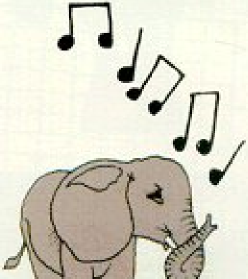
Putting one loop inside another will play a particular sound effect several times over. This next program could be one of those laser blasters!

```
10 VOL 8
20 FOR B=1 TO 5
30 FOR N=1015 TO 0 STEP-15
40 SOUND 1,N,.5
50 NEXT N
60 NEXT B
```

Random sounds

Using RND with SOUND is also fun. Try this:

```
10 VOL 8
20 FOR N=1 TO 200
30 T=INT(1015*RND(1))
40 SOUND 1,T,5
50 NEXT N
```



To hear a noise like a swarm of bees, just change the 1015 here to 115!

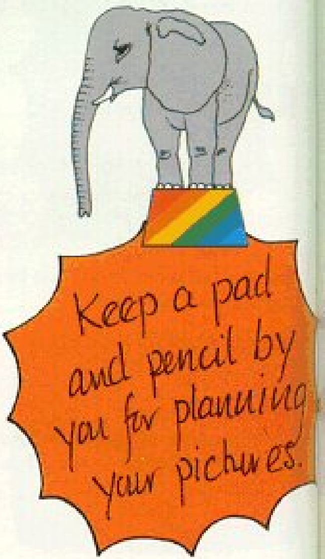
Drawing

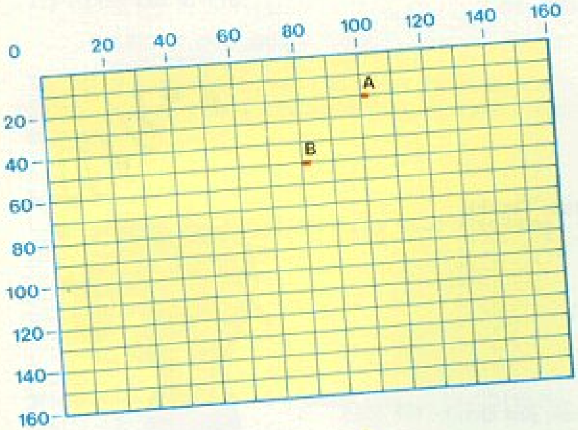
You can draw shapes and patterns with your Commodore 16. Start with simple lines and shapes, and you will soon learn how to produce striking results.

When you draw pictures with your Commodore 16, the screen display is divided up like a grid or sheet of graph paper. You have a choice of five types of screen display, known as GRAPHICS MODES. For the moment we will use MODE 4. In Mode 4 the screen display grid measures 160 by 160 squares. Each point or PIXEL on that grid has a set of numbers or co-ordinates.

When you give the co-ordinates for each pixel you count the squares across from the left first and give that number. Then you count the squares down from the top, and give that number next.

In the diagram, Point A is 100 squares along and 20 squares down. Point B is 80 squares along and 50 squares down. The co-ordinates of A are 100,20. The co-ordinates of B are 80,50.





MORFAX

Each point or dot on the screen is called a PIXEL.

In computing and maths, the squares going across are said to be on the X axis. Those going down are on the Y axis.

GRAPHIC

Before you can use Mode 4 you need to type in:

```
    10 GRAPHIC 4,1
```

```
10 GRAPHIC 4,1
20 DRAW 1,80,80

READY.
```

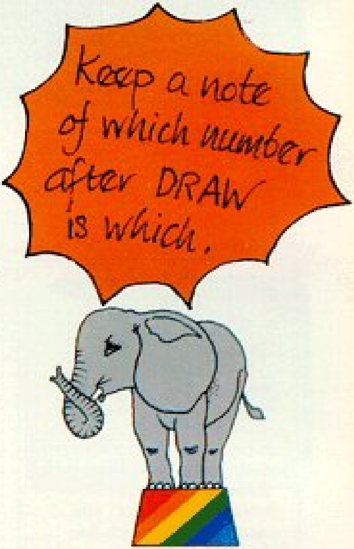
The 4 stands for the mode you are using, the 1 clears the screen before your picture is drawn.

Drawing points

To draw a small point on the screen you use the BASIC word DRAW. This is followed by three numbers. The first number says whether you want the point to be seen or not. If you do then you use the number 1. The second and third numbers are the co-ordinates of that point. To draw a point in the middle of the screen, add this line:

```
    20 DRAW 1,80,80
```

Run the program. Now try drawing some more points, all over the screen.



GRAPHIC 0

When you list the program, something odd happens. Instead of using all the screen space, the program only lists on the bottom four lines of the screen. The rest of the screen is still used for the graphics display. If you want to list the program in the normal way, type in the instruction:

```
    GRAPHIC 0
```

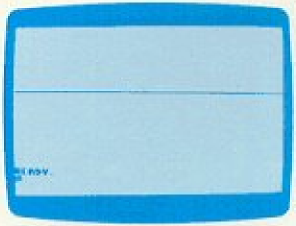
and press RETURN.

Drawing lines

To draw a line between two points, you also use DRAW. This is followed by the co-ordinates of the first point, the word TO and then the co-ordinates of the second point.

To see how this works, change line 20 to this:

```
    20 DRAW 1,0,80 TO 159,80
```



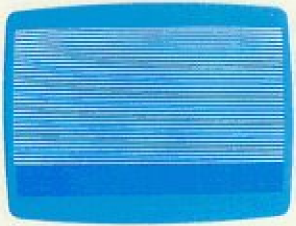
```
READY.
```

Notice that you only need two numbers after the word TO here. Have a go now at drawing some lines of your own.

Loopy lines

Just as you used loops to make sounds, you can also use them to draw pictures. In this next program a loop is used to alter the Y co-ordinate for each line drawn so that the screen fills with lines. Type in:

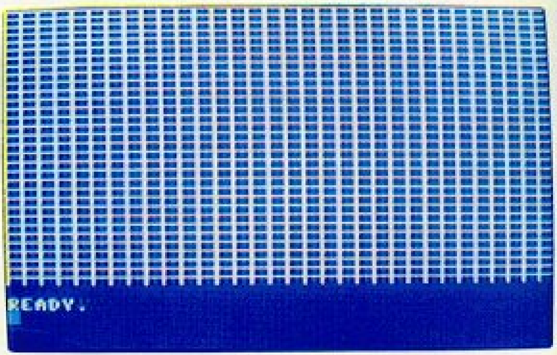
```
10 GRAPHIC 4,1
20 FOR Y=0 TO 160 STEP 5
30 DRAW 1,0,Y TO 159,Y
40 NEXT
```



Run the program. You can add colour to graphics programs in the usual way. For instance, add this line to the last program:

```
15 COLOR 0,7,2:COLOR 1,8
```

Try changing this program to draw vertical lines across the screen.



```
READY.
```

Loops can be used to draw all sorts of lines and patterns. In this next program the loop is used to set the start and end points of the line it draws. As you will see it produces a rather attractive and colourful result.

```
10 GRAPHIC 4,1
20 COLOR 4,1: COLOR 0,1
30 COLOR 1,3,3
40 FOR X=0 TO 160 STEP 4
50 DRAW 1,X,0 TO 160-X,159
60 NEXT X
```

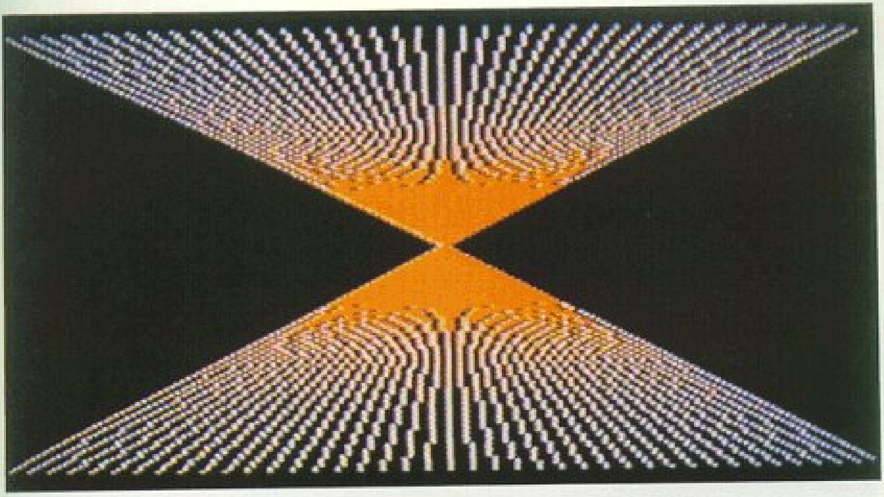
'MORFAX'

In some modes, part of the screen is reserved for graphics only (or text only). This area is then called a 'WINDOW'.

'MORFAX'

To draw very detailed pictures, the squares which make up the display grid have to be very small; so do the pixels. The computer is then said to have 'high resolution graphics'. With 'low resolution graphics', you cannot produce such fine detail.

You can buy a special device called a LIGHT PEN. With this you can draw directly on to your VDU screen. Your computer then stores that picture in its memory.

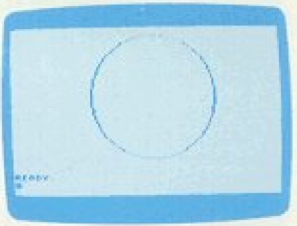


CIRCLE

There is a way to draw circles using DRAW, but it is very complicated. Luckily there is an easier way. You use the word CIRCLE.

Like DRAW, CIRCLE has to be followed by several numbers. Type these lines in:

```
10 GRAPHIC 4,1: COLOR 1,1
20 CIRCLE 1,80,80,35
```



Run the program. The first number after CIRCLE draws the circle so that it can be seen. The next two numbers refer to the co-ordinates for the centre of the circle. The last number tells the computer what the circle's radius will be.

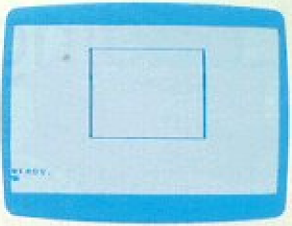
CIRCLE can be used with loops and colour too. Try this program:

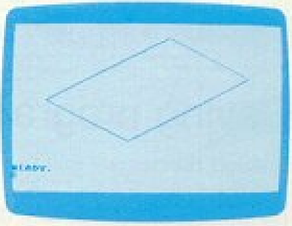
```
10 GRAPHIC 4,1
20 COLOR 0,14
30 COLOR 1,7,3
40 FOR R=3 TO 36 STEP 3
50 CIRCLE 1,80,80,R
60 NEXT
```

Drawing a box

The word BOX is used for drawing boxes. Type in these lines:

```
10 GRAPHIC 4,1:COLOR 1,1
20 BOX 1,45,30,110,130
```





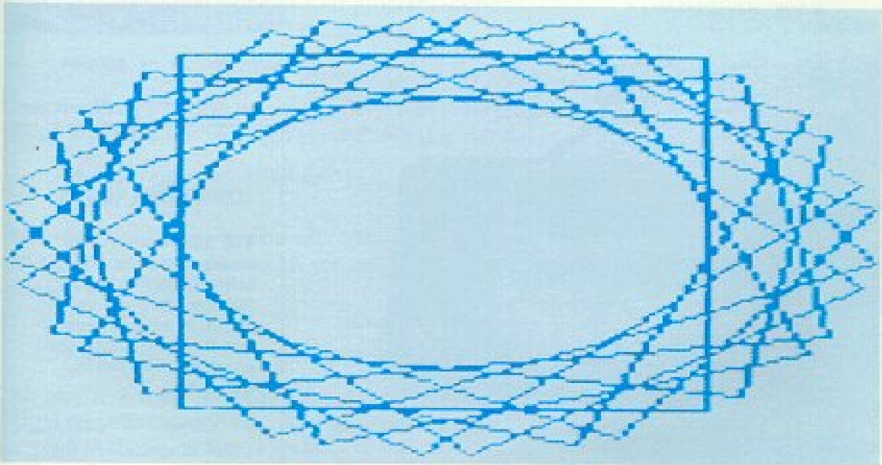
Run the program. The first number following BOX draws the box so that it can be seen. The next two numbers are the co-ordinates for the top left hand corner of the box. The last two numbers are the co-ordinates for the bottom right hand corner.

Angles
You can draw the box at an angle by adding another number to line 20 like this:

```
20 BOX 1,45,30,110,130,45
```

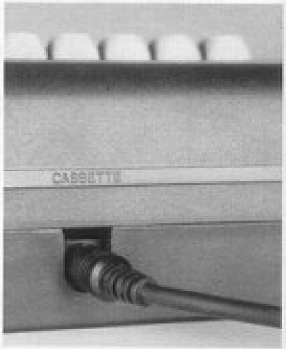
Run the program. If you change the size of the angle by putting it in a loop you will produce an interesting picture. Type in:

```
10 GRAPHIC 4,1:COLOR 1,1
20 FOR A=0 TO 360 STEP 20
30 BOX 1,45,30,110,130,A
40 NEXT
```



Loading and Saving

You don't need to type your programs in every time you want to use them. Instead you can record or SAVE them on cassette tape, then feed or LOAD them back in each time. Use the cassette unit that is supplied with the Commodore 16 and a blank cassette tape.

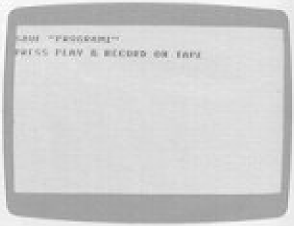


Saving programs

Connect the cassette unit to your computer. Plug the lead into the socket marked CASSETTE, at the back of the computer. The cassette unit does not need a separate power lead.

Type in a short program now, any of the ones in this book will do. Put a tape in your cassette unit. Make sure that it is wound past the leader tape at the beginning of the cassette. You now have to tell your Commodore 16 that you want to save or record the program. So, type in on a separate line:

```
    SAVE "PROGRAM1"
```



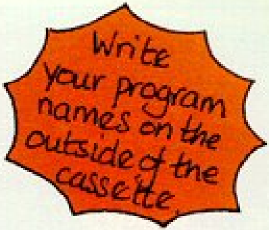
```
SAVE "PROGRAM1"
PRESS PLAY & RECORD ON TAPE
```

"NAMING"
PROGRAMS

You have to give your programs a name. This helps you to find them on the tape when you want to use them again. Try not to give the same name to two programs. This can be confusing.

We have called the program PROGRAM1 in this case, but you could give it a name of your own which describes it. Press RETURN. You will see this message on the screen:

```
PRESS PLAY & RECORD ON TAPE
```

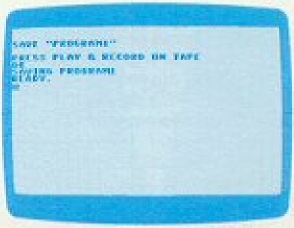


Press the RECORD and PLAY buttons on your cassette unit. The screen will go blank for a while (the time it takes will depend on the length of the program).

When the program has been saved, you will see this message on your screen:

```
SAVING PROGRAM1
READY
```

This means that your program has been saved, ie recorded successfully. The tape will have stopped automatically but it is a good idea now to press the STOP button on your cassette unit.



```
SAVE "PROGRAM1"
PRESS PLAY & RECORD ON TAPE
OK
SAVING PROGRAM1
READY.
```

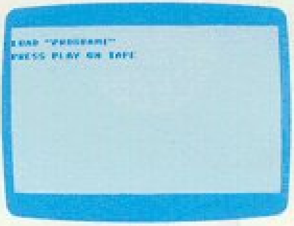
Loading programs

When you want to use a program that you've saved on tape you have to LOAD it in. Put the tape in the cassette unit. Remember to rewind it. Now type in:

```
LOAD "PROGRAM1"
```

Remember to put the name of your program inside quotation marks. You will see the message:

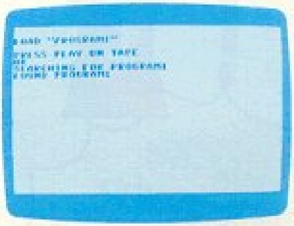
```
PRESS PLAY ON TAPE
```



```
LOAD "PROGRAM1"
PRESS PLAY ON TAPE
```

Press the PLAY button on your cassette unit. The screen will go blank for a while, then you will see this message:

```
Searching for PROGRAM1
FOUND PROGRAM1
```

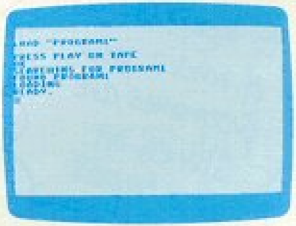


```
LOAD "PROGRAM1"
PRESS PLAY ON TAPE
OK
SEARCHING FOR PROGRAM1
FOUND PROGRAM1
```

Press the Commodore key on your keyboard. The screen will again go blank, after which you will see another message which says:

```
LOADING
READY
```

This means that the program has loaded. Press the STOP button on your cassette unit. You can now run your program in the usual way.



```
LOAD "PROGRAM1"
PRESS PLAY ON TAPE
OK
SEARCHING FOR PROGRAM1
FOUND PROGRAM1
LOADING
READY.
```

MORFAX

When you save a program on tape, your computer translates all the words in your program into a series of electronic pulses. These produce a strange noise, which you can hear if you try playing one of your tapes on a normal tape recorder. Because programs can be turned into sounds, computer programs can be transmitted in the same way as radio programmes or phone calls. It is now possible to link a telephone to your computer so that a program can be 'phoned' through directly into your computer's memory.

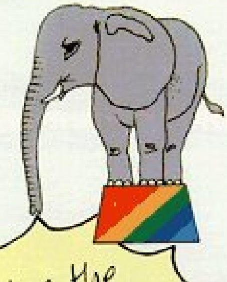
If you want to load in a program which is in the middle of a tape, and you have rewound your tape back to the beginning, your computer looks at the name of each program in turn. If the name it finds is not the program you want, you need not worry. Don't press any keys or buttons. After a short pause it will carry on automatically, looking for the next program on the tape until it finds the one you want.

To list all of the programs on a particular tape, just type in LOAD " followed by a program name you know is not on the tape. As your computer reads through the tape you will see the name of each program printed on the screen.

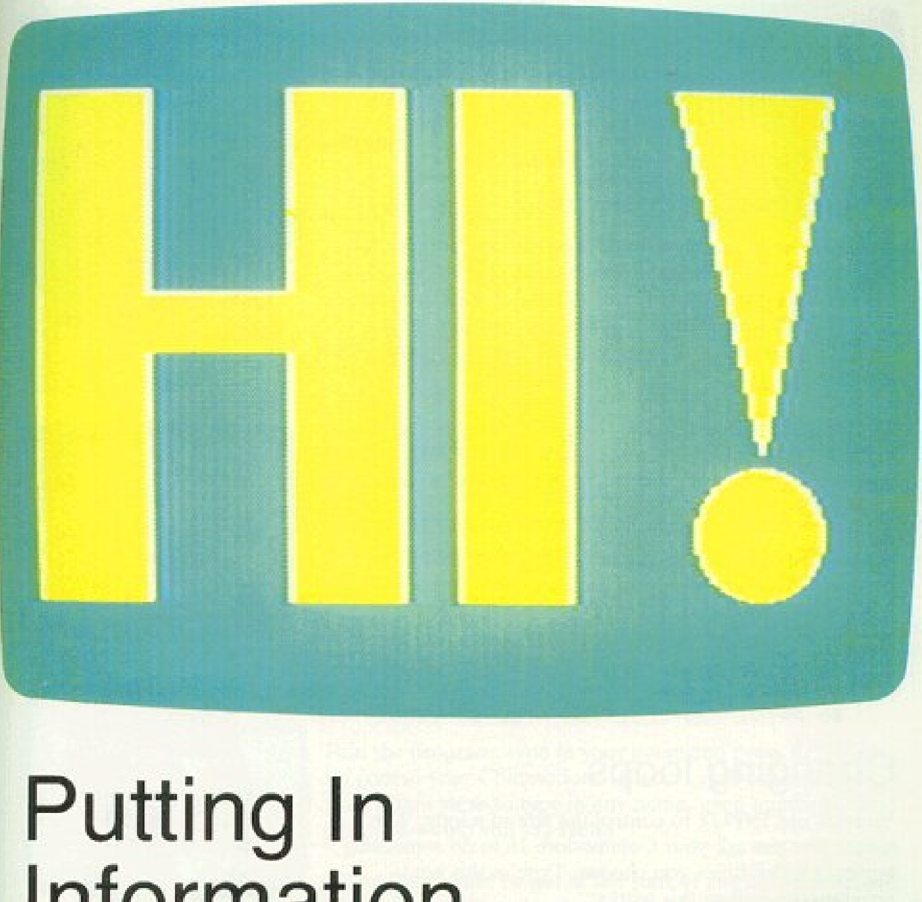
A program to save

Here is a program for you to type in and save. Can you guess what it does?

```
 10 GRAPHIC 3,1
 20 COLOR 4,13,0: COLOR 0,13,0
 30 COLOR 1,9,4
 40 BOX 1,5,5,25,190
 50 PAINT 1,10,10
 60 BOX 1,50,5,70,190
 70 PAINT 1,55,10
 80 BOX 1,25,80,50,100
 90 PAINT 1,35,90
100 BOX 1,80,5,100,190
110 PAINT 1,90,10
120 DRAW 1,120,5 TO 150,5
130 DRAW 1 TO 135,135
140 DRAW 1 TO 120,5
150 PAINT 1,130,10
160 CIRCLE 1,135,165,12
170 PAINT 1,135,165
180 GETKEY A$
190 GRAPHIC 0
200 COLOR 4,7:COLOR 0,2
210 COLOR 1,1
```



Use the tape counter to keep a record of where your programs are stored on your tapes.



Putting In Information

You don't need to be a computer programmer to be able to use a computer. People use computers all the time. They may never see the programs that they use, nor do they need to do so. When you play a computer game you probably won't see the program for it and even if you do, you don't have to understand it. This is because computers let you feed information into them directly.

INPUT

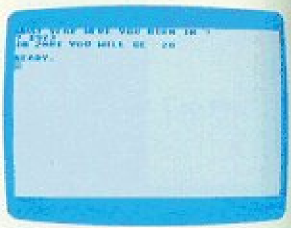
You can do this using the BASIC word INPUT. You need not then go to the trouble of listing and changing your programs. Let's look at how to use INPUT. Type in:

```
10 PRINT "GIVE ME A NUMBER"
20 INPUT A
30 PRINT "YOU TYPED IN";A
```

INPUT prints a question mark on the screen and stores whatever you type in after it.

Now run the program. You will see the question printed and, on the next line, a question mark. This is the Commodore 16's way of teling you that it wants you to put in or INPUT information. Type in your answer, press RETURN and the computer will tell you what number you typed in.

Your answer is stored in the computer's memory as the variable A, so the computer can now use that information. Run the program a few times. Each time, put in a different answer to see what happens.



```
WHAT YEAR WERE YOU BORN IN ?
? 1973
IN 2001 YOU WILL BE  28

READY.
```

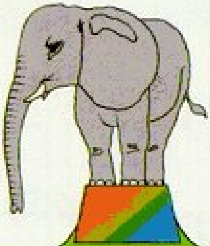
All sorts of information can be INPUT and used in your program. This next program is called '2001'. Type it in and run it to see why:

```
10 SCNCLR
20 PRINT "WHAT YEAR WERE YOU BORN IN ?"
30 INPUT YEAR
40 PRINT "IN 2001 YOU WILL BE";2001-YEAR
```

Changing loops

You can use INPUT to control the size of a loop. This means you can ask your Commodore 16 to do something however many times you choose. Type in this next program to see how this works:

```
10 SCNCLR
20 PRINT "HOW MANY PRINTINGS ?"
30 INPUT N
40 FOR X=1 TO N
50 PRINT "COMMODORE 16 RULES OK";
60 NEXT X
```



REMEMBER! Always press RETURN after answering a question.

Now run the program. Don't forget to press RETURN after you've typed in your number. Your message will be printed as many times as you choose. Why not change the message at line 50?

This next program is a more serious use of INPUT. You ask your Commodore 16 to print out the multiplication table of any number you choose. The number is stored as the variable NUM. Type in:

```
10 SCNCLR
20 PRINT "CHOOSE A NUMBER"
30 INPUT NUM
40 SCNCLR
50 PRINT "THE";NUM;"TIMES TABLE"
60 PRINT
70 FOR T=1 TO 12
80 PRINT NUM;"*";T;"=";NUM*T
90 NEXT
```

```
[illegible]
READY.
```

Run the program, type in your number and then press RETURN. The program will print out the multiplication table for your number.

```
HELLO, WHO ARE YOU ?
? DIANA
```

```
HELLO DIANA
READY.
```

Putting in words

You can also INPUT words and strings. This is almost exactly the same as INPUTTING numbers except that you have to remember to use the $ sign after the variable name. Try this now:

```
10 SCNCLR
20 PRINT "HELLO, WHO ARE YOU ?"
30 INPUT NAME$
40 SCNCLR
50 PRINT "HELLO ";NAME$
```

Run the program, type in your name and press RETURN. Of course your Commodore 16 hasn't really recognized you. If you were to type in any name, even numbers, the program would still say Hello.

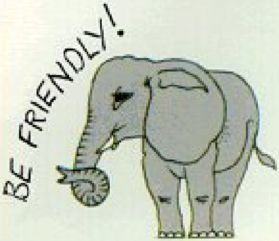
These lines could be put at the start of any of the programs you write. It will make them more USER FRIENDLY.

MORFAX

You can use INPUT on its own without PRINT. To do this, type in:

```
10 INPUT "WHO ARE YOU ";NAME$
20 SCNCLR
30 PRINT "HELLO ";NAME$
```

This saves memory space and is easier to type in.



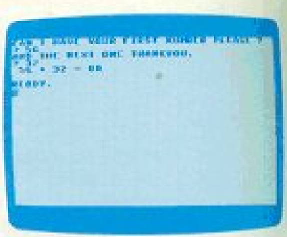
Calculator programs

Thanks to INPUT you can write a program to do your addition sums. When you run it, all you need to do is type in the numbers you want added up.

Try to get the spacing right so that your screen display looks neat. Type in:

```
   10 SCNCLR
   20 PRINT "CAN I HAVE YOUR FIRST NUMBER
PLEASE ?"
   30 INPUT N1
   40 PRINT "AND THE NEXT ONE THANK YOU."
   50 INPUT N2
   60 PRINT N1;"+";N2;"=";N1+N2
```

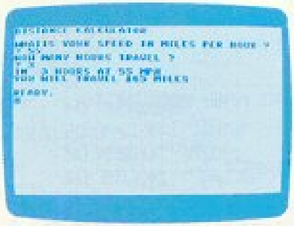
Run the program now. Press RETURN after you've given each number. Try it several times, then have a go at writing a program that adds three numbers together.



```
CAN I HAVE YOUR FIRST NUMBER PLEASE ?
? 56
AND THE NEXT ONE THANKYOU.
? 32
 56 + 32 = 88

READY.
```

"MORFAX"

In this last program you don't really need lines 20 or 40. They make the program easier for people to use and understand. You should always aim to make your programs USER FRIENDLY.



```
DISTANCE CALCULATOR

WHAT IS YOUR SPEED IN MILES PER HOUR ?
? 55
HOW MANY HOURS TRAVEL ?
? 3
IN  3 HOURS AT 55 MPH
YOU WILL TRAVEL 165 MILES

READY.
```

Speedometer

This next program uses INPUT with multiplication. It tells you how far you would go in a certain time, at a particular speed. Type in:

```
   10 SCNCLR
   20 PRINT "DISTANCE CALCULATOR":PRINT
   30 PRINT "WHAT IS YOUR SPEED IN MILES PER
HOUR ?"
   40 INPUT SPEED
   50 PRINT "HOW MANY HOURS TRAVEL?"
   60 INPUT NUM
   70 PRINT "IN";NUM;"HOURS AT";SPEED;  "MPH"
   80 PRINT "YOU WILL TRAVEL";SPEED*NUM;
"MILES"
```

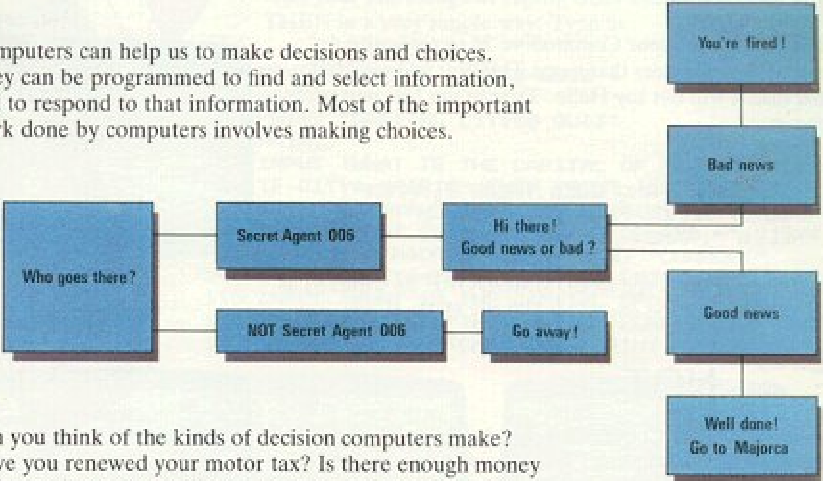
The screen display title in line 20 makes this program more USER FRIENDLY.

"MORFAX"

What makes computers so useful is the speed with which they can sort through information. Today, large mainframe computers are used by many organizations to store their records. Instead of having to spend hours searching through shelves full of files, facts and figures can be easily collected and checked using the power of computers.

Making Decisions

Computers can help us to make decisions and choices. They can be programmed to find and select information, and to respond to that information. Most of the important work done by computers involves making choices.

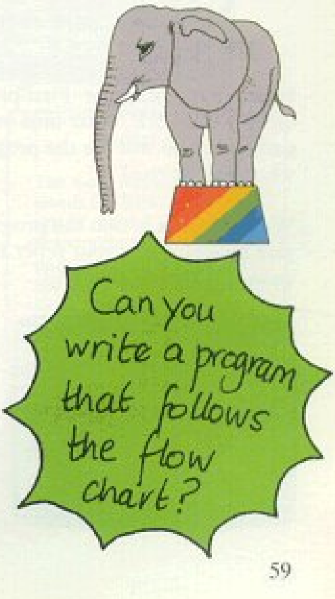


Can you think of the kinds of decision computers make? Have you renewed your motor tax? Is there enough money in the bank to pay the bills? Are there any seats on that holiday flight to Bermuda?

When a computer makes a decision it checks through the information stored in its memory. IF it finds the number or words it is looking for, THEN it will carry out one set of instructions. IF it doesn't, THEN it will carry out another set of instructions.

Because of this, many programs contain not just one but many sets of instructions. They become rather like road maps showing all sorts of different routes and destinations. Depending on the answers and responses you give to the computer, you may be given a number of different answers.

Many computer programmers actually draw maps or diagrams of the programs they write. These are known as FLOW CHARTS. They help the programmer to plan programs.

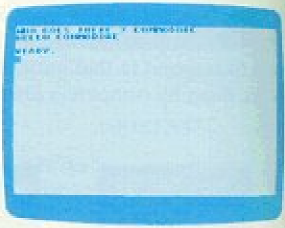


IF . . . THEN

To program your Commodore 16 to make decisions, you use the BASIC words IF and THEN. Let's look at a simple decision program.

We shall program your Commodore 16 to recognize a name. IF it recognizes that name THEN it will say Hello. Otherwise it will not say Hello. Type in the first part of the program:

```
    10 SCNCLR
    20 INPUT "WHO GOES THERE ";NAME$
    30 IF NAME$="COMMODORE" THEN PRINT
 "HELLO ";NAME$
```



```
WHO GOES THERE ? COMMODORE
HELLO COMMODORE

READY.
```

Run the program. Type in COMMODORE (in capitals) as your answer, and the computer will say Hello to you.

IF . . . NOT

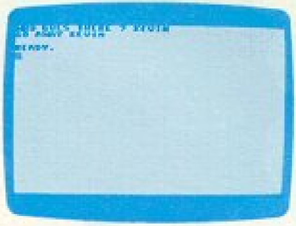
Now let's tell your Commodore 16 what to do IF your name is NOT Commodore. This time you use the BASIC word NOT after the word IF. Add this line:

```
    40 IF NOT NAME$="COMMODORE" THEN PRINT
 "GO AWAY ";NAME$
```

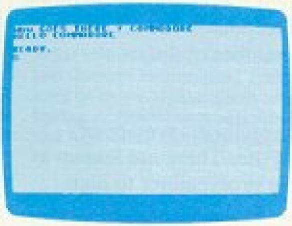
```
10 SCNCLR
20 INPUT "WHO GOES THERE ";NAME$
30 IF NAME$="COMMODORE" THEN PRINT "HEL
LO ";NAME$
40 IF NOT NAME$="COMMODORE"THEN PRINT "G
O AWAY ";NAME$
```

Now run the program. First pretend that your name is COMMODORE. Next time you run it, type in your own name. As you will see the program is not quite so friendly when you do this!

What happens is that the program decides which answer to give by looking at your reply and comparing it with the answer 'COMMODORE'.



```
WHO GOES THERE ? KEVIN
GO AWAY KEVIN

READY.
```



```
WHO GOES THERE ? COMMODORE
HELLO COMMODORE

READY.
```

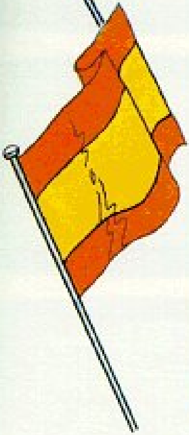
"MORFAX"

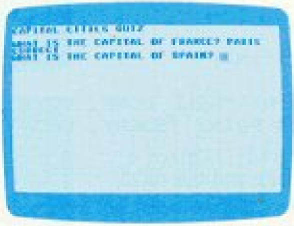
Your Commodore 16 treats capital letters as different from lower case (little) letters. It will not regard 'Commodore' as being the same as 'COMMODORE'.

Quiz programs

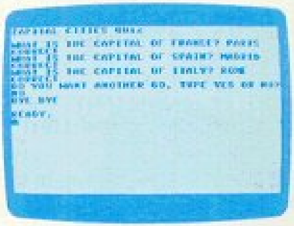
You can use IF and THEN to test your friends' general knowledge. This next program is a simple quiz program. It tests your knowledge of capital cities and uses IF and THEN in a very simple way. Type in:

```
 10 REM * CAPITAL CITIES QUIZ *
 20 SCNCLR
 30 PRINT "CAPITAL CITIES QUIZ"
 40 PRINT
 50 INPUT "WHAT IS THE CAPITAL OF FRANCE";CITY$
 60 IF CITY$="PARIS" THEN PRINT "CORRECT"
 70 IF NOT CITY$="PARIS" THEN PRINT "NO, WRONG."
 80 INPUT "WHAT IS THE CAPITAL OF SPAIN";CITY$
 90 IF CITY$="MADRID" THEN PRINT "CORRECT"
100 IF NOT CITY$="MADRID" THEN PRINT "NO,WRONG."
110 INPUT "WHAT IS THE CAPITAL OF ITALY";CITY$
120 IF CITY$="ROME"THEN PRINT "CORRECT"
130 IF NOT CITY$="ROME" THEN PRINT "NO,WRONG"
```





```
CAPITAL CITIES QUIZ

WHAT IS THE CAPITAL OF FRANCE? PARIS
CORRECT
WHAT IS THE CAPITAL OF SPAIN? ▮
```



```
CAPITAL CITIES QUIZ

WHAT IS THE CAPITAL OF FRANCE? PARIS
CORRECT
WHAT IS THE CAPITAL OF SPAIN? MADRID
CORRECT
WHAT IS THE CAPITAL OF ITALY? ROME
CORRECT
DO YOU WANT ANOTHER GO, TYPE YES OR NO?
NO
BYE BYE

READY.
▮
```

Why not add some more lines to this program, or write a quiz of your own? Your quiz could be about all sorts of things, from football to pop music.

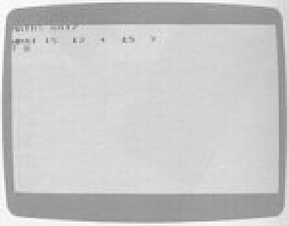
Another go?
Using IF and THEN with the BASIC word GOTO, you can program your computer to let you have another go at your quiz. The next few lines (provided that you alter the line numbers) can be put at the end of any of your programs. They allow you to have another go at running the program. Add them to the program you have just written:

"MORFAX"

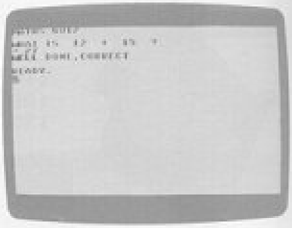
The word REM at line 10 stands for REMARK. The rest of the line is a description of the program. This line doesn't affect the program but acts like a title, so that when you look at the listing you know what the program is about. Using REM like this helps make your listings clearer.

```
140 INPUT "DO YOU WANT ANOTHER GO, TYPE YES OR NO ";AN$
150 IF AN$="YES" THEN GOTO 10
160 IF NOT AN$="YES" THEN PRINT "BYE BYE"
```

Test your maths



```
MATHS QUIZ
WHAT IS 12 + 15 ?
?
```



```
MATHS QUIZ
WHAT IS 12 + 15 ?
WELL DONE,CORRECT
READY.
```

You can use IF . . . THEN with numbers too. The last few pages in this book will show you how you can write a program that uses many of the BASIC words and instructions we've looked at so far. It will also introduce some useful new signs.

The next program tests your maths skills! It asks you to add two numbers together. It then tells you whether or not you have given the right answer. Type in:

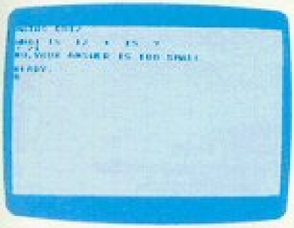
```
 10 REM * MATHS QUIZ *
 20 SCNCLR
 30 COLOR 4,15,1:COLOR 1,1
 40 PRINT "MATHS QUIZ"
 50 PRINT
 60 LET N1=12
 70 LET N2=15
 80 LET RIGHT=N1+N2
 90 PRINT "WHAT IS";N1;"+";N2;"?"
100 INPUT ANS
110 IF ANS=RIGHT THEN PRINT "WELL DONE, CORRECT"
120 IF NOT ANS=RIGHT THEN PRINT "SORRY, WRONG"
```

Run the program now, typing in the wrong and the right answer. Can you see how the right answer has been stored as a variable, at line 80? When you want to make the quiz more difficult, just change the numbers stored as N1 and N2.

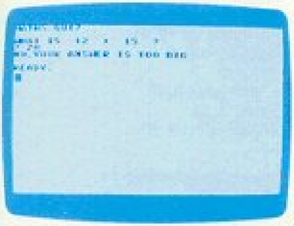
Too big or too small?
You can change this program so that it gives you a little more help. It can tell you whether your wrong answer is too big or too small. To do this you use the symbols < and > (bottom row of the keyboard). The > sign stands for greater or bigger than. The < stands for less or smaller than.

```
   120 IF ANS>RIGHT THEN PRINT "NO,YOUR ANSWER IS TOO BIG. TRY
AGAIN":GOTO 90
   130 IF ANS<RIGHT THEN PRINT "NO,YOUR ANSWER IS TOO SMALL. TRY
AGAIN":GOTO 90
```

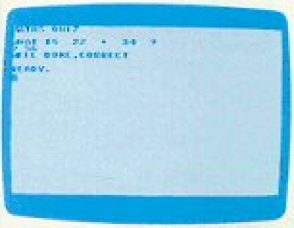
GOTO sends your program back to line 90 so that you can have another go if you get the answer wrong. Run the program a few times now, giving the right and wrong answers.



```
[illegible]
WHAT IS  12  +  15  ?
? [illegible]
NO,YOUR ANSWER IS TOO SMALL
READY.
```



```
[illegible]
WHAT IS  12  +  15  ?
? [illegible]
NO,YOUR ANSWER IS TOO BIG
READY.
```



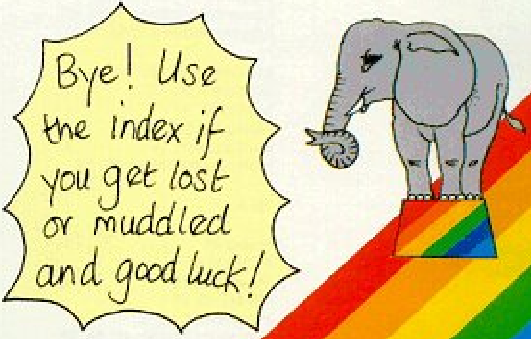
```
[illegible]
WHAT IS  22  +  34  ?
? 56
WELL DONE,CORRECT
READY.
```

Lucky draw

Why not let your Commodore 16 choose the numbers for you and make things more interesting? You can use RND with the numbers in this program. Change lines 60 and 70 like this:

```
60 LET N1=INT(50*RND(1))+5
70 LET N2=INT(50*RND(1))+5
```

Each time you run it now you'll get a different set of numbers and a different sum to do. If you want to use bigger numbers, decide on the range you want. Use that number at lines 60 and 70 and away you go. To make it even harder, remove the word INT from both lines!



Bye! Use the index if you get lost or muddled and good luck!

Well, that's all for now. I hope you've enjoyed this book and that it has given you lots of ideas. Don't stop now though. You've only just started! Be adventurous, start to write your own programs. Be prepared for hard work, a few mistakes and many hours of fun!

Index