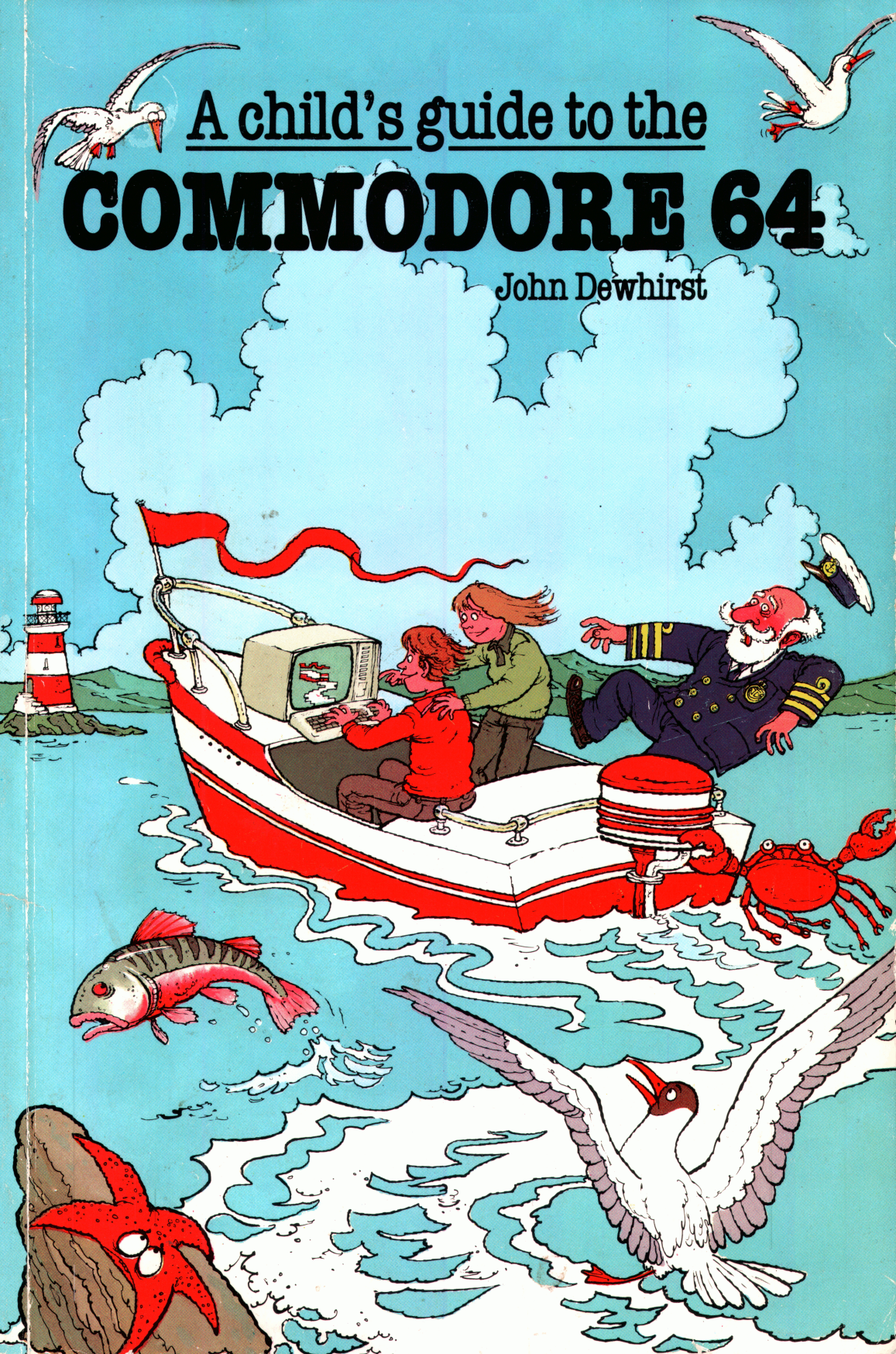
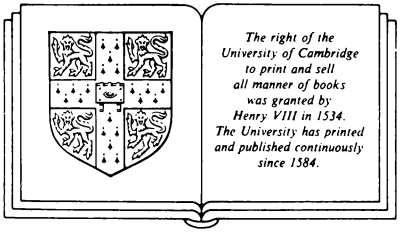
A child's guide to the COMMODORE 64

John Dewhirst



A CHILD'S GUIDE TO THE COMMODORE 64

John Dewhirst



CAMBRIDGE UNIVERSITY PRESS
Cambridge
London New York New Rochelle
Melbourne Sydney

Published by the Press Syndicate of the University of Cambridge
The Pitt Building, Trumpington Street, Cambridge CB2 1RP
32 East 57th Street, New York, NY 10022, USA
10 Stamford Road, Oakleigh, Melbourne 3166, Australia

First published 1985

British Library cataloguing in publication data

Dewhirst, John
A child's guide to the Commodore 64.
1. Commodore 64 (Computer) – Juvenile literature
I. Title
001.64'04 QA76.8.C64
ISBN 0 521 31538 7

Produced by Chancerel Publishers Ltd, 40 Tavistock Street,
London WC2.
Designed by Valerie Sargent.
Cover and cartoons by David Parkins.
Printouts by James Ryan.
Typesetting and graphic reproduction by ReproSharp Ltd, 47 Farringdon Road, London EC1.
Printed in Great Britain at
the University Press, Cambridge.

Acknowledgements
The author would like to thank Mr H. Morris and Mr P. Fox for their help in producing this book.

CONTENTS

INTRODUCTION

Each section of the book is written by a different person. Each person is an expert at their job and gives very useful advice. Read their advice very carefully. The advice is given at the start of their section of the book. If you follow the advice it will make the work you do much easier.

SECTIONS

1 The keyboard
by Pru Comet – typist.
Pru Comet shows you the ways in which the various types of keys work and the position of the keys on the keyboard.

2. Getting started
by P. C. Truemo – investigator.
Constable Truemo shows you how to get started on the computer, how to make it print numbers and words, calculate answers and finally store items in its memory.

3. Writing programs
by Prof. O. Crumpet – designer.
Professor O. Crumpet shows you how to copy, adapt, then finally design your own programs.

4. Special features
by Mort Puce – artist.
Mort Puce shows you how to draw coloured pictures on the screen, and how to use the function keys and real-time clock.

5. Finding out
by Ms O. C. Termup – librarian.
Ms O. C. Termup shows you how to use the reference sections.

Mep Court,
Portcume.

Dear Reader,

Hello there. We are the people who have written this book and you will find out more about us as you read through it. When you meet us we shall introduce ourselves.

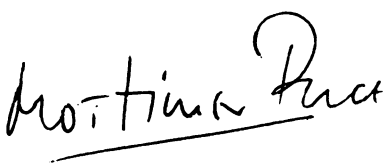
The book has been written for people who know nothing about computers, but who want to find out for themselves. It has been written to show you not just what the computer can do, but more importantly, what *you* can make the computer do for you.

We have put together all our good ideas so that it is easy for you to find out just how the Commodore 64 micro works. The book is written in five sections and we have each written a section. The first four sections are to be read and worked through in order, but the last section is for reference and this can be used at any time for checking.

Happy computing!

Prudence Comet

Bobby Truemo



Mortimer Puce

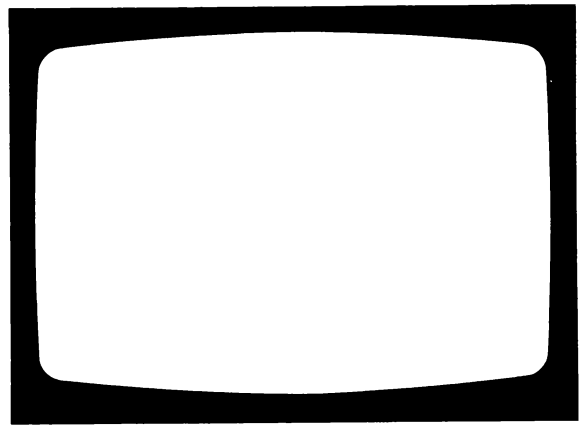
Oliver Crumpet

Olive Constance Termup

In each section of the book you will come across different types of diagrams. The things for you to try, that is various exercises and experiments, will all be given a 'You try' box.
The 'You try' box looks like this.

You try

The things which are displayed on the screen will all be shown in a 'Display' box. The 'Display' box looks like this.



The things which you need to make a note about and remember will all be given in a 'Make a note' box. The 'Make a note' box looks like this.

Make a note

THE KEYBOARD

Pru Comet

My advice

Don't be afraid of the computer. I think of it as a typewriter with a difference. Try things out. In this way you will learn quickly just what it can and cannot do. When you try things out, the computer will send messages back to you. This is the way the computer talks to you. It can make noises, change colours on the screen and do lots of other things, so you'll need a bit of time to learn about everything the computer can do. Your first job is to learn about the keyboard, find out where all the keys are and what they all do. A good idea is to make your own notes in a book of all the things you find out about the computer.

As you read through the keyboard section of the book, try things out as you go along. Don't be put off by any messages the computer sends to you. It does not realise that you are a beginner and just trying out a few things for yourself. As you work through the rest of the book you will learn the language that the computer understands, but like learning any language you can only learn a bit at a time.

Pru Comet

There are 66 keys on the Commodore 64 micro, and they can be divided into the following six groups:

Control keys	Edit keys
Number keys	Symbol keys
Letter keys	Function keys

But before you starting looking at the keys in detail there are a couple of things you need to know.

1. You can use INST/DEL as a rub out key. That is, by touching INST/DEL any unwanted characters can be removed. But beware, this key repeats itself, and if you keep your finger on it you could rub out more than you intended.

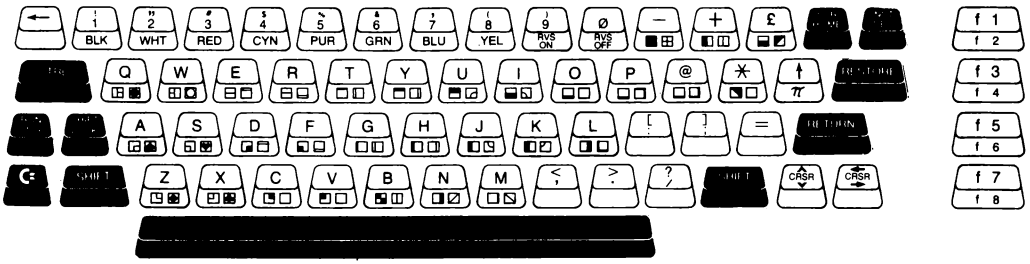
2. If the computer continues to go over something again and again and you want to stop this then press RUN/STOP.

The function keys are different from the other keys: they do not do anything until you have told the computer what you want them to do. So we look at them in the 'Special features' chapter on page 74.

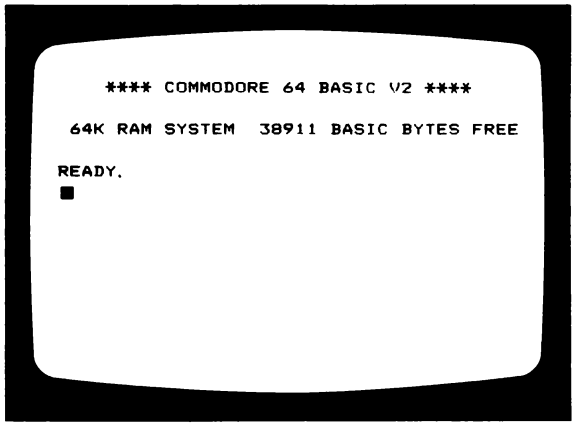
Control keys

There are eleven of these keys, ten of which have words in white on them (the eleventh is the long key at the bottom of the keyboard).

RETURN	INST DEL
SHIFT (two keys)	RESTORE
SHIFT LOCK	RUN STOP
CLR HOME	CTRL
SPACE BAR	C=

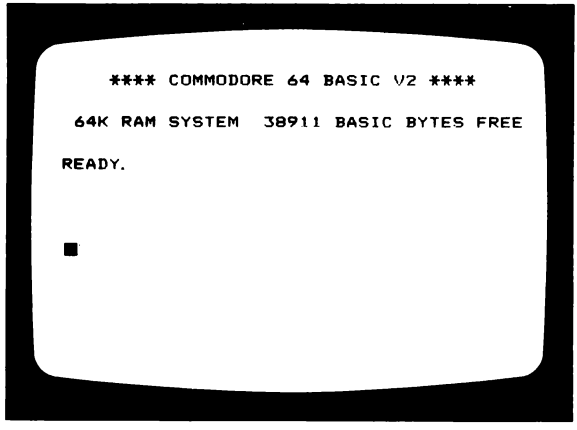


When you first switch the computer on, the screen shows:



RETURN moves the cursor to the start of the next line. The cursor is a flashing square which shows you where the next character you type will appear on the screen.

This is what will appear on the screen:



You try

Press RETURN three times.

Make a note

RETURN moves the cursor to the next line.

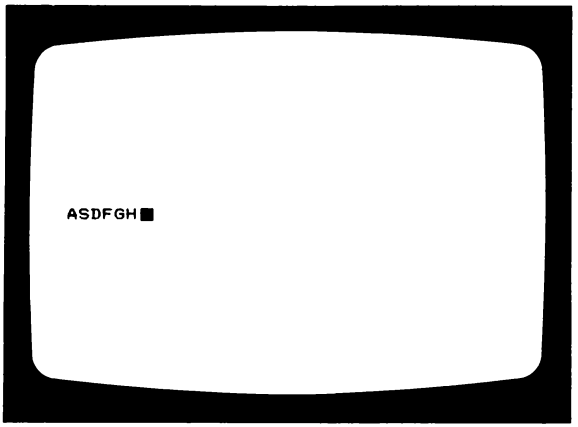
SHIFT If either of these keys is held down, a letter key will print the graphic character shown on the right of the key and a symbol or number key will print the upper symbol. If you let SHIFT go, you will get letters, numbers or lower symbols again.

SHIFT LOCK If this key is pressed once, the right-hand graphic characters or upper symbols will be printed without the trouble of holding down SHIFT . Pressing SHIFT LOCK a second time gives you letters, numbers or lower symbols again.

You try

Press in turn each of the keys marked ASDFGH

This is what will appear on the screen:

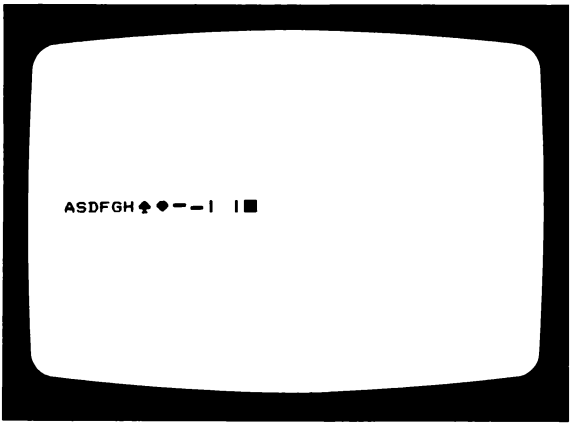


You try

Press SHIFT LOCK and type ASDFGH again.

Now press SHIFT LOCK a second time so that you can print letters again.

This is what will appear on the screen:

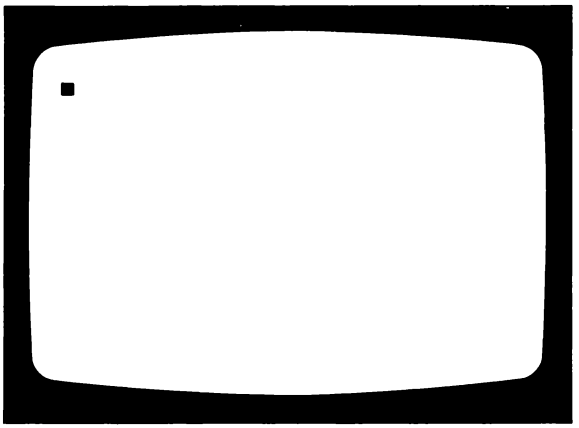


Make a note

When SHIFT LOCK is on, or when SHIFT is held down, you can print the right-hand graphic characters or upper symbols.

CLR HOME When this key is pressed, the cursor will jump to the top left-hand corner of the screen. If you hold down SHIFT , or if SHIFT LOCK is on, pressing CLR/HOME will clear the screen as well as moving the cursor to the top left of the screen.

This is what will appear on the screen:

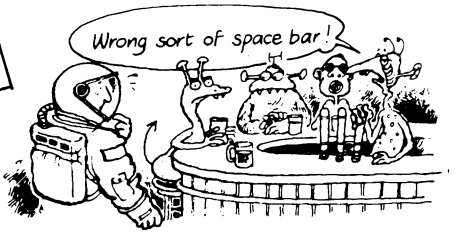


You try

Hold down SHIFT and press CLR/HOME .

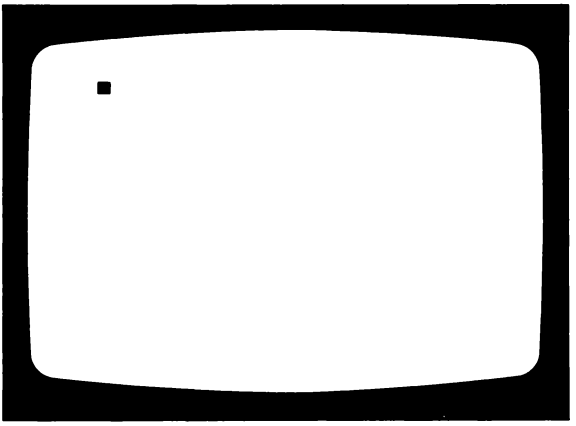
Make a note

Holding down SHIFT and pressing CLR/HOME clears the screen.



SPACE BAR This is the long key at the bottom of the keyboard. It is not named like the other keys. It puts in a space.

This is what will appear on the screen:



You try

Press the SPACE BAR (long key) carefully three times.

Notice that if you hold down the [SPACE BAR], it will repeat itself. Hold the [SPACE BAR] down until the cursor goes onto the next line.

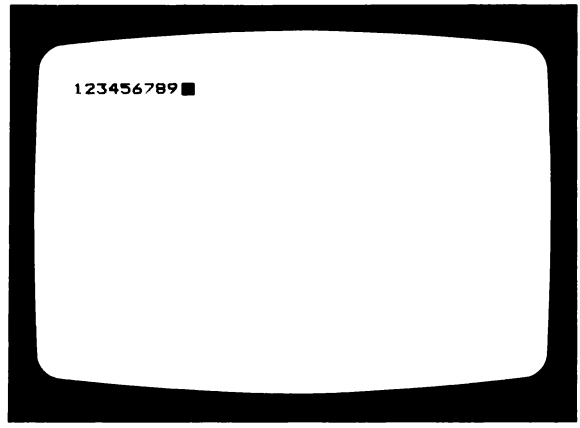
When you are ready, press [CLR/HOME] to put the cursor back at the top of the screen.

[INST DEL] rubs out the character to the left of the cursor.

You try

Type 123456789

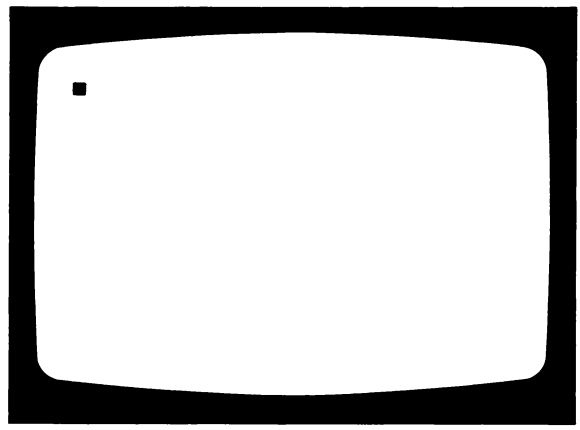
This is what will appear on the screen:



You try

Press [INST DEL] nine times.

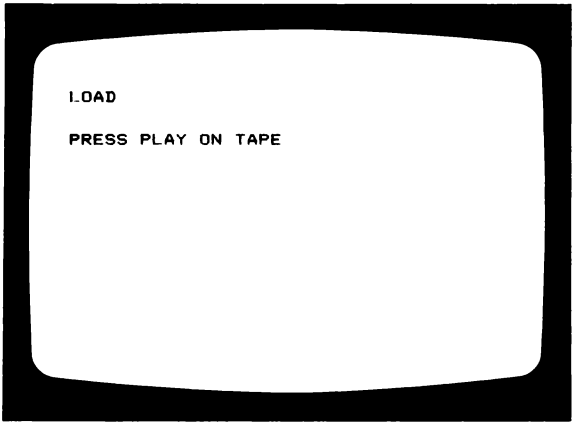
This is what will appear on the screen:



[RESTORE] is used to return the computer to the state it was in before it carried out a set of instructions. Sometimes [RESTORE] is needed along with [RUN/STOP] to make the computer stop what it is doing.

RUN STOP This key is used to break out of what the computer is doing. We will meet it in the chapter on 'Writing programs'. If SHIFT is held down and RUN/STOP is pressed, the computer will automatically try to load a program from tape.

This is what will appear on the screen:

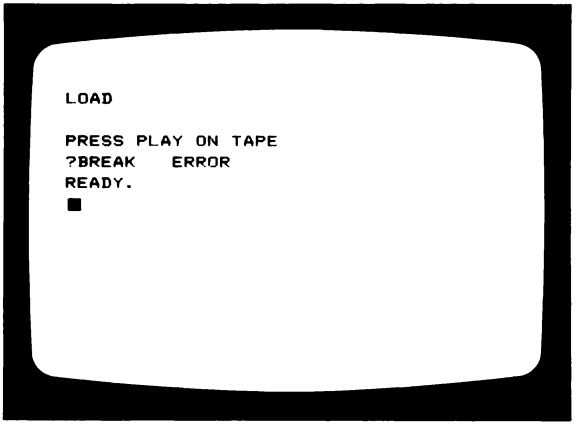


You try

Hold down SHIFT and press RUN/STOP .

For now, we are not going to load a games program! So we should stop the computer looking for a program.

This is what will appear on the screen:



You try

Press RUN/STOP without holding down SHIFT .

Do not worry about the message saying '?BREAK ERROR' – nothing has been broken! The computer is just telling you that its search for a program has been interrupted.

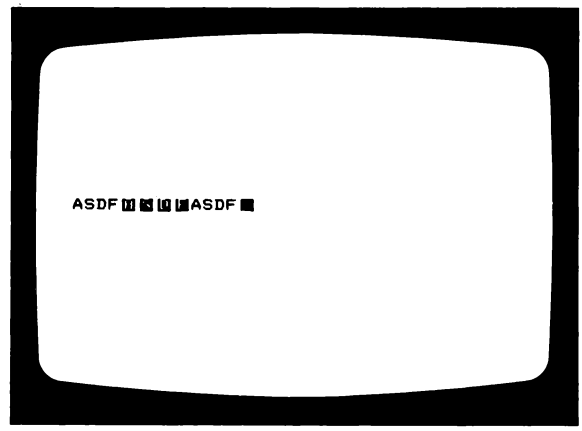
CTRL When pressed together with other keys, this key carries out various special jobs. It is most often used to choose the colours in which characters are printed on the screen, as we will see in the chapter on 'Special features', page 66.

CTRL is also used to switch between 'true video' and 'reverse video' styles of printing characters on the screen.

You try

Type ASDF. Now hold down CTRL and press key 9. Type ASDF again, and notice the difference. Hold down CTRL and press key Ø. Type ASDF a third time.

This is what will appear on the screen:

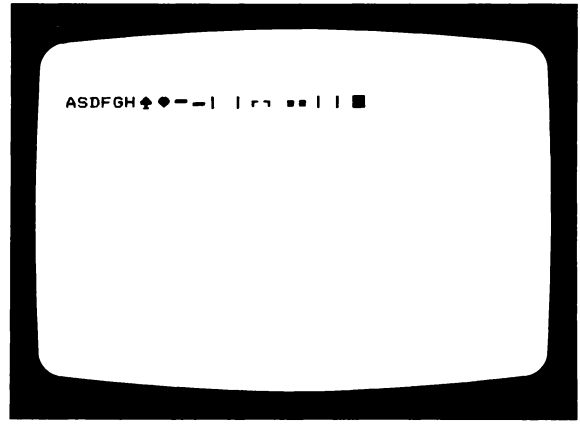


C= If this key is held down, a letter key will print the graphic character shown on the left of the key. If you hold down SHIFT and press C= once, the letter keys will produce small letters instead of capitals. Now holding down SHIFT again will produce capital letters.

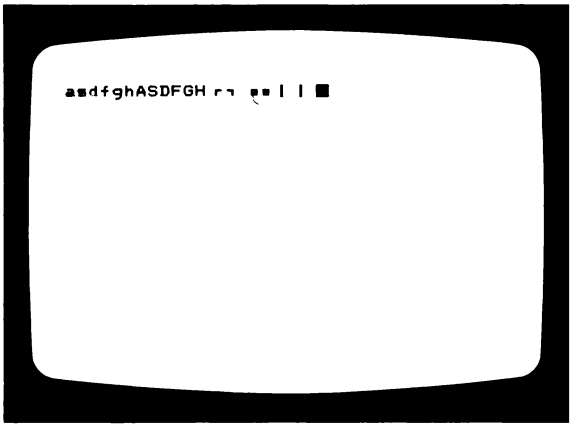
You try

Press SHIFT and CLR/HOME to clear the screen. Type ASDFGH. Hold down SHIFT and type ASDFGH again. Now release SHIFT, hold down C=, and type the same letters a third time.

This is what will appear on the screen:



This is what will appear on the screen:



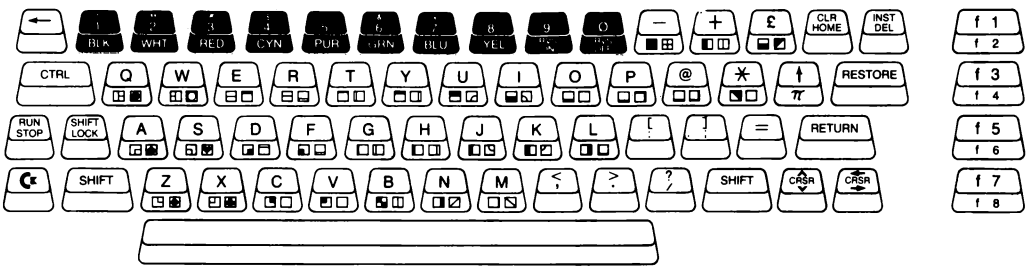
You try

Press C and SHIFT together.

Pressing C and SHIFT together a second time will turn small letters back into capitals and capitals into right-hand graphic characters. But left-hand graphic characters still stay the same.

Number keys

These are the ten keys which print the numbers 1, 2, 3, 4, 5, 6, 7, 8, 9, Ø.
On the keyboard they are set out in a line:
1 2 3 4 5 6 7 8 9 Ø
(The Ø key is used for zero so that it is not confused with the letter O.)



You try

Type out these figures. Press RETURN at the end of each line.

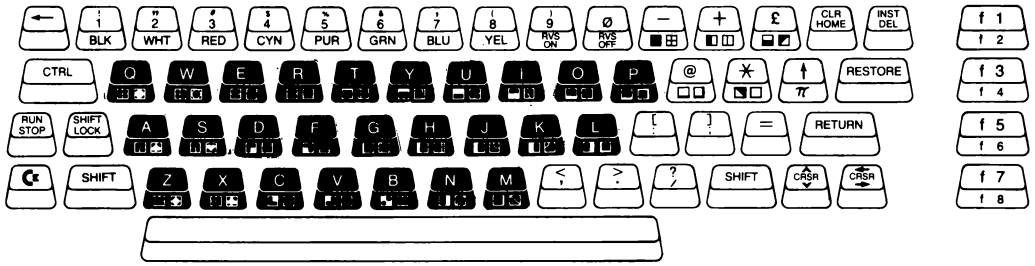
1
121
1331
14641

Letter keys

These are the 26 keys which print the letters of the alphabet.
A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z
and
a, b, c, d, e, f, g, h, i, j, k, l, m, n, o, p, q, r, s, t, u, v, w, x, y, z.
On the keyboard they are set out in capitals as

Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M

which is exactly the same as you would find them on a typewriter.



You try

Type out the following. Press [RETURN] at the end of each line.
1ST ROW QWERTYUIOP
2ND ROW ASDFGHJKL
3RD ROW ZXCVBNM

You try

Type out the names of the members of your family and their birthdays. Press [RETURN] at the end of each line.

You try

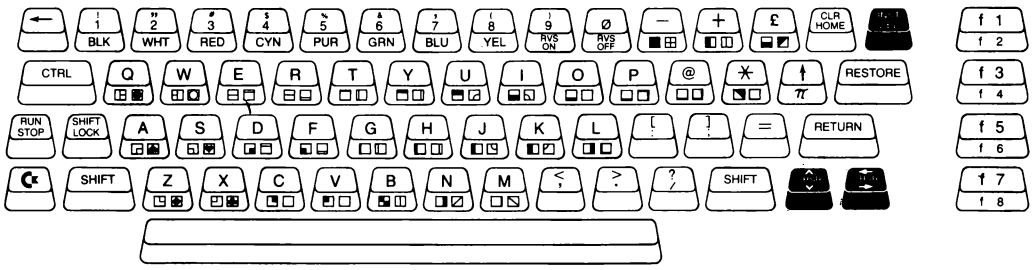
Type out:
I can use the keyboard.
Press [RETURN] .

If you do not like the computer saying '?SYNTAX ERROR' then put a number at the beginning of each line.

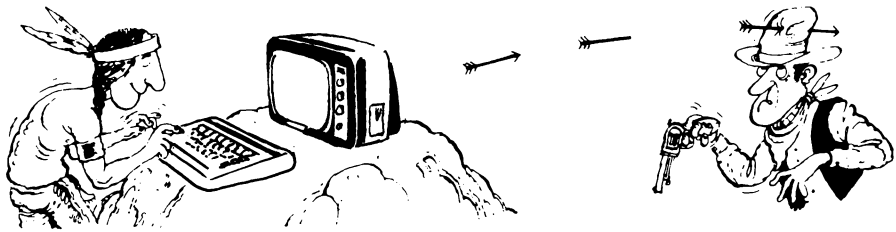
By pressing the [C=] and [SHIFT] keys and then using the [SHIFT] key at the right times, try printing in a mixture of capital and small letters.

Edit keys

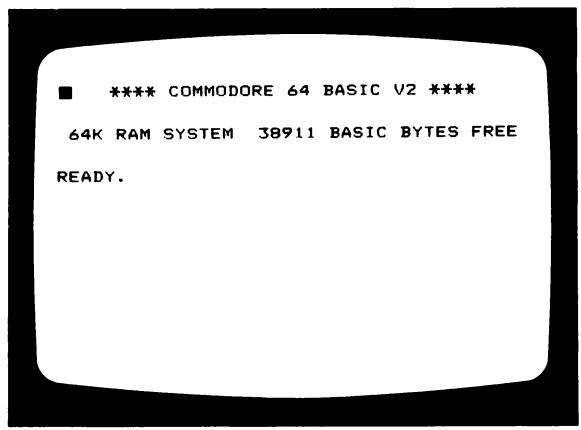
There are three edit keys. Two of the keys have arrow signs on them, pointing up and down on one and left and right on the other. The third edit key is INST/DEL .



All of these keys repeat themselves if you hold them down. Pressing the CRSR UP/DOWN key on its own moves the cursor down; pressing CRSR UP/DOWN and SHIFT together moves the cursor up. Pressing the CRSR LEFT/RIGHT key on its own moves the cursor right; pressing CRSR LEFT/RIGHT and SHIFT together moves the cursor left.



This is what will appear on the screen:



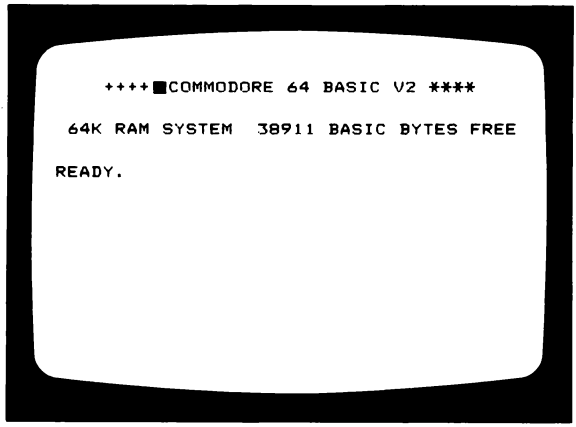
You try

Switch the computer off and then on again, so that the 'signing on' message appears. Now hold down SHIFT and press CRSR UP/DOWN five times.

You try

Press CRSR LEFT/RIGHT (without holding down SHIFT) four times. Now type + + + +

This is what will appear on the screen:



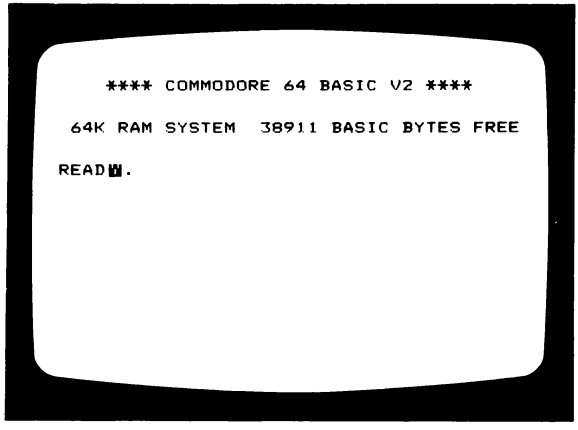
Notice that what you type appears on the screen instead of what was there before. Using the CRSR UP/DOWN , CRSR LEFT/RIGHT and SHIFT keys, move the cursor around and change the screen display in some different ways.

We have already seen how pressing INST/DEL rubs out characters to the left of the cursor. That was the 'DEL' part of 'INST/DEL', because DEL is short for DELETE. INST is short for INSERT, and pressing INST/DEL together with SHIFT allows you to put in letters without printing on top of what is there already.

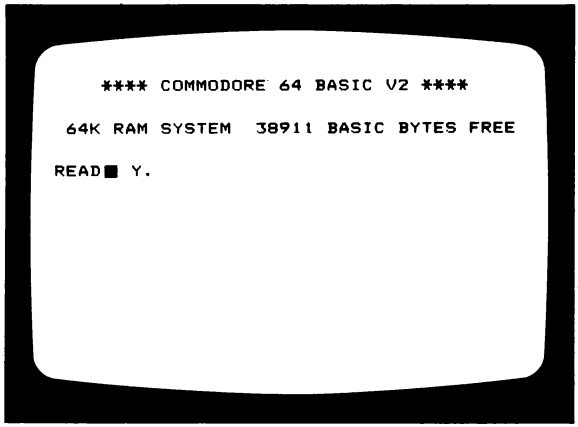
This is what will appear on the screen:

You try

Switch the computer off and then on again to produce 'signing on' message. Hold down SHIFT and press CRSR UP/DOWN once. Release SHIFT and press CRSR LEFT/RIGHT four times.



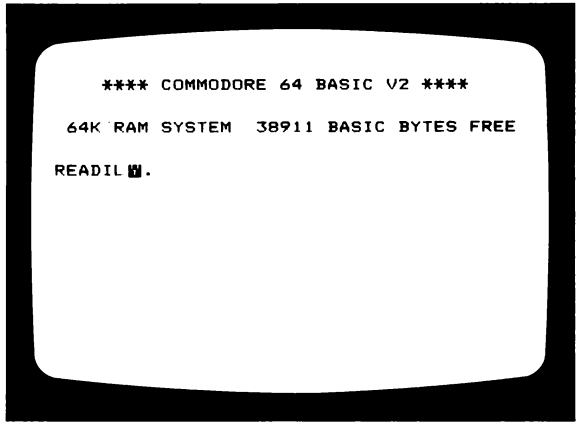
This is what will appear on the screen:



You try

Hold down SHIFT and press INST/DEL twice.

This is what will appear on the screen:



You try

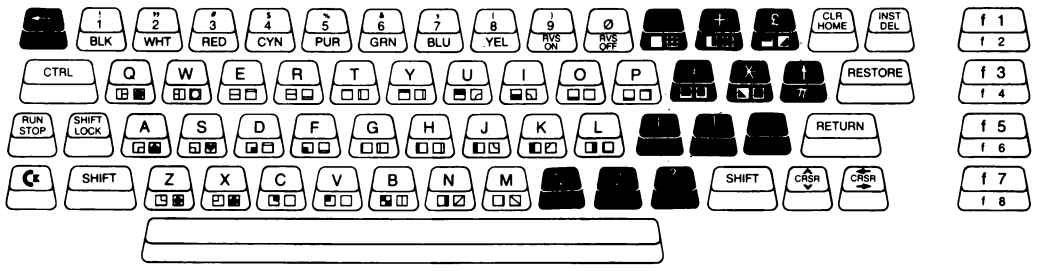
Type I and then L.

You try

Using the two arrow keys together with INST/DEL and SHIFT, alter the screen display in different ways by putting in and taking out words and letters.

Symbol keys

These are the 13 keys which print symbols and are mainly grouped together on the right-hand side of the keyboard. There are also some more symbol keys which are above the number keys and to use these SHIFT must be held down. When there are two symbols on a key, hold down SHIFT for the one on the top.



You try

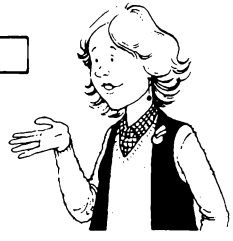
Type out the following. Press RETURN at the end of each line.

1 !
2 "
3 #
4 $
5 %
6 &
7 '
8 (
9)

You try

Type out the following. Press RETURN at the end of each line.

1 PUNCTUATION ! " ' ; : , . ?
2 ARITHMETIC % ( ) + − * / = ↑
3 OTHER # $ & @ £ ← [ ] < >



Find all the keys,
And type with ease.

GETTING STARTED

P. C. Truemo

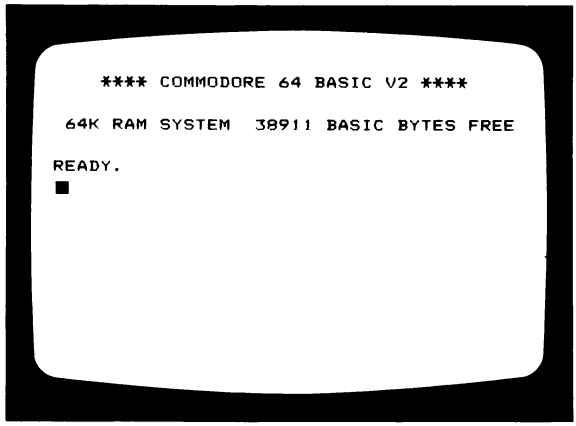
My advice

When you start to make the computer do things for you always check them very carefully. The computer is very exact in the way it understands things. A missing comma, a full stop in the wrong place or a wrong letter will result in the computer not understanding you.
When this happens – INVESTIGATE – track down every possible mistake until you get everything correct. It may take some time at first to track down your mistakes, but gradually you will not only become quicker at doing this, but you will also make fewer mistakes.
Make notes of the things you find useful, so that you can use them again easily whenever you want. Finally I find these points very useful.
1. Remember that the computer needs all its instructions in CAPITAL letters.
2. Remember that after every instruction you must press RETURN .

Bobby Truemo

In this section you will be shown how to use the computer in three different ways.
1 Using the computer as a typewriter.
You will be shown how to PRINT numbers and characters on the screen and how to space them out in various ways.
2 Using the computer as a calculator.
You will be shown how to add, take away, multiply and divide numbers using the computer.
3 Using the computer for storing information.
You will be shown how to put numbers into the memory of the computer.

When switched on the Commodore 64 micro will display the following:



COMMODORE 64 – the name of the computer.

BASIC V2 – the computer language used by the Commodore 64 computer.

64K RAM SYSTEM – this tells you the size of the computer's memory.

38911 BASIC BYTES FREE – this tells you how many units of memory (called 'bytes') you can use in the BASIC language.

READY – the signal that the computer is ready for you to type in instructions.

■ This is the cursor. It shows you where the letters that you type will appear on the screen. The cursor normally moves to the right, but it moves to the left and rubs out what is there if [INST/DEL] is pressed without [SHIFT] .

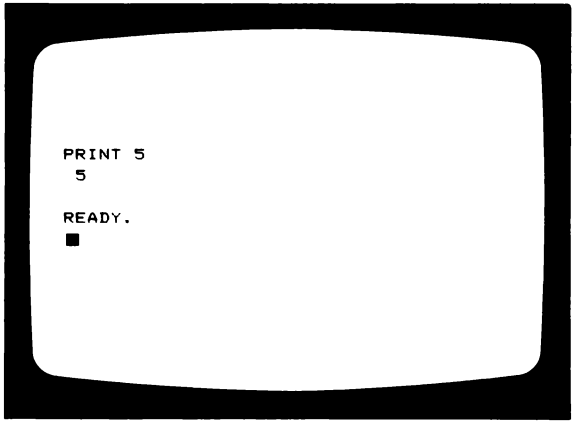
The computer as a typewriter

To make the computer write things we use PRINT. After the statement is complete, you must remember to press [RETURN] . Rather than typing the letters PRINT, you can type ? instead.

Printing numbers

[ ]

This is what will appear on the screen:



You try

Type PRINT 5 then press [RETURN] .

[ ]

Make a note

To print a number type PRINT or ? followed by the number and then press [RETURN] .

At the end of a line, Press RETURN, that's fine.

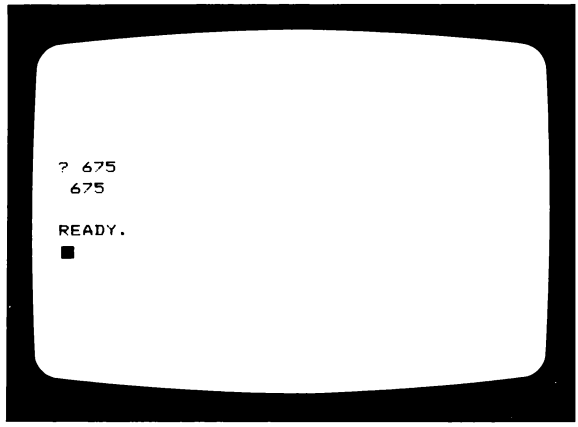
You try

Type ? 675 then press RETURN .

You try

Using the same idea print some numbers of your own.

This is what will appear on the screen:



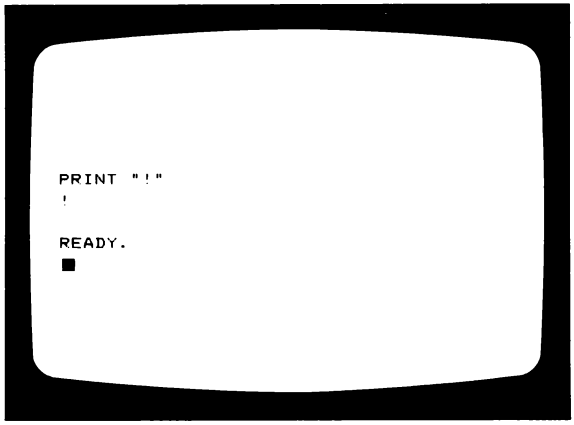
Printing symbols, letters and words

Commodore 64 User Manual 32

You try

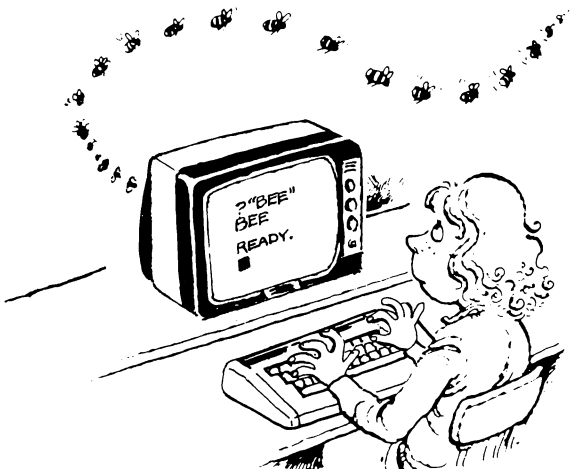
Type PRINT "!" then press RETURN .

This is what will appear on the screen:

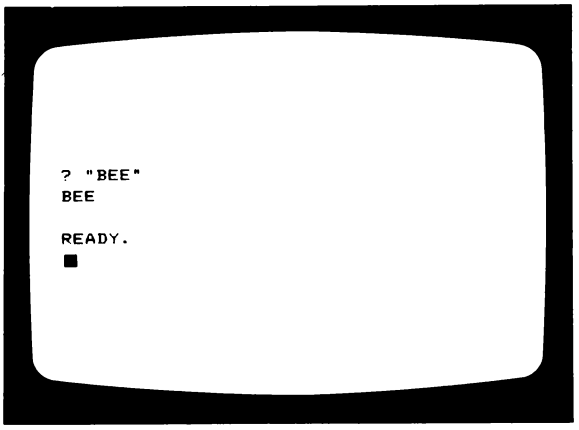


Make a note

To print a symbol, letter or word type PRINT or ? followed by the symbol, letter or word inside speech marks, then press RETURN .



This is what will appear on the screen:



You try

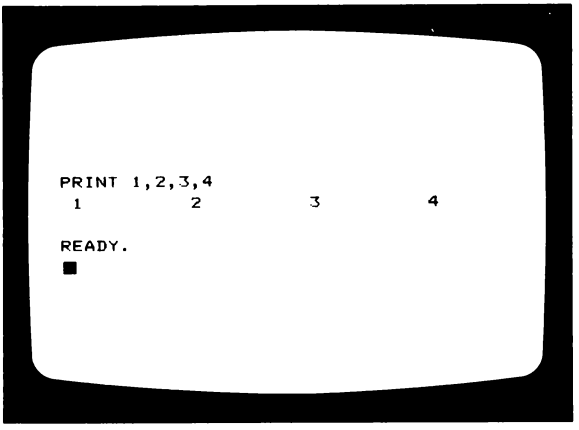
Type ? "BEE"
then press RETURN .

You try

Use PRINT or ? to make the computer write your name.

Spacing numbers

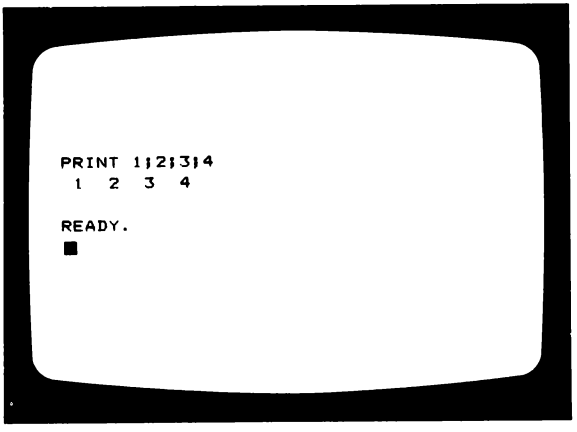
This is what will appear on the screen:



You try

Type PRINT 1,2,3,4
then press RETURN .

This is what will appear on the screen:



You try

Type PRINT 1;2;3;4
then press RETURN .

You try

Type PRINT 12,,,,34 then press RETURN .

This is what will appear on the screen:

```
PRINT 12,,,,34
 12
 34

READY.
■
```

You try

Using the same ideas space out some numbers of your own. Notice that the computer always prints numbers with one space either side.

Don't stand and wait, INVESTIGATE.

Make a note

The symbol
, prints spaced out
; prints numbers with two spaces in between
,,,, prints on two lines

Spacing words

You try

Type PRINT "DOG","CAT" then press RETURN .

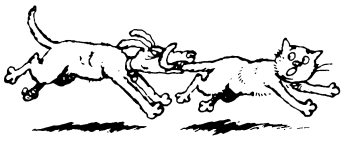
This is what will appear on the screen:

```
PRINT "DOG","CAT"
DOG         CAT

READY.
■
```

You try

Type PRINT "DOG";"CAT" then press RETURN .



This is what will appear on the screen:

```
PRINT "DOG";"CAT"
DOGCAT

READY.
■
```

You try

Type PRINT "DOG",,,,"CAT" then press RETURN .

This is what will appear on the screen:

```
PRINT "DOG",,,,"CAT"
DOG
CAT

READY.
■
```

You try

Using the same idea space out your first and second names.

Make a note

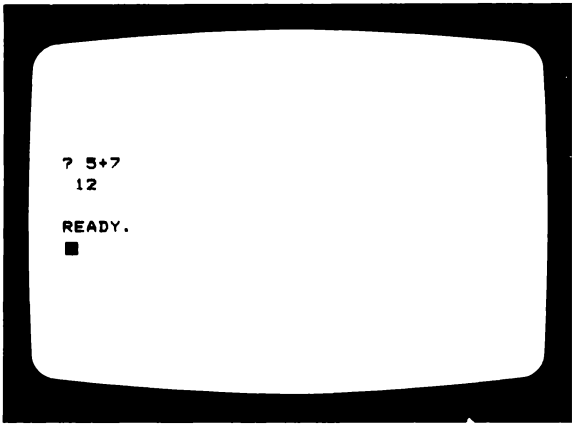
The symbol
, prints words spaced out
; prints words without spaces
,,,, prints words on two lines

The computer as a calculator

The PRINT statement can also be used to make the computer act like a calculator. It can be given a question and it will supply the answer. Remember that ? can be used instead of PRINT. After the statement is complete RETURN should be pressed.

```
                    
```

This is what will appear on the screen:



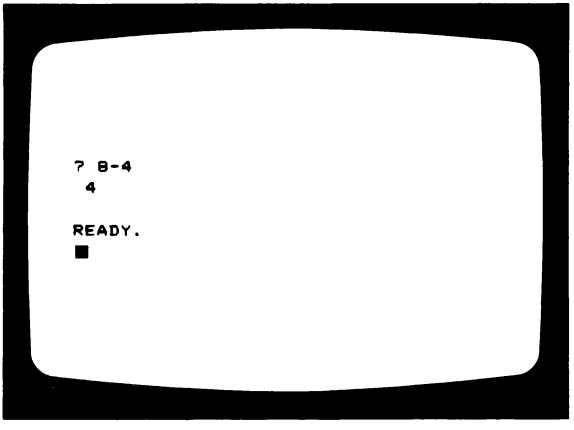
You try

Type ? 5+7
then press RETURN .

```
                    
```

```
                    
```

This is what will appear on the screen:



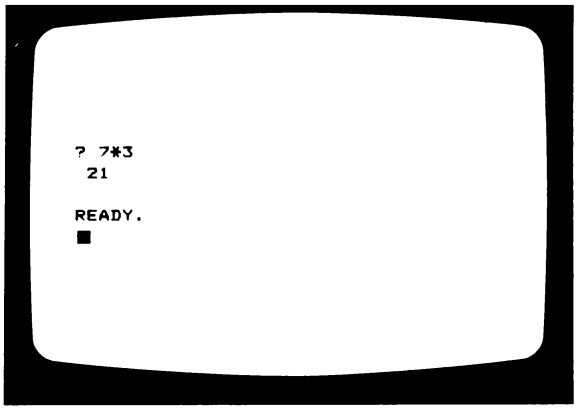
You try

Type ? 8−4
then press RETURN .

```
                    
```

You try

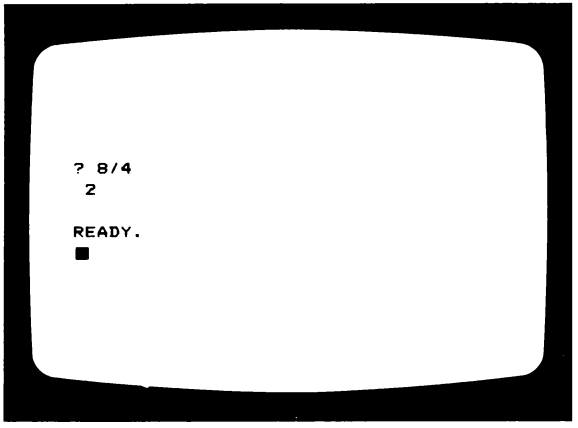
Type ? 7*3 then press RETURN .



This is what will appear on the screen:

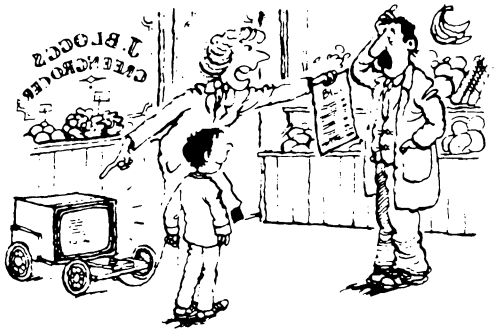
You try

Type ? 8/4 then press RETURN .



You try

1. Using the same idea make up some of your own sums.
2. Check a shopping bill with the computer.



Make a note

The symbols used are

\+ add

\- subtract (or take away)

* multiply (or times)

/ divide (or share)

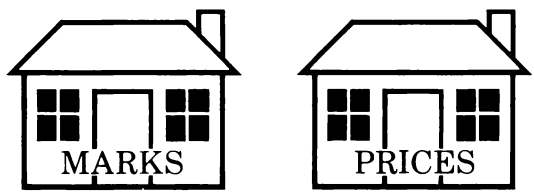
The computer memory

You can also use the computer to remember information. It is important to know just how the computer does this before you go on to the next part of this section.

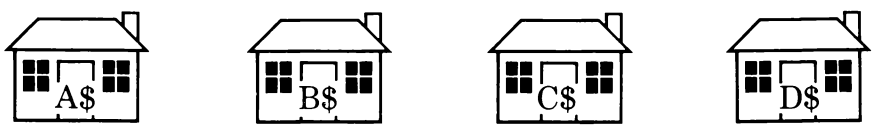
Just as you live at one address, your friend at another, and your teacher at yet another address, so the computer uses a similar system of addresses for storing information.

It stores numbers at the following addresses in its memory. The addresses can either be letters or words. For example

or



It stores words at addresses like these.



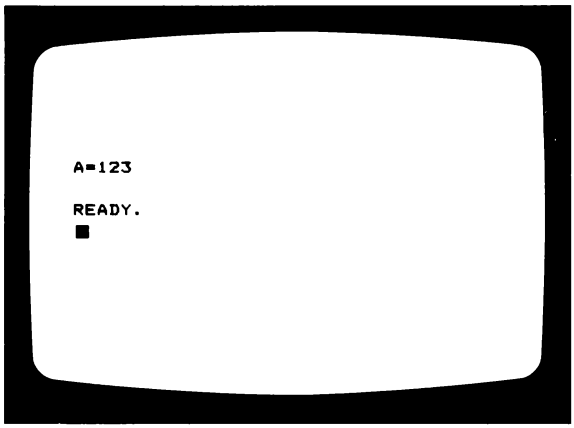
Make a note

The computer uses A, B, C, D etc. for addresses to store numbers and uses A$, B$, C$, D$ etc. for addresses to store words.

Using the computer for storing information

The = sign is used to put numbers or words into the memory of the computer.

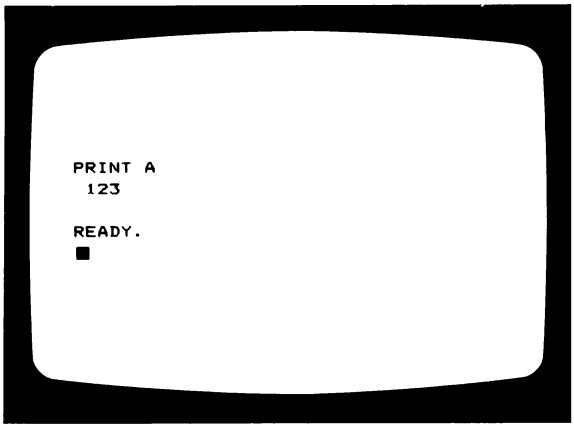
This is what will appear on the screen:



You try

Type A=123
then press RETURN .

This is what will appear on the screen:



You try

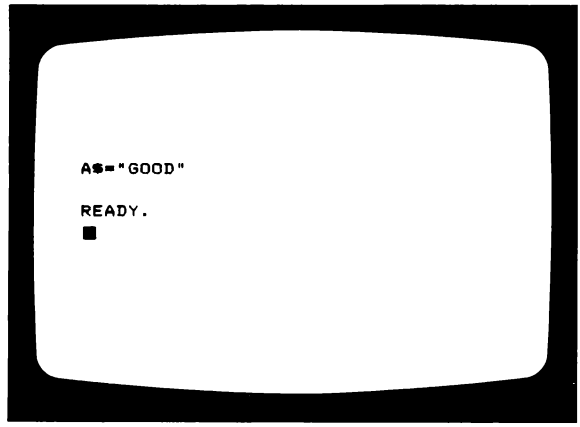
Type PRINT A then press RETURN .

A=123 put the number 123 into the memory of the computer at address A. PRINT A recalled the number which was in the memory of the computer at address A.

You try

Use the = sign to put the number 12 into address B and number 4 into address C. Then type PRINT B+C,B−C,B*C,B/C then press RETURN .

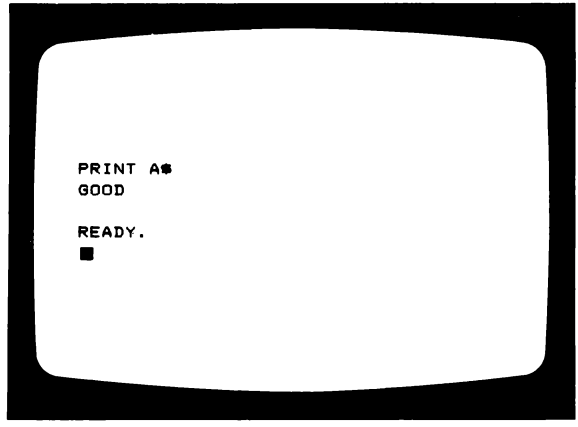
This is what will appear on the screen:



You try

Type
A$="GOOD"
then press RETURN .

This is what will appear on the screen:



You try

Type PRINT A$ then press RETURN .

A$="GOOD" put the word GOOD into the memory of the computer at address A$. PRINT A$ recalled the word which was in the memory of the computer at address A$.

You try

Use the = sign to put "I" in the address B$ and "AM" in the address C$. Type PRINT C$,B$,A$ and press RETURN .
Type PRINT B$,C$,A$ then press RETURN

WRITING PROGRAMS

Prof. O. Crumpet

My advice

You want ideas that work. You will probably need to spend some time thinking about them. If your idea does not work as you thought it would, then try to find out why. It may be that one very small alteration will make it work exactly as you intended.

However, it could be that your idea will never work, no matter how you alter it, so be prepared at times to start all over again. Sorting out ideas is not easy, but it is easier if you have some plan to work to.

Often an idea does not work quite as you wanted, but it could be used in another way. Make a note of it so that you can use it later. You must, of course, keep a very careful record of all your really good ideas.

Oliver Crumpet

The computer can do many things. Here are just a few. Try them for yourself.

Holding down [SHIFT] and pressing [CLR/HOME] gives a blank screen.

PRINT (or ?) followed by [RETURN] gives a blank line.

PRINT "COMPUTER" followed by [RETURN] gives the print out of COMPUTER.

PRINT "PROGRAMMER" followed by [RETURN] gives the print out of PROGRAMMER.

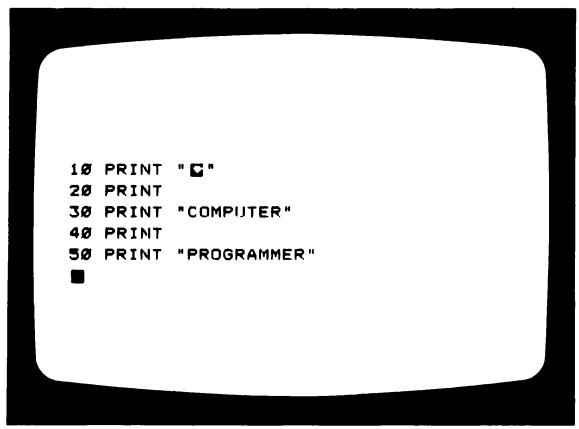
The computer can be given many other things to do, but instead of giving them to the computer one at a time, they can be given together. When the computer is given a list of things to do in order it is called a computer program. The lines begin with line numbers to tell the computer the order in which they should be carried out. Examples of line numbers are: 1Ø 2Ø 3Ø 4Ø 5Ø 6Ø etc.

An example of a short program is

```
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø PRINT
3Ø PRINT "COMPUTER"
4Ø PRINT
5Ø PRINT "PROGRAMMER"
```

You are making a start
At the programming art.

This is what will appear on the screen:



Notice that where you pressed [SHIFT] and [CLR/HOME] a heart appeared on the screen in reverse video.

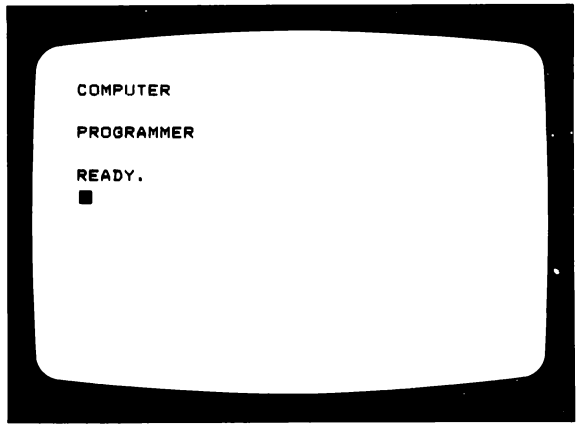
You try

Type the program above, line by line. In the first line press the [SHIFT] and [CLR/HOME] keys together rather than typing the words [SHIFT] and [CLR/HOME] . After you have written a complete line press [RETURN] to start the next line.

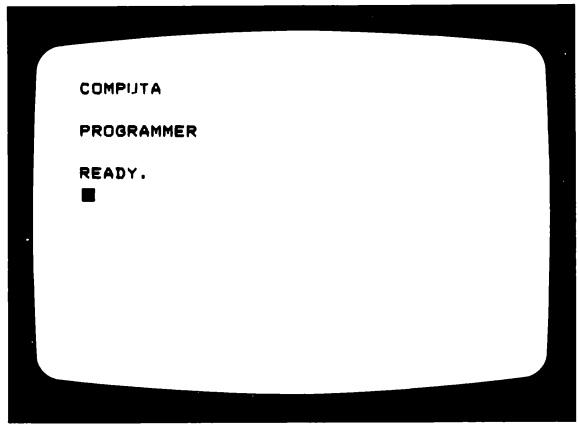
You try

Now type the word RUN and press RETURN .

This is what will appear on the screen:



If you have made any mistakes you can correct them by retyping the line in which they occur. Suppose your screen showed



The mistake is in line number 3Ø, so by retyping the line number 3Ø this mistake can be corrected:
3Ø PRINT "COMPUTER"
(followed by RETURN).
This will correct the line in the program and give you the correct display.
You may also alter lines in the program in the same way.
Type in the following:
3Ø PRINT "A COMPUTER"
(followed by RETURN).
This will alter the line in the program.

While you are writing programs it is quite useful to be able to look back at the program you have written. To do this type the word LIST followed by RETURN

To find things missed, just type LIST.

This is what will appear on the screen:

```
LIST

10 PRINT "■"
20 PRINT
30 PRINT "A COMPUTER"
40 PRINT
50 PRINT "PROGRAMMER"
READY.
■
```

The program line numbers go up in tens so it is possible to add extra lines. For example, try typing
15 PRINT "I AM" (followed by [RETURN]).
This puts the line into the program as follows:
1Ø PRINT " [SHIFT] [CLR/HOME] "
15 PRINT "I AM"
2Ø PRINT
3Ø PRINT "A COMPUTER"
4Ø PRINT
5Ø PRINT "PROGRAMMER"
6Ø PRINT
You can check that this has been done by typing LIST followed by [RETURN] . Type RUN followed by [RETURN] .

This is what will appear on the screen:

```
I AM

A COMPUTER

PROGRAMMER

READY.
■
```

Make a note

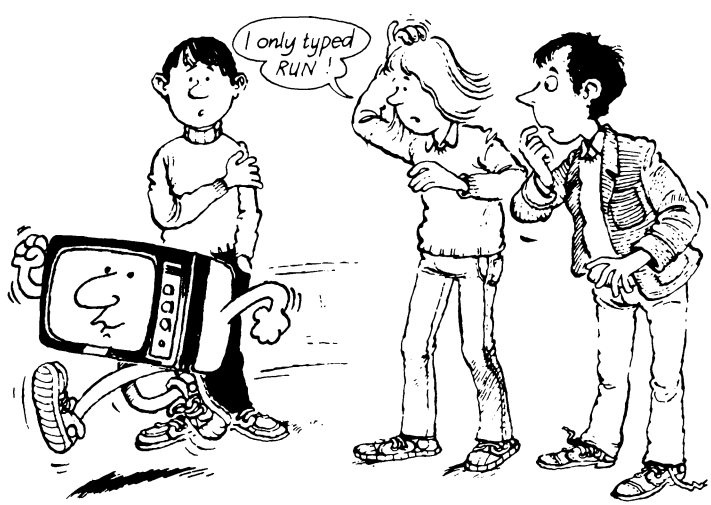
1. Number each line 1Ø, 2Ø, 3Ø etc.
2. Press [RETURN] when the line is complete.
3. Type RUN and press [RETURN] when the program is complete.
4. Correct mistakes by retyping the line.
5. Type LIST and press [RETURN] to look at the program.

Writing programs

In order to write your own programs you need to find out just what the computer can do. The exercises which follow should help you.
Before you start the next section you need to have understood the last section. If you are at all unsure about anything, work through it again.
Type in these programs just as the earlier program was typed in, that is one line at a time followed by RETURN to get to the start of the next line. Check your program when it has all been typed in and correct any mistakes by rewriting the faulty line. When you are sure that the program is complete and correct type RUN and press RETURN and see what happens.
So that the computer does not get one program confused with the next, type NEW and press RETURN before typing in a new program. The computer then forgets the old program.
When you have copied and run the programs in each section then try the exercises. If you come up with any ideas for similar programs, try them out and see how they work.

At the end of this chapter and the next one, there are some projects which are ideas for longer and more interesting programs. For most of the projects, you are helped with writing the basic program, and then you are given some ideas to improve it.

To see what is done,
Just type RUN.



Programs using PRINT

PRINT (?) is used to print out lists, information, diagrams and instructions.

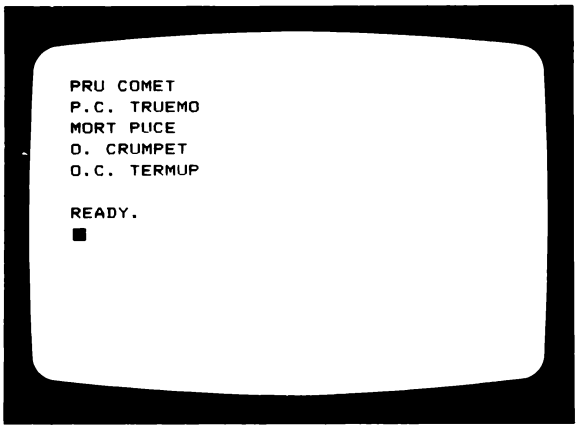
You try

Type NEW then press [RETURN] .
Type in the following program.

```
10 PRINT " [SHIFT] [CLR/HOME] "
20 PRINT "PRU COMET"
30 PRINT "P.C. TRUEMO"
40 PRINT "MORT PUCE"
50 PRINT "O. CRUMPET"
60 PRINT "O.C. TERMUP"
```

Type RUN then press [RETURN] .

This is what will appear on the screen:



When writing a program, you Must remember to start with NEW.

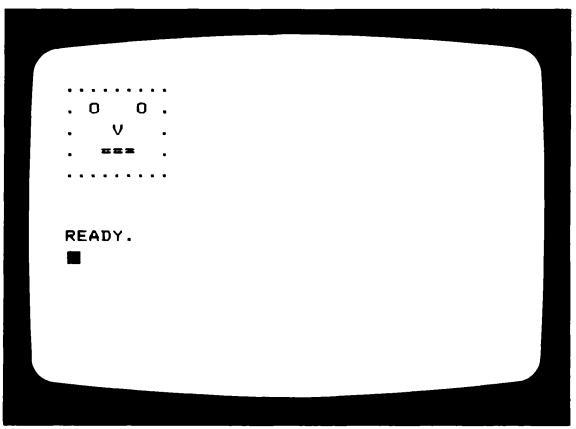
You try

Type NEW then press [RETURN] .
Type in the following program.

```
10 PRINT " [SHIFT] [CLR/HOME] "
20 PRINT " . . . . . ."
30 PRINT " .  O   O  ."
40 PRINT " .   V     ."
50 PRINT " .  ===    ."
60 PRINT " . . . . . ."
```

Type RUN then press [RETURN] .

This is what will appear on the screen:



You try

Make up your own programs using PRINT to do the following:

1. Write out a shopping list.
2. Design a motorway sign.
3. Draw a space shuttle.

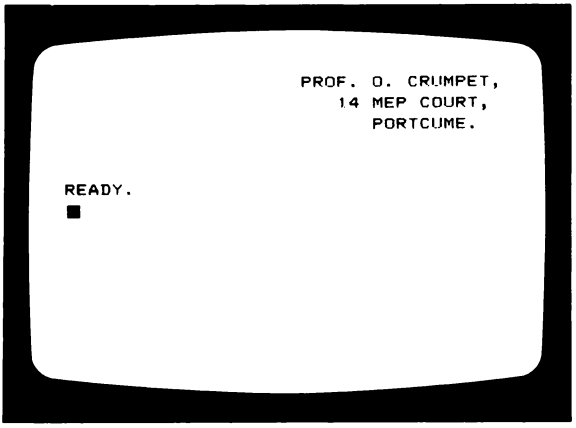
Programs using TAB

TAB is used to set out information or diagrams.

You try

Type NEW then press RETURN .
Type in the following program.
10 PRINT " SHIFT CLR/HOME "
20 PRINT TAB (20) "PROF. O. CRUMPET,"
30 PRINT TAB(23) "14 MEP COURT,"
40 PRINT TAB(26) "PORTCUME."
Type RUN then press RETURN .

This is what will appear on the screen:

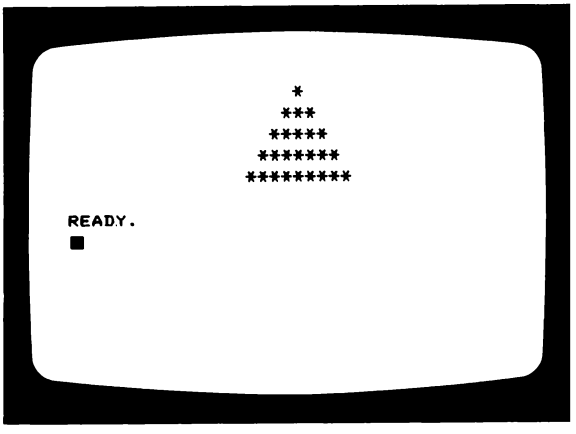


Make sure that you do not put a space between TAB and the first bracket, or the program will not work.

You try

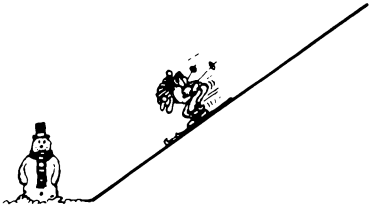
Type NEW then press RETURN .
Type in the following program.
10 PRINT " SHIFT CLR/HOME "
20 PRINT TAB(19)"*"
30 PRINT TAB(18)"***"
40 PRINT TAB(17)"*****"
50 PRINT TAB(16)"*******"
60 PRINT TAB(15)"*********"
Type RUN then press RETURN .

This is what will appear on the screen:



You try

Make up your own programs using TAB to do the following:
1. Draw a staircase.
2. Set out your address.
3. Draw a sloping line.



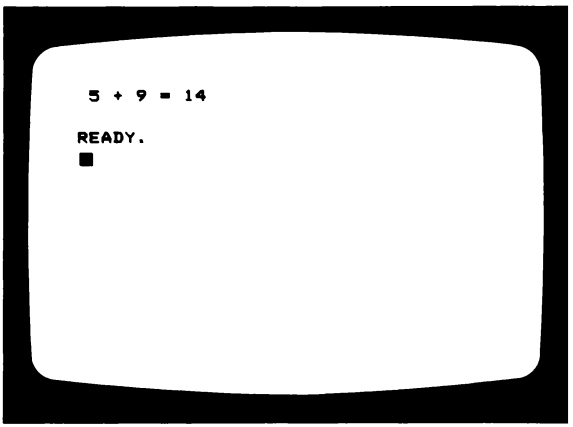
Programs using =

The = sign is used to put a number into an address.

You try

Type NEW then press RETURN .
Type in the following program.
10 PRINT " SHIFT CLR/HOME "
20 A=5
30 B=9
40 S=A+B
50 PRINT A;"+";B;"=";S
Type RUN then press RETURN .

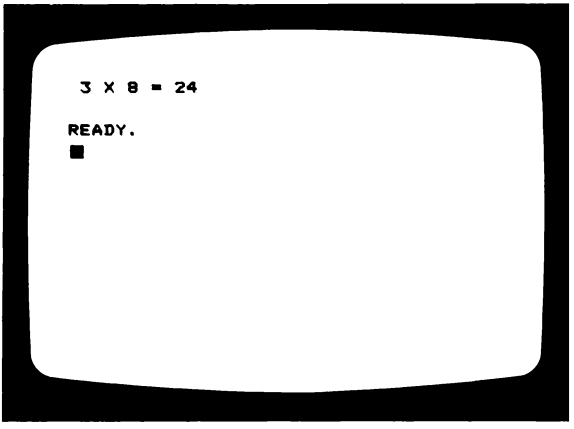
This is what will appear on the screen:



You try

Type NEW then press RETURN .
Type in the following program.
10 PRINT " SHIFT CLR/HOME "
20 E=3
30 F=8
40 P=E*F
50 PRINT E;"X";F;"=";P
Type RUN then press RETURN .

This is what will appear on the screen:



You try

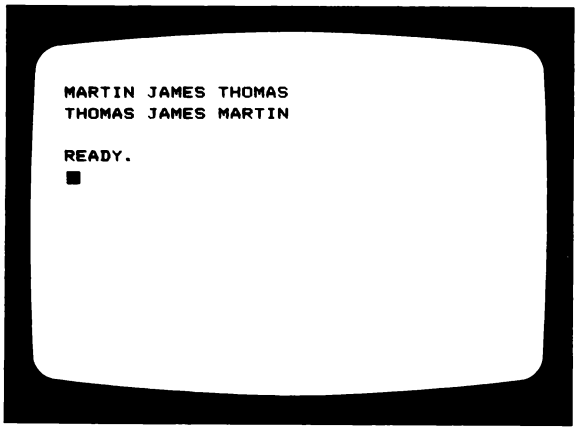
Make up your own programs using = to do the following.
1. Add two other numbers.
2. Multiply two other numbers.
3. Subtract or divide numbers.

The = sign is also used to put words in addresses.

You try

Type NEW then press RETURN .
Type in the following program.
1Ø PRINT " SHIFT CLR/HOME "
2Ø A$="MARTIN"
3Ø B$="JAMES"
4Ø C$="THOMAS"
5Ø PRINT A$" "B$" "C$
6Ø PRINT C$" "B$" "A$
Type RUN then press RETURN .

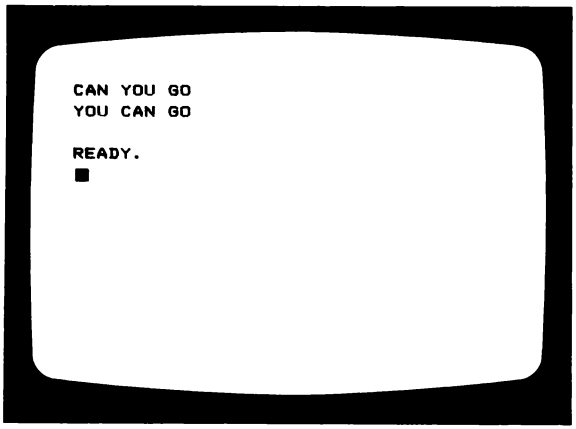
This is what will appear on the screen:



You try

Type NEW then press RETURN .
Type in the following program.
1Ø PRINT " SHIFT CLR/HOME "
2Ø A$="CAN"
3Ø B$="YOU"
4Ø C$="GO"
5Ø PRINT A$" "B$" "C$
6Ø PRINT B$" "A$" "C$
Type RUN then press RETURN .

This is what will appear on the screen:



You try

Make up your own programs using = to do the following.
1. Make A$="IN", B$="ON", C$="SET", D$="SIDE", E$="TO", F$="UP", G$="WARDS". By putting these words together, see how many longer words you can print.
2. Make sentences from words.
3. Select items from a menu.

Programs using INPUT

INPUT is used to put numbers into programs.

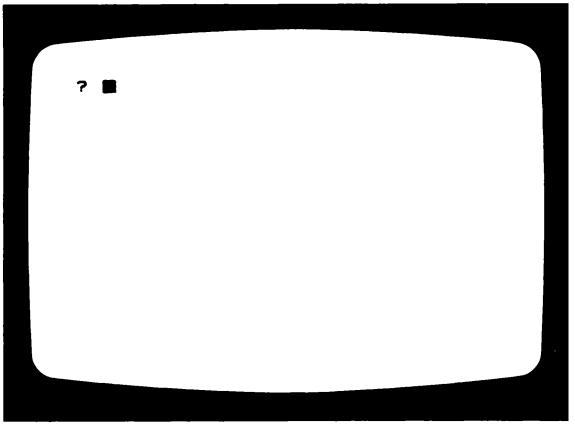
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 INPUT A
30 INPUT B
40 D=A-B
50 PRINT A;"-";B;"=";D
```

Type RUN then press RETURN .

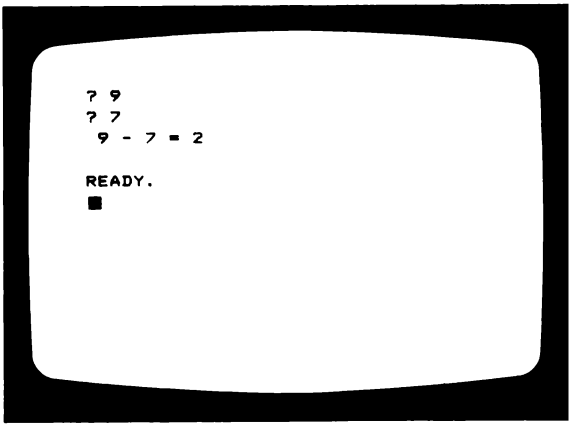
This is what will appear on the screen:



You try

Type in a number, for example 9, then press RETURN .
Type in a number, for example 7, then press RETURN .

This is what will appear on the screen:



You try

Type RUN and press RETURN again. Now put your own numbers into the program.

If remembering numbers is a bore, INPUT can be used to store.

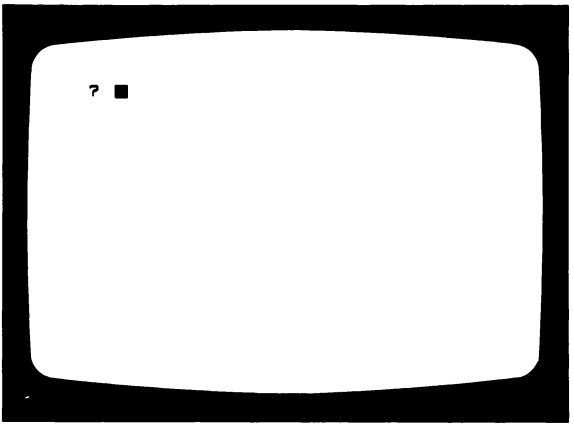
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 INPUT N
30 INPUT D
40 Q=N/D
50 PRINT N;"/";D;"=";Q
```

Type RUN then press RETURN .

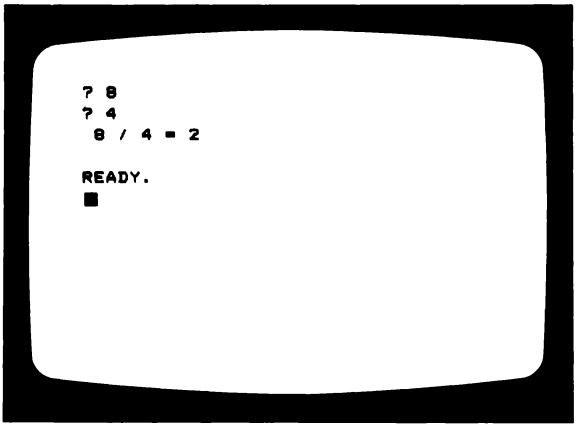
This is what will appear on the screen:



You try

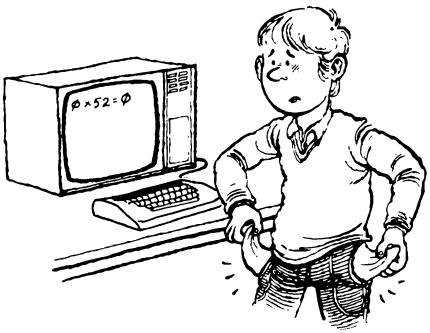
Type in a number, for example 8, then press RETURN .
Type in a number, for example 4, then press RETURN .

This is what will appear on the screen:



You try

Type RUN then press RETURN
Now put your own numbers into the program.



You try

Make up your own programs using INPUT to do the following.

1. Add two numbers together.
2. Multiply two numbers.
3. Ask for the amount of pocket money you receive in a week and multiply this by 52 to give the amount you receive in a year.

INPUT is also used to put words into programs.

You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 PRINT "ANN,EVE,KATE,MARY"
30 PRINT "FIND THE
PALINDROME."
40 INPUT A$
50 PRINT "THE PALINDROME IS
EVE."
```

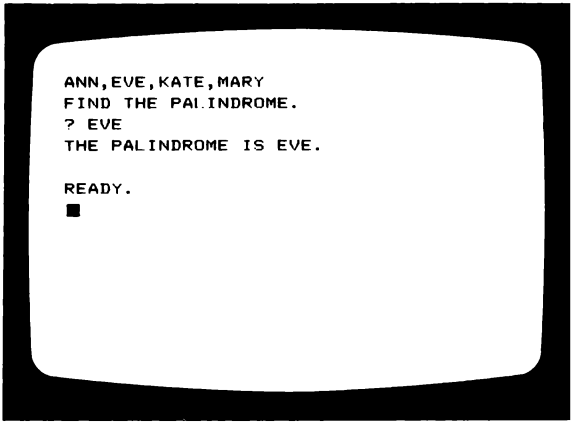
Type RUN then press RETURN .

This is what will appear on the screen:

You try

Type in the palindrome EVE then press RETURN .

This is what will appear on the screen:



(A palindrome is a word which is spelt the same way backwards as forwards.)

You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT "  SHIFT CLR/HOME  "
20 PRINT "GIVE ME A WORD TO
FIT THIS SENTENCE."
30 PRINT "TODAY THE WEATHER
IS  "
40 INPUT A$
50 PRINT "TODAY THE WEATHER
IS  ";A$
```

This is what will appear on the screen:

```
GIVE ME A WORD TO FIT THIS SENTENCE.
TODAY THE WEATHER IS
? ■
```

When a line is too long to fit on the screen just keep typing normally and the computer will sort it out. Do not press RETURN until you have reached the end of the instruction. Type RUN then press RETURN .

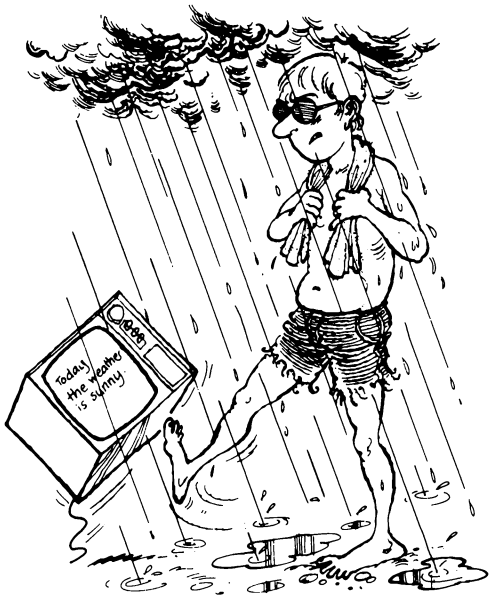
You try

Type in SUNNY (or another word to describe the weather) then press RETURN .

This is what will appear on the screen:

```
GIVE ME A WORD TO FIT THIS SENTENCE.
TODAY THE WEATHER IS
? SUNNY
TODAY THE WEATHER IS SUNNY

READY.
■
```



You try

Make up your own programs using INPUT to do the following.

1. Find a missing word.
2. Answer a simple question.
3. Make the computer have a conversation with you.

Programs using FOR/TO/STEP/NEXT

FOR/TO/STEP/NEXT is used to repeat the same lines in turn for a set of numbers.

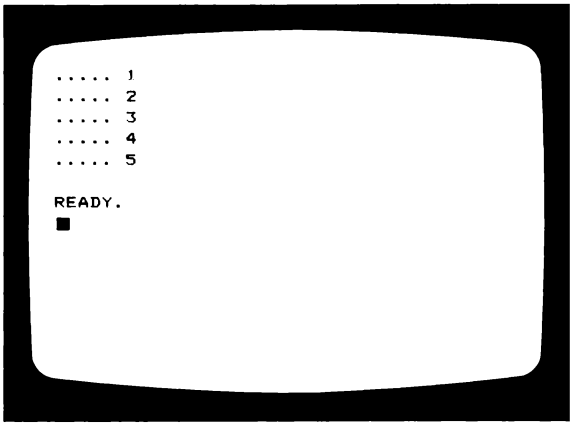
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 FOR C=1 TO 5 STEP 1
30 PRINT ". . . . . ";C
40 NEXT C
```

Type RUN then press RETURN .

This is what will appear on the screen:



For each value of C between 1 and 5, the row of dots and value of C was printed.

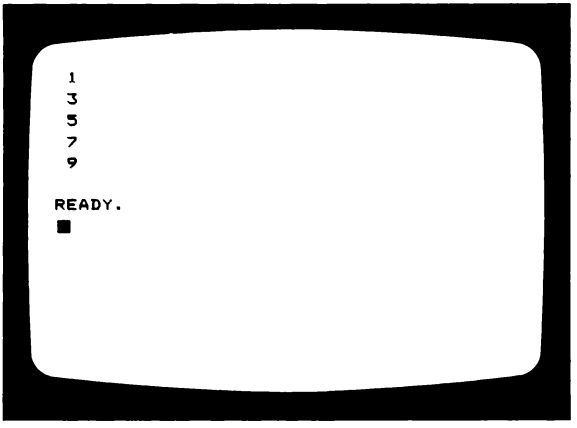
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 FOR N=1 TO 9 STEP 2
30 PRINT N
40 NEXT N
```

Type RUN then press RETURN .

This is what will appear on the screen:



You try

Make up your own programs using FOR/TO/STEP/NEXT to do the following.

1. Print out the numbers 1 to 20.
2. Print out your name and address three times.
3. Print out the three times table.

If you do not tell the computer the size of the step it assumes 'STEP 1', so we could have left this out in the first example.

Programs using DATA/READ

DATA/READ is used to put data (information) into the computer and read (recall) it when it is needed.

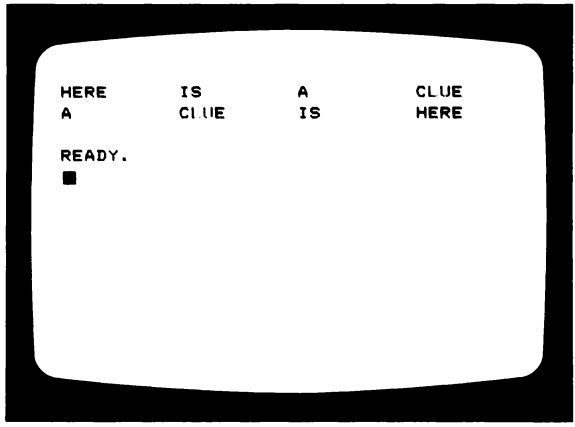
You try

Type NEW then press [RETURN] .
Type in the following program.

```
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø DATA IS,CLUE,A,HERE
3Ø READ W$,X$,Y$,Z$
4Ø PRINT Z$,W$,Y$,X$
5Ø PRINT Y$,X$,W$,Z$
```

Type RUN then press [RETURN] .

This is what will appear on the screen:



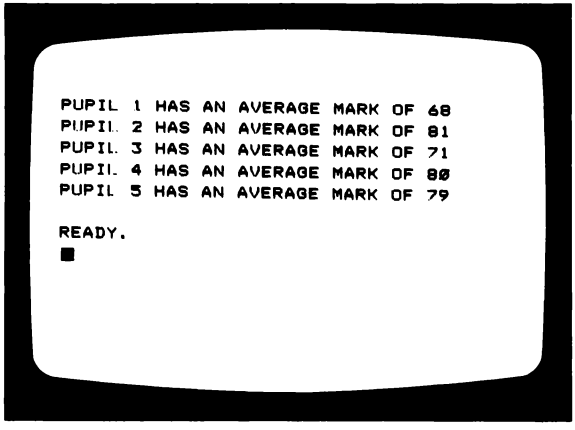
You try

Type NEW then press [RETURN] .
Type in the following program which, for each of five pupils (P), reads in their maths mark (M) and English mark (E) and works out the average (A).

```
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø DATA 75,61,83,79,
64,78,93,67,86,72
3Ø FOR P=1 TO 5
4Ø READ M,E
5Ø A=(M+E)/2
6Ø PRINT "PUPIL";P;"HAS AN
AVERAGE MARK OF";A
7Ø NEXT P
```

Type RUN then press [RETURN] .

This is what will appear on the screen:



You try

Make up your own programs using READ/DATA to do the following:

1. Rearrange a list of items.
2. Make a list of months giving the number of days in each month.

Programs using LEFT$, MID$ and RIGHT$

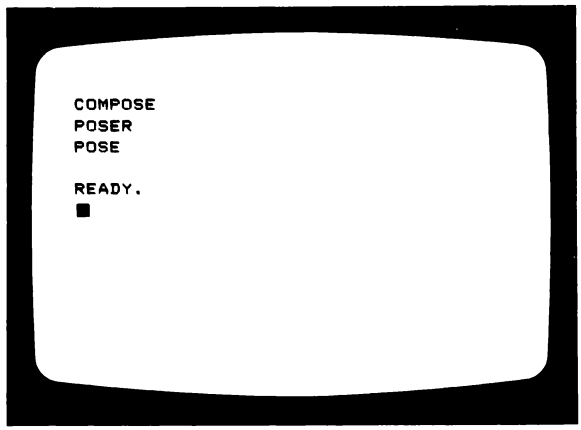
Commodore 64 User Manual 128

The word functions LEFT$, MID$ and RIGHT$ can be used to pick out parts of words.

You try

Type NEW then press [RETURN] .
Type in the following program.
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø W$="COMPOSER"
3Ø PRINT LEFT$(W$,7)
4Ø PRINT RIGHT$(W$,5)
5Ø PRINT MID$(W$,4,4)
Type RUN then press [RETURN] .
Use LEFT$, MID$ and RIGHT$ to print as many words as you can if W$="ANOTHER".

This is what will appear on the screen:



Programs using INT and RND

Commodore 64 User Manual 48

INT gives the whole number part of a number. For example, INT(2.4)=2. RND(1) chooses a random number between Ø and 1.

You try

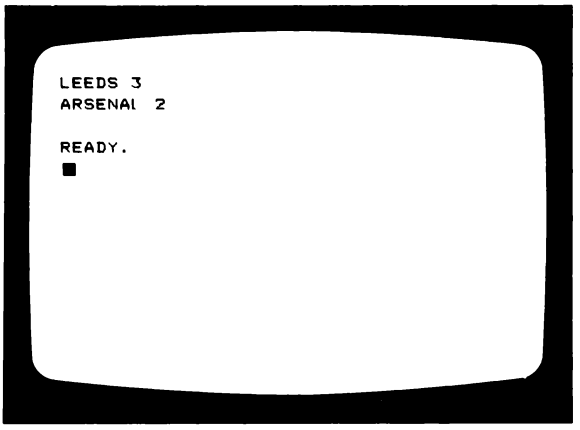
Type NEW then press [RETURN] .
Type in the following program.
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø FOR N=1 TO 2Ø
3Ø PRINT N/4, INT(N/4)
4Ø NEXT N
Type RUN then press [RETURN] .

This program shows how INT works. We need to use INT with RND(1) when we want the answers to be whole numbers. RND(1) chooses a number between Ø and 1 (but not equal to 1), so RND(1)*4 chooses a number between Ø and 4 (but not equal to 4). INT(RND(1)*4) chooses one of the numbers Ø, 1, 2 or 3.

You try

Type NEW then press [RETURN] .
Type in the following program.
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø L=INT(RND(1)*4)
3Ø A=INT(RND(1)*4)
4Ø PRINT "LEEDS";L
5Ø PRINT "ARSENAL";A
Type RUN then press [RETURN] .

This is what may appear on the screen:

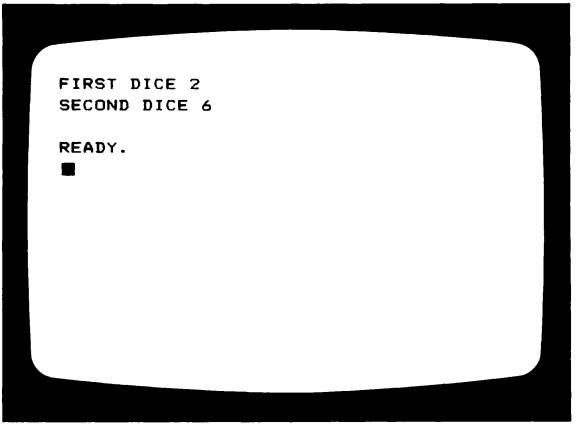


Since the computer chooses numbers at random it is unlikely that the numbers in this display or the next one will be the same as yours.

You try

Type NEW then press [RETURN] .
Type in the following program.
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø F=1+INT(RND(1)*6)
3Ø S=1+INT(RND(1)*6)
4Ø PRINT "FIRST DICE";F
5Ø PRINT "SECOND DICE";S
Type RUN then press [RETURN] .

This is what may appear on the screen:



The instruction F=1+INT(RND(1)*6) chooses one of the numbers 1, 2, 3, 4, 5 or 6 at random and puts the chosen number into the address F. Try running the program again and different numbers will probably be chosen.

You try

Make up your own programs using INT and RND(1) to do the following:
1. Choose a winning raffle ticket from 1ØØØ tickets.
2. Select numbers for a game of bingo (numbers 1 to 9Ø).
3. Make up sums and print the correct answer when you have had a go.

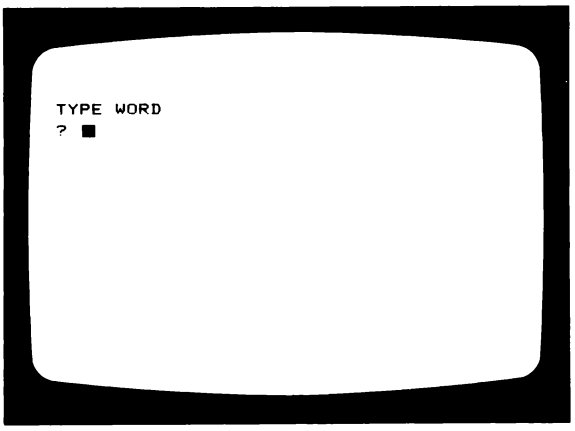
Programs using IF/THEN

IF/THEN is used to introduce alternatives.

You try

Type NEW then press [RETURN] .
Type in the following program.
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø PRINT "TYPE WORD"
3Ø INPUT W$
4Ø IF W$="WORD" THEN PRINT "CORRECT"
5Ø IF W$<>"WORD" THEN PRINT "WRONG"
Type RUN then press [RETURN] .
<> means 'is not equal to'.

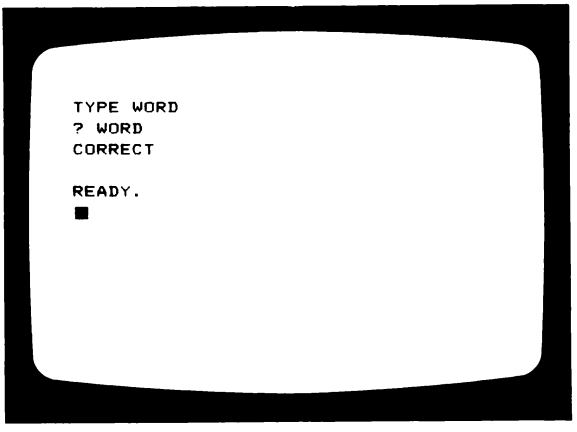
This is what will appear on the screen:



You try

Type WORD then press [RETURN] .

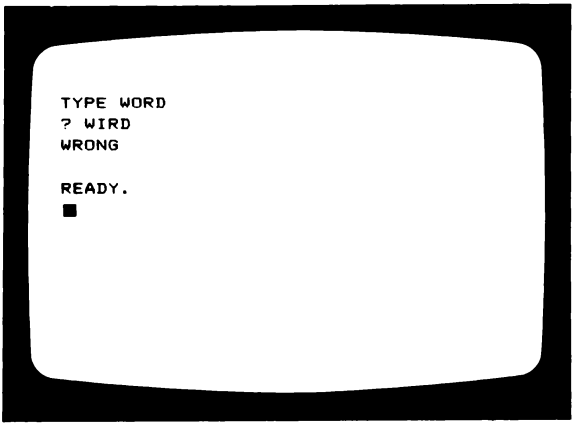
This is what will appear on the screen:



You try

Run the program again.
Type WIRD then press [RETURN] .

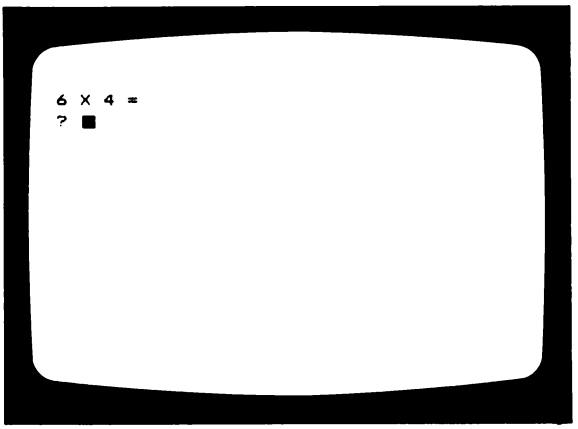
This is what will appear on the screen:



You try

Type NEW then press RETURN .
Type in the following program.
10 PRINT " SHIFT CLR/HOME "
20 PRINT "6 × 4 ="
30 INPUT A
40 IF A=24 THEN PRINT
"CORRECT"
50 IF A<>24 THEN PRINT
"WRONG"
Type RUN then press RETURN .

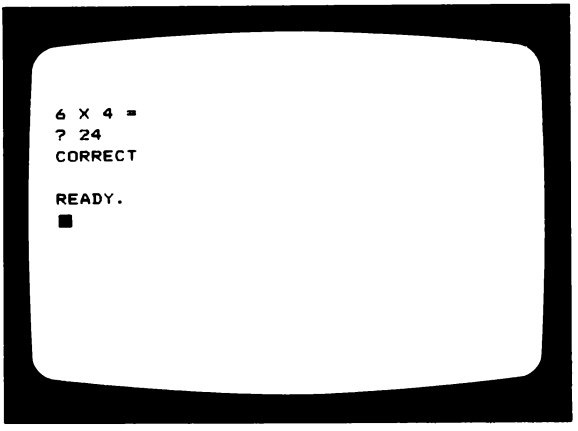
This is what will appear on the screen:



You try

Type 24 then press RETURN .

This is what will appear on the screen:

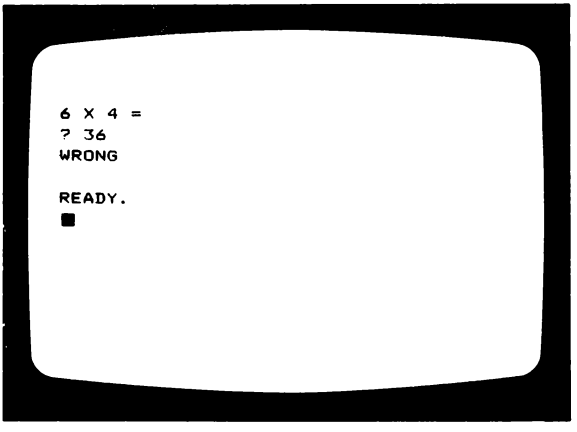


You try

Run the program again.
Type 36 then press RETURN .

Make up your own programs using IF/THEN to do the following:
1. Check an answer to a simple question.
2. Make the computer ask if you are well and give a suitable reply to your answer.

This is what will appear on the screen:



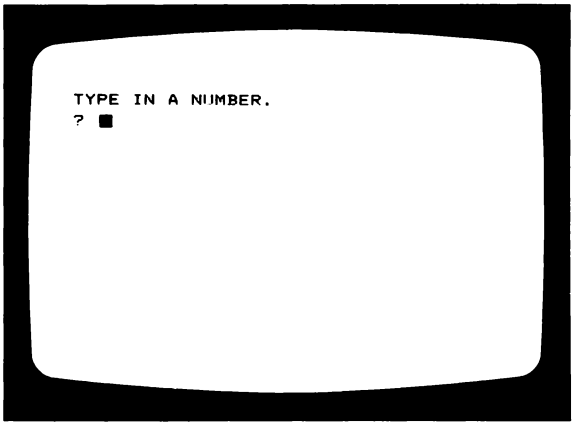
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT "  SHIFT CLR/HOME  "
20 PRINT "TYPE IN A NUMBER."
30 INPUT N
40 IF N<100 THEN GOTO 70
50 PRINT N "IS MORE THAN 99."
60 END
70 PRINT N "IS LESS THAN 100."
```

Type RUN then press RETURN .

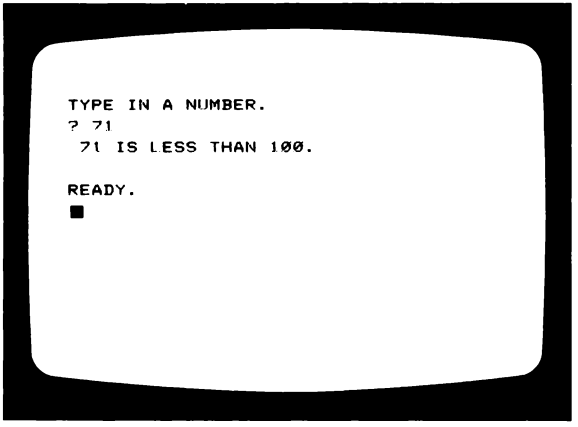
This is what will appear on the screen:



You try

Type in a number, for example 71, then press RETURN .

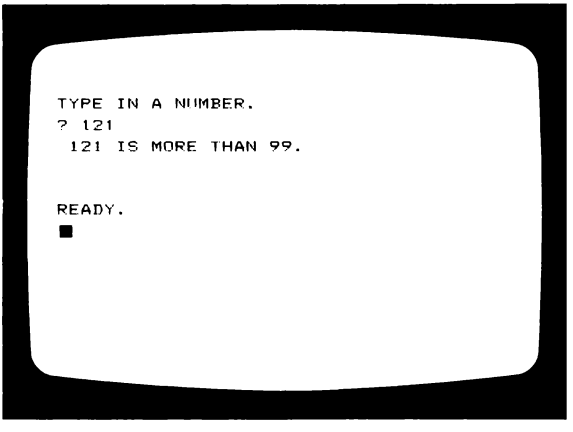
This is what will appear on the screen:



You try

Run the program again. Type in the number 121, the press RETURN .

This is what will appear on the screen:



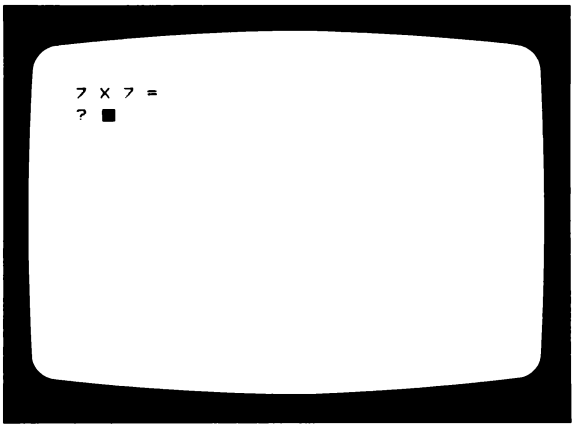
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 PRINT "7 × 7 = "
30 INPUT A
40 IF A=49 THEN GOTO 70
50 PRINT "WRONG"
60 GOTO 20
70 PRINT "CORRECT"
```

Type RUN then press RETURN .

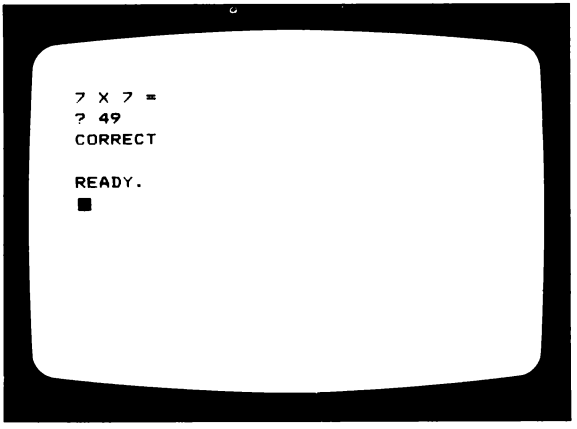
This is what will appear on the screen:



You try

Type in your answer then press RETURN .

This is what will appear on the screen:



You try

Make up your own programs using GOTO to do the following.

1. PRINT YES or NO in answer to a question.
2. Print a word in English or French.
3. Input the cost of five items of shopping and print the total cost. You will need to make the computer keep count of the number of items input so far.

Programs using GOSUB/RETURN

A subroutine is a useful way of repeating some lines of a program. GOSUB is rather like GOTO and is used to jump to the line named after it. RETURN is used to jump back to the program line after the line saying GOSUB. The instruction RETURN must be typed out: it is not the same as pressing the RETURN key.

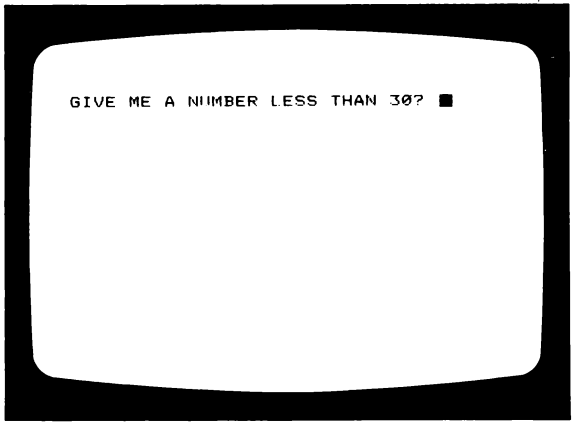
You try

Type NEW then press RETURN .
Type in the following program.
If you prefer, you can use the word 'DIVIDED' instead of the word 'SHARED'.

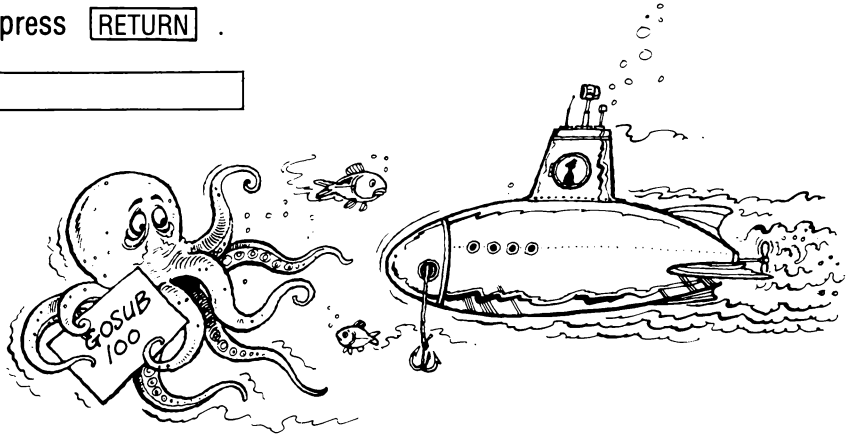
```
1Ø PRINT " SHIFT CLR/HOME "
2Ø INPUT "GIVE ME A NUMBER
LESS THAN 3Ø";N
3Ø M=2:GOSUB 1ØØ
4Ø M=3:GOSUB 1ØØ
5Ø M=5:GOSUB 1ØØ
6Ø END
1ØØ REM SUBROUTINE
11Ø PRINT "THE REMAINDER
WHEN";
12Ø PRINT N;"IS SHARED "
13Ø PRINT "BY";M;"IS";
N-M*INT(N/M)
14Ø RETURN
```

Type RUN then press RETURN .

This is what will appear on the screen:



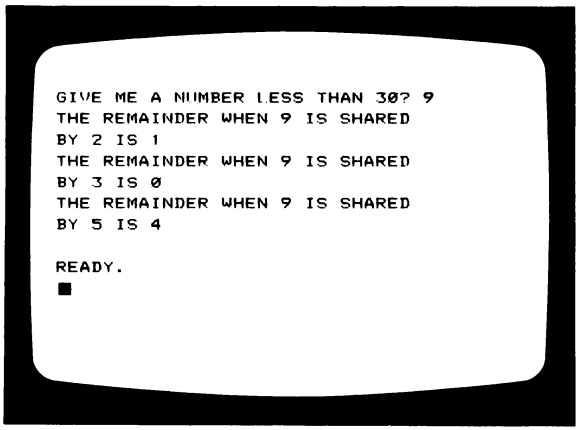
Notice how two instructions can be given on one line if they are separated by a colon. And notice the word 'REM' in line 1ØØ. REM is short for REMARK. REM lines are used to explain what the program is doing, but the computer ignores any writing after REM.



You try

Type in a number, for example 9, then press RETURN .

This is what will appear on the screen:



This program can be turned into a game by leaving unchanged lines 3Ø, 4Ø, 5Ø, 1ØØ, 11Ø, 13Ø and 14Ø and typing in the following lines:

```
2Ø N=INT(3Ø*RND(1))
6Ø INPUT "WHAT IS MY NUMBER";D
7Ø PRINT "MY NUMBER IS";N
8Ø END
12Ø PRINT " MY NUMBER IS SHARED "
```

Take care to put spaces in the right places, otherwise the screen will look very messy. Type RUN then press RETURN . Try to guess the number.

You may like to change the program so that it checks whether you have guessed the right number and prints 'RIGHT' or 'WRONG'.

You try

Type NEW then press RETURN .
Type in the following program.

```
9Ø PRINT " SHIFT CLR/HOME "
1ØØ J$="WENT TO MOW "
11Ø K$=J$+"A MEADOW "
12Ø I$="ONE MAN "
13Ø GOSUB 1ØØØ
14Ø GOSUB 2ØØØ
15Ø PRINT
16Ø I$="TWO MEN "
17Ø GOSUB 1ØØØØ
18Ø I$=I$+", ONE MAN "
19Ø GOSUB 2ØØØ
2ØØ END
1ØØØ REM FIRST LINE SUBROUTINE
1Ø1Ø PRINT I$+J$+","
1Ø2Ø PRINT K$
1Ø3Ø RETURN
2ØØØ REM SECOND LINE
SUBROUTINE
2Ø1Ø PRINT I$+"AND HIS DOG "
2Ø2Ø PRINT K$
2Ø3Ø RETURN
```

Type RUN then press RETURN .
Add a few lines to the program so that it will print out a third verse.

AND is used to check two conditions.

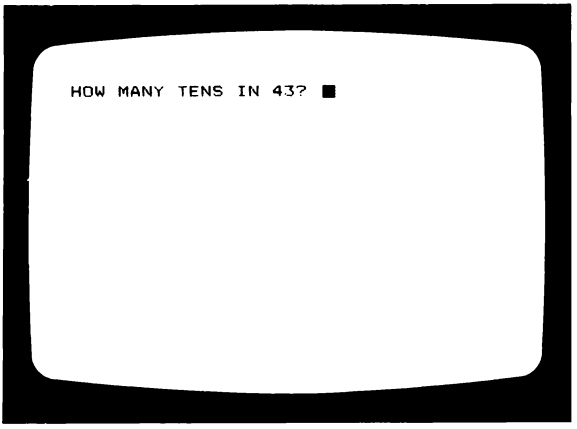
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT " SHIFT CLR/HOME "
20 PRINT "HOW MANY TENS IN
43";
30 INPUT T
40 PRINT "HOW MANY UNITS IN
43";
50 INPUT U
60 IF T=4 AND U=3 GOTO 80
70 GOTO 20
80 PRINT "BOTH CORRECT"
```

Type RUN then press RETURN .

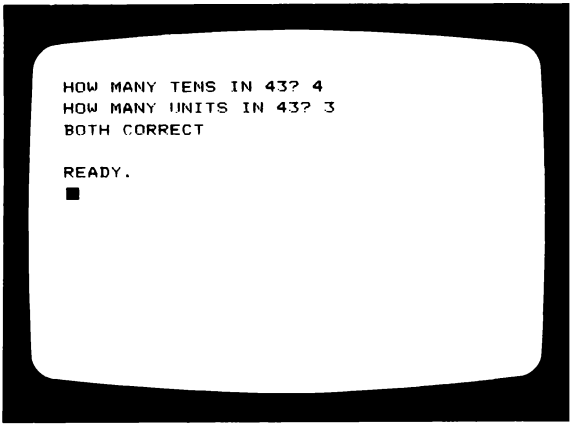
This is what will appear on the screen:



You try

Type 4 then press RETURN .
Type 3 then press RETURN .

This is what will appear on the screen:



You try

Run the program again.
Type in one or two incorrect answers and notice what happens.

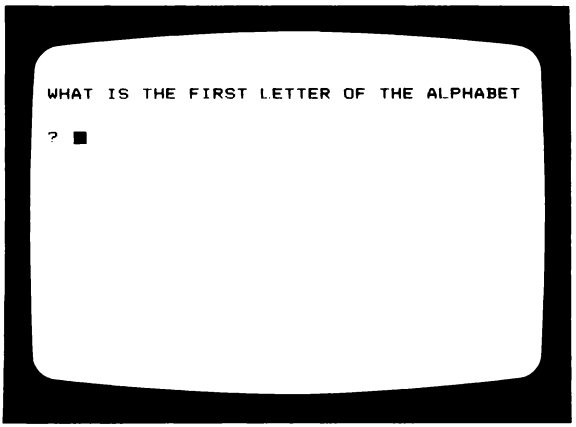
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT "  SHIFT CLR/HOME  "
20 PRINT "WHAT IS THE FIRST
LETTER OF THE ALPHABET"
30 INPUT A$
40 PRINT "WHAT IS THE LAST
LETTER OF THE ALPHABET"
50 INPUT Z$
60 IF A$="A" AND Z$="Z" GOTO
80
70 GOTO 20
80 PRINT "BOTH CORRECT"
```

Type RUN then press RETURN .

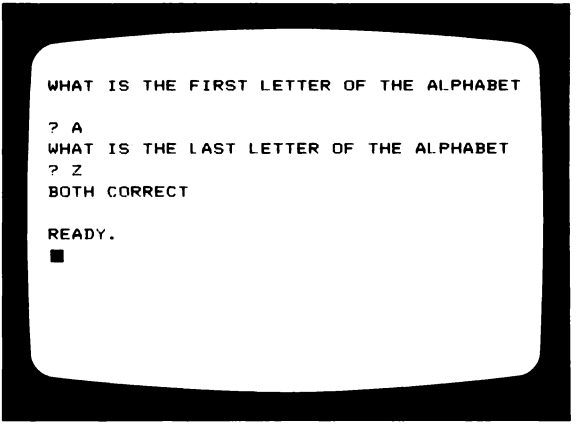
This is what will appear on the screen:



You try

Type A then press RETURN .
Type Z then press RETURN .

This is what will appear on the screen:



You try

Run the program again.
Type in one or two incorrect letters and notice what happens.

You try

Make up your own programs using AND to do the following:

1. Check for two possible answers to a question.
2. Select two words from a list of words.
3. Select two numbers from a list of numbers.

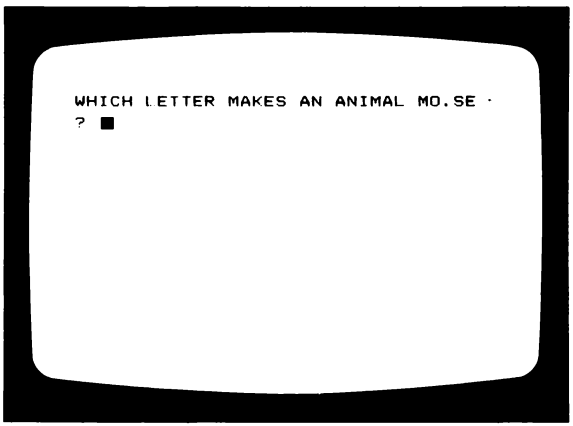
Programs using OR

OR is used to check for one of two conditions.

You try

Type NEW then press RETURN .
Type in the following program.
10 PRINT " SHIFT CLR/HOME .
20 PRINT "WHICH LETTER MAKES AN ANIMAL MO.SE"
30 INPUT L$
40 IF L$="U" OR L$="O" THEN PRINT "CORRECT":END
50 GOTO 20
Type RUN then press RETURN .

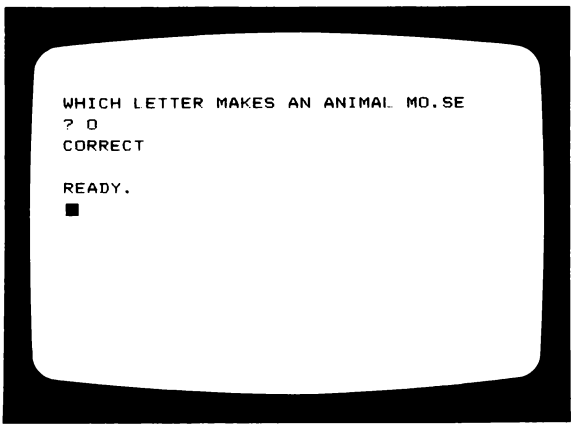
This is what will appear on the screen:



You try

Type O then press RETURN .

This is what will appear on the screen:

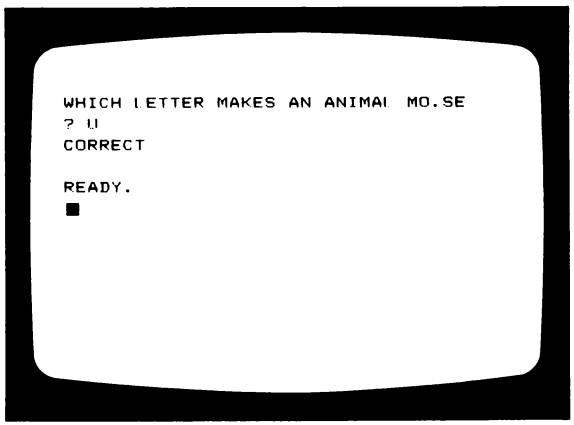


Run the program again.

You try

Type U then press RETURN .

This is what will appear on the screen:



You try

Run the program a few more times. Type in incorrect answers and notice what happens.

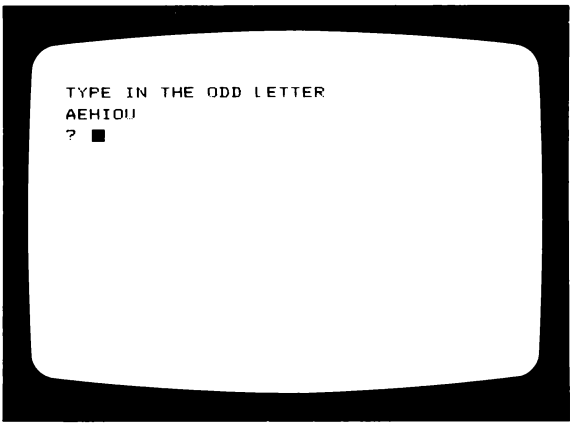
You try

Type NEW then press RETURN .
Type in the following program.

```
10 PRINT "  SHIFT CLR/HOME  "
20 PRINT "TYPE IN THE ODD
LETTER"
30 PRINT "AEHIOU"
40 INPUT L$
50 IF L$="H" OR L$="E" GOTO
70
60 GOTO 20
70 PRINT "CORRECT"
```

Type RUN then press RETURN .

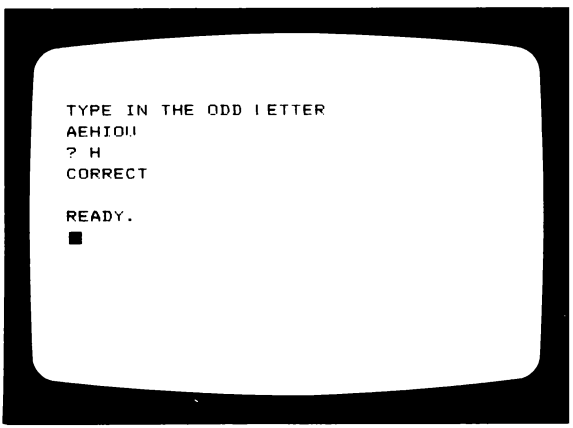
This is what will appear on the screen:



You try

Type H then press RETURN .

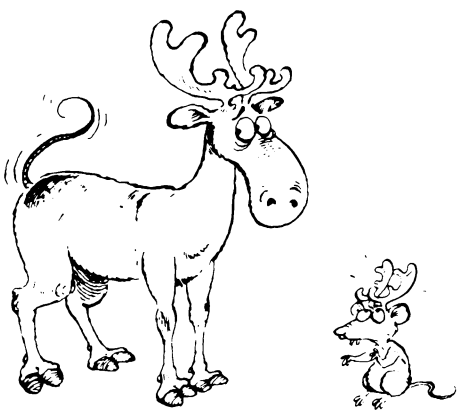
This is what will appear on the screen:



You try

Run the program again and try typing in other letters.

Make up your own programs using OR to do the following.
1. Check for one of two possible answers to a question.
2. Check an answer which may be written in either English or French.
3. Check a numerical answer which may be one of two numbers.



Project 1 – Knock, knock

The aim of this program is to make the computer tell knock, knock jokes. First we must set up some starting values and give some data. The main steps are given below.

1. Make C=Ø. (We are going to count the number of jokes so that we know when we have reached the end. This is to start the count. We shall increase the count by one each time a joke is told.)
Make J=6. (This gives the number of jokes the computer can tell.)
The data for this program can be entered as
DATA "IVOR","SORE HAND FROM KNOCKING ON YOUR DOOR","HOWARD","I KNOW"
DATA "MR","LAST BUS HOME","AMOS","QUITO", "LETTUCE","IN AND YOU'LL FIND OUT"
DATA "KEN","I COME IN"

2. Now we can set up the main part of the program. The computer should be made to

(a) ask if you want to hear a knock, knock joke;
(b) INPUT the answer;
(c) if the answer is NO then END the program;
(d) READ N$,L$ (name and last line);
(e) PRINT "KNOCK, KNOCK"
"WHO'S THERE?"
N$
N$+" WHO?"
N$+" "+L$
(This will take several PRINT statements.)
(f) Make C=C+1 (increasing the count by 1).
(g) If count <6 then go to part **(a)** so that the next joke is told.

Improvements and variations

A. Add some more jokes.

B. Make the telling of the jokes into a conversation with the computer. Then the computer will PRINT "KNOCK, KNOCK", you will INPUT "WHO'S THERE?", etc.

C. Make the computer PRINT a message when it has run out of jokes to tell.

D. Include a pause between telling one joke and asking if you wish to hear another. This can be done by inserting the line
FOR N+1 TO 3ØØØ:NEXT N
This makes the computer go round and round a loop which does nothing. Using a number greater than 3ØØØ would make the pause longer.

Project 2 – Countdown

In this project you will program the computer to play a game with you. In this game you start with 2Ø counters. You and the computer take it in turns to remove 1, 2, 3 or 4 counters. The one to remove the last counters so that none are left is the winner. The program can be written as follows.

1. Make the computer explain the game using PRINT.

2. Set up the number of counters to be 2Ø.

3. Have your turn which consists of the following steps.
- **(a)** Make the computer ask for the number of counters you wish to remove and INPUT the answer.
- **(b)** Take away your answer from the number of counters to give the number of counters remaining and PRINT the result.
- **(c)** Check whether the number of counters is zero. If it is then make the computer tell you that you have won and END the game.

4. Make the computer have a turn by taking the following steps.
- **(a)** If the number of counters is less than five (<5), then the computer can choose to remove all the counters. Otherwise the computer chooses to remove a random number of counters between 1 and 4: use INT(4*RND(1))+1. (Use IF/THEN and GOTO for this part.)
- **(b)** Calculate the new number of counters and PRINT the result.
- **(c)** Check whether the number of counters is zero. If it is then the computer says that it has won and ends the game, otherwise it returns to your turn (step **3**). Make sure that the name for the number of counters is the same as at the beginning of step **3**.

Improvements and variations

A. The computer could ask whether you want the first turn, and if not have the first turn itself.

B. Instead of having the computer write down the number of counters, make it draw the counters. (Try PRINT and " SHIFT Q" for this.)

C. Make the screen clearer by using PRINT " SHIFT CLR/HOME ", PRINT on its own for a blank line, and extra spaces.

D. It would be a good idea to make the computer check that you are not cheating by trying to remove too many counters or none at all.

E. Put in empty FOR . . . NEXT loops to slow the game down at the right moments.

F. When you have played the game a few times and know how to beat the computer, try making the computer play more intelligently.

Project 3 – Lions and antelopes

The purpose of this program is to set up a situation like the one faced by the warden of a game park in Africa. We shall look at the lions and antelopes in the game park. The lions kill the antelopes for food (about 2ØØ each in a year), and the warden has to choose how many lions to kill to keep the numbers under control.
The structure of the program is as follows.

1. Make the computer explain the situation for someone new coming to the program, using PRINT.

2. Give starting values for the number of lions and the number of antelopes. It is sensible to start with about 15 lions and 1ØØØØ antelopes.

3. Make the computer ask for the number of lions to be killed this year and INPUT the reply. This number is called the cull.

4. Work out the numbers of lions and antelopes for the next year using the following equations (L=lions, A=antelopes, C=cull):
L=INT(L−C+L*(A−4ØØ*L)/1ØØØØ)
A=INT(A−2ØØ*L+A*(2ØØØØ−A)/3ØØØØ)
INT makes sure that the answer is a whole number. The equation for the lions says that the new number of lions is the number from the previous year, minus the number killed, plus a number of births and deaths depending on the number of lions available to breed and the number of antelopes available as food. The equation for the antelopes is the number from the previous year, minus the number killed by lions, plus the number of births and deaths. Make the computer PRINT the new numbers of lions and antelopes.

5. Make the computer ask if you want another go, and if so then go to step **3**.

Improvements and variations

A. Alter the number of lions and antelopes at the start either by using INPUT so that the user chooses them, or by using RND so that the computer chooses them.

B. Make the computer keep a count of the year and PRINT the year number each time round. The computer could then ask if you wish to continue after every ten years, instead of every year.

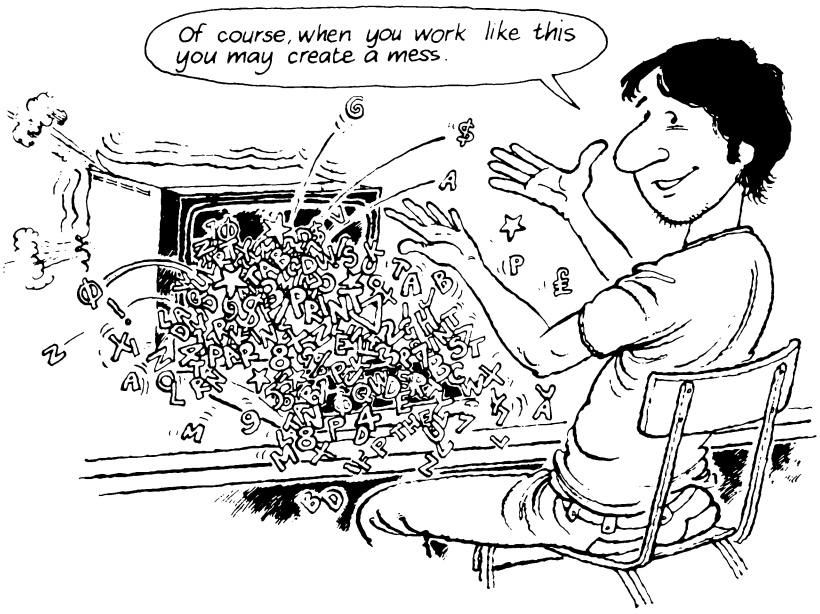
C. Include a check that the number of lions and the number of antelopes do not become negative.

D. Include messages of congratulations if the numbers of lions and antelopes become large and scold the user if they become small.

E. Make the writing on the screen clearer by using PRINT " SHIFT CLR/HOME " and PRINT on its own, and put in extra spaces between words so that no words are split between lines.

F. If you feel ambitious you could try drawing a picture of the park, using symbols for lions and antelopes (say one symbol to represent a lion and another to represent 1ØØ antelopes).

SPECIAL FEATURES



Mort Puce

My advice

My advice is to look at new commands carefully. When you think you understand the idea, experiment and try things out. You will learn what looks right. Of course, when you work like this you may create a mess. Don't be put off: try again. Remember your successes and forget about your failures.

Try out all the ideas you can think of. Explore the computer's graphics characters, shapes and colours. You can draw almost anything on the screen and colour it in different ways. Try out the Commodore's function keys, and practise using the real-time clock.

When you hit on a good idea make sure you write some notes about it, so that you will be able to use the idea again.

Mort Puce

Graphics and colour

The Commodore 64 can do more than print numbers and letters on the screen. It can also draw pictures in 16 different colours. These pictures are usually called 'graphics', and they can be produced in several ways. We will look at three of these ways: printing the special graphics characters directly from the keyboard; using CHR$ codes; and POKEing the screen memory.

Graphics from the keyboard

When we were learning to use the keyboard, we discovered that by holding down the SHIFT key we can print the graphics characters shown on the right front face of many keys. By holding down the C= key we can print the left-hand graphics characters.

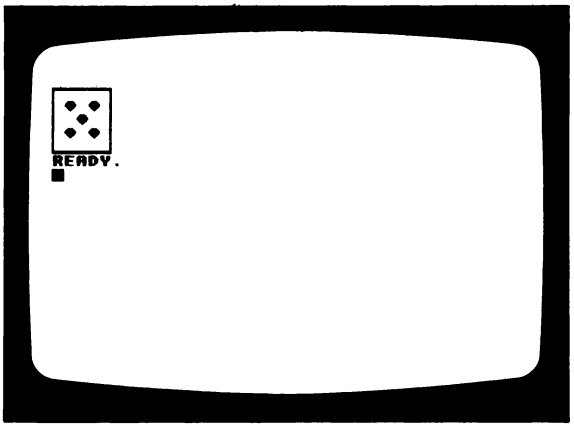
You try

Type NEW then press RETURN. Using the SHIFT and C= keys together with the keys O, Y and P (for line 2Ø), H, S and N (for lines 3Ø, 4Ø and 5Ø), and L, P and @ (for line 6Ø), type in the following program.

```
10 PRINT "C"
20 PRINT "
30 PRINT "
40 PRINT "
50 PRINT "
60 PRINT "
```

You will see the card being built up as you type in the lines of the program. When you have entered the program correctly, type RUN then press RETURN.

This is what will appear on the screen:



Make a start
On computer art.

There is one thing wrong with our card program: it printed the card in light blue, but everyone knows that hearts are red!

You try

Do not type NEW! Type in this extra line.
15 PRINT " CTRL 3"
You should see a pound sign appear in reverse video where you pressed CTRL and 3.
Type RUN then press RETURN .

The 'You try' exercise prints the card in red instead of light blue.

The reversed pound sign is called a 'control character'. When you are typing in a program and you press CTRL together with one of the keys 1 to 8 inside quote marks, a symbol will appear in reverse video. But when you run the program, you will not see the symbol – just its effects.

CTRL and the number keys 1 to 8 are used to change the colour of what is printed on the screen. Outside quote marks they work immediately without printing a control character. Try pressing CTRL and a number, then type a few words. Try another number, and do the same again. Notice that the printing changes to the colour which is written in shortened form on the front of each number key (BLK stands for BLACK, WHT for WHITE, and so on).

Pressing C= together with keys 1 to 8 produces another eight colours, though there is nothing on the keys to say so. Try them and see. The complete set of colours can be obtained as follows:

CTRL	1	black	C=	1	orange
CTRL	2	white	C=	2	brown
CTRL	3	red	C=	3	light red
CTRL	4	cyan	C=	4	dark grey
CTRL	5	purple	C=	5	medium grey
CTRL	6	green	C=	6	light green
CTRL	7	blue	C=	7	light blue
CTRL	8	yellow	C=	8	light grey

Using CHR$ codes

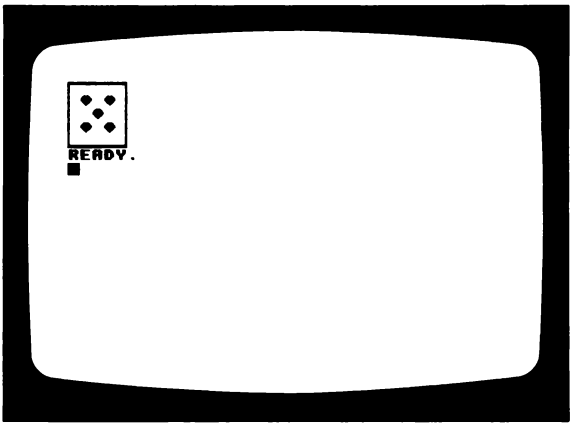
You try

Type NEW then press RETURN .
Type in the following program.

```
1Ø PRINT " SHIFT CLR/HOME "
2Ø DATA 111,183,183,183,112
3Ø DATA 165,115,160,115,167
4Ø DATA 165,160,115,160,167
5Ø DATA 165,115,160,115,167
6Ø DATA 1Ø8, 175,175,175,186
7Ø PRINT CHR$(28)
8Ø FOR L=1 TO 5
9Ø READ A, B, C, D, E
1ØØ PRINT CHR$(A);CHR$(B);
CHR$(C);CHR$(D);CHR$(E)
11Ø NEXT L
```

Type RUN then press RETURN .

This is what will appear on the screen:



This program produces exactly the same result as the last one we tried. Even though it is longer, the method it uses has some advantages. But first, how does the program work?

Every letter, number and graphics character has a code number given to it. These numbers are called ASCII or CHR$ codes. For example, the CHR$ code for capital A is 65. Try typing
PRINT CHR$(65)
then press RETURN . The computer will print A. In our program, the shapes that make up the hearts and the card borders have CHR$ codes 111, 183 and so on. So when the computer has read the data, it prints the shapes corresponding to the CHR$ codes.

In fact, CHR$ codes can be used to make the computer do other things besides print characters. Notice line 7Ø in our program: CHR$(28) does the same job as CTRL 3 – it tells the computer to print in red.

You try

Type NEW then press RETURN .
Type in the following program.

```
10 POKE 53281,12
20 PRINT " SHIFT CLR/HOME "
30 PRINT CHR$(5) "WHITE"
40 PRINT CHR$(28) "RED"
50 PRINT CHR$(30) "GREEN"
60 PRINT CHR$(31) "BLUE"
70 PRINT CHR$(144) "BLACK"
80 PRINT CHR$(156) "PURPLE"
90 PRINT CHR$(158) "YELLOW"
100 PRINT CHR$(159) "CYAN"
```

Type RUN then press RETURN .

The program prints the words in the appropriate colours. Line 10 is included so that the screen becomes grey and provides a good background for the colours. We will see how it works in the next section. Meanwhile, if you find the colours distorted in any way you may have to adjust your TV or monitor.

Make a note

COLOUR	CHR$
white	5
red	28
green	30
blue	31
black	144
purple	156
yellow	158
cyan	159

You can find a complete list of CHR$ codes and what they stand for in the **Commodore 64 User Manual 135-137**.

The main advantage of using CHR$ codes instead of typing characters directly from the keyboard is that you can make the computer do a lot of work for you. Since CHR$ codes are numbers, the computer can add to them, subtract from them and so on.

You try

Type NEW then press [RETURN] .
Type in the following program.

```
10 PRINT " [SHIFT] [CLR/HOME] "
20 FOR C=33 TO 127
30 PRINT CHR$(C);
40 NEXT C
50 FOR C=161 to 191
60 PRINT CHR$(C);
70 NEXT C
```

Type RUN then press [RETURN] .

The 'You try' exercise prints the Commodore 64's entire character set in upper-case/graphics mode. Press [C=] and [SHIFT] together to see the character set in lower-case mode.

You try

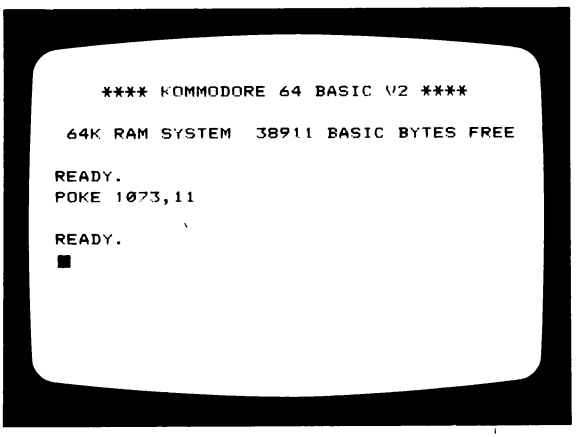
Use CHR$ codes to draw a house, a road sign or a person, and colour your picture.

POKEing the screen memory

You try

Switch the computer off and then on again so that the 'welcome' message appears. Now type POKE 1073,11 then press [RETURN] .

This is what will appear on the screen:



All the positions on the screen are numbered starting at 1Ø24 (the top left-hand corner) and finishing at 2Ø23 (the bottom right-hand corner).

1Ø73 is the position of the letter C in COMMODORE when the computer is first switched on. 11 is the 'screen code' for the letter K. So POKE 1Ø73,11 tells the computer to put the letter K where C was before.

Beware! Screen code numbers are not the same as CHR$ code numbers, so do not get them confused. A complete list of screen codes is given in the **Commodore 64 User Manual 132-134.**

You try

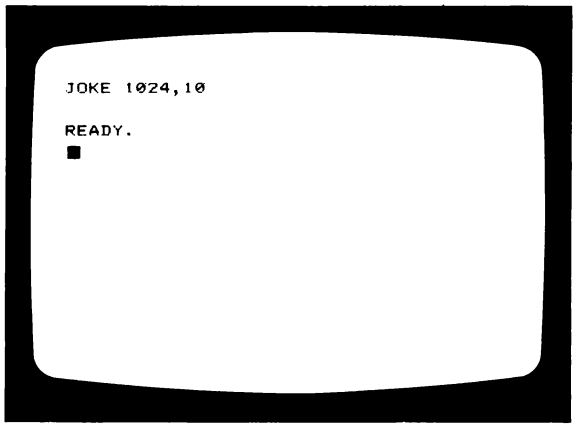
Type POKE 1Ø73,3 then press RETURN .

The screen returns to the original display.

This is what will appear on the screen:

You try

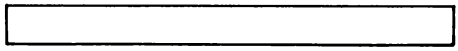
Clear the screen by pressing SHIFT and CLR/HOME . Now type POKE 1Ø24,1Ø then press RETURN .



As you would expect, 1Ø is the screen code for J. If you look it up, you will find that 81 is the screen code for a solid circle. 1524 is the number for the location at the centre of the screen. So typing POKE 1524,81 should print a solid circle in the centre of the screen. Try it.

Nothing happened!

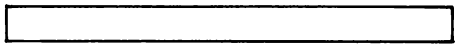
Actually, the computer did print a solid circle in the middle of the screen, but you cannot see it because it is the same colour as the background.



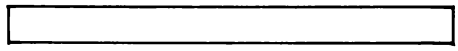
You try

Type POKE 55796,Ø then press RETURN .

A black dot appears in the centre of the screen.



As well as a screen memory map, the Commodore 64 has a colour memory map. On the colour memory map, all the screen positions are numbered starting at 55296 (the top left-hand corner) and ending at 56295 (the bottom right-hand corner).



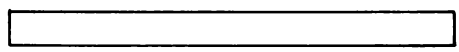
You try

Type NEW then press RETURN .
Type in the following program.

```
1Ø PRINT " SHIFT CLR/HOME "
2Ø POKE 1442,79: POKE 55714,1
3Ø POKE 1443,119: POKE 55715,1
4Ø POKE 1444,8Ø: POKE 55716,1
5Ø POKE 1482,1Ø1: POKE 55754,1
6Ø POKE 1483,9Ø: POKE 55755,2
7Ø POKE 1484,1Ø3: POKE 55756,1
8Ø POKE 1522,76: POKE 55794,1
9Ø POKE 1523,111: POKE 55795,1
1ØØ POKE 1524,122: POKE
55796,1
```

Type RUN then press RETURN .

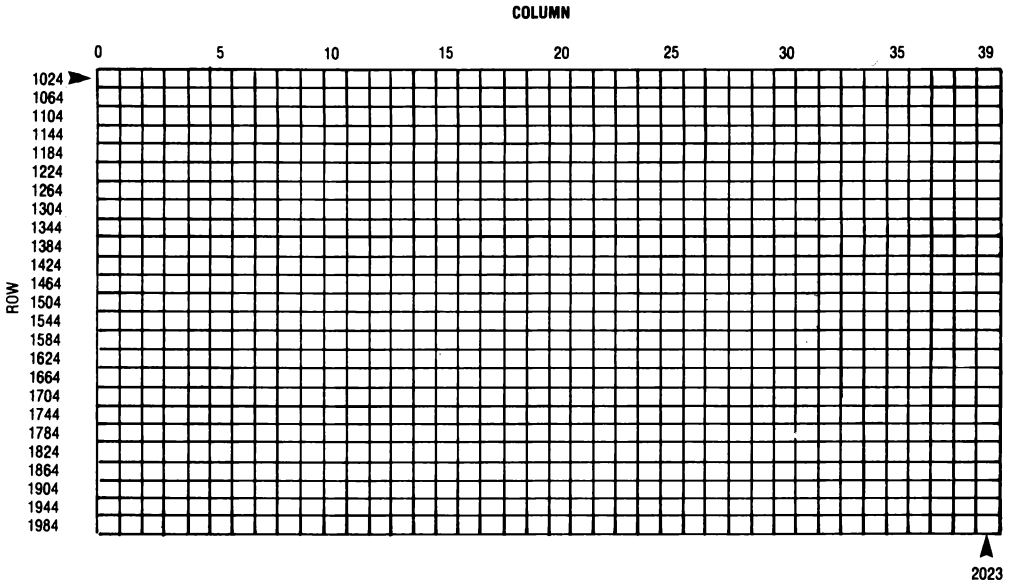
An ace of diamonds should appear in the centre of the screen. The diamond should be red and the border white.



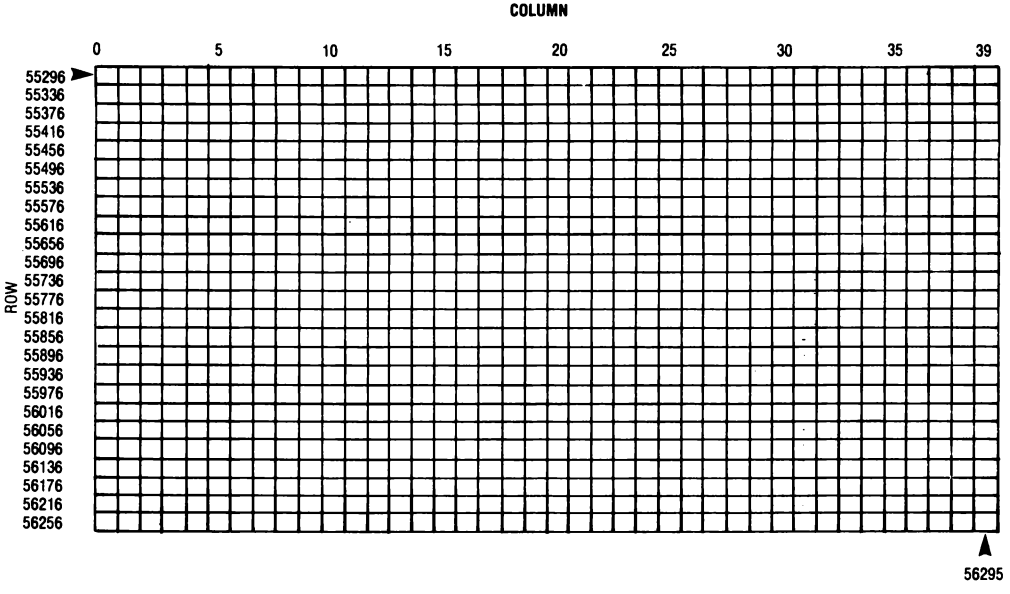
The codes for POKEing colours onto the colour memory map are as follows:

Ø black	4 purple	8 orange	12 medium grey
1 white	5 green	9 brown	13 light green
2 red	6 blue	1Ø light red	14 light blue
3 cyan	7 yellow	11 dark grey	15 light grey

Here is a grid showing the screen memory map:



And this grid shows the colour memory map:



To find the number of a particular area on the screen memory map, add the row number and the column number

together. For example, to find the bottom right-hand corner:

row number 1984
column number 39
2Ø23

To find the number of an area on the colour memory map, once again add the row number and column number together. An easier way is sometimes to add 54272 to the screen memory number. So to find the bottom right-hand corner of the colour memory map:

screen memory 2Ø23
+54272
56295

It is possible to change the colours of the border and background, too.

You try

Type POKE 5328Ø,11
then press RETURN .

The border colour will change to dark grey.

You try

Type POKE 53281,12
then press RETURN .

The background colour will change to medium grey.

You try

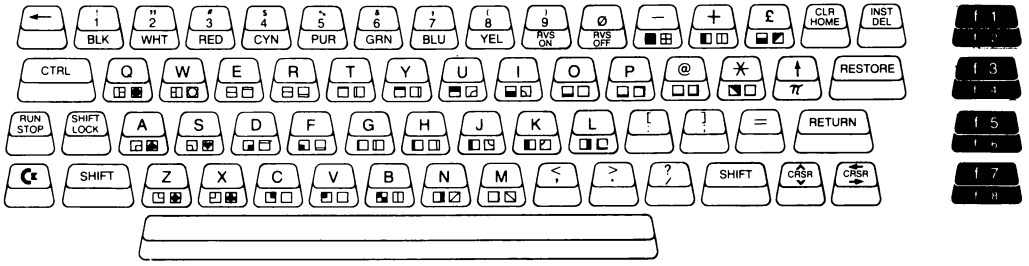
Using POKE, write a program to draw a traffic light which switches the red, orange and green lights on and off in the correct order.

Make a note

To change the border colour, type POKE 5328Ø, followed by the number of the colour you want. To change the background colour, type POKE 53281, followed by the appropriate number.

Function keys

The Commodore 64 has four special keys, coloured brown, which are on the right-hand side of the keyboard. These are the function keys. They are marked f1/f2, f3/f4, f5/f6 and f7/f8.



The top key is f1 when it is pressed on its own, and f2 when pressed together with SHIFT . The other keys work in the same way. These keys can be made to do various things and are sometimes called user-programmable keys.

You try

Type NEW then press RETURN .
Type in the following program.

```
10 GET A$: IF A$="" GOTO 10
20 IF A$=CHR$(133) GOTO 200
30 IF A$=CHR$(134) GOTO 300
40 IF A$=CHR$(135) GOTO 400
50 IF A$=CHR$(136) GOTO 500
200 POKE 53281,0
250 GOTO 10
300 POKE 53281,1
350 GOTO 10
400 POKE 53281,2
450 GOTO 10
500 POKE 53281,3
550 GOTO 10
```

Type RUN then press RETURN .

Nothing much seems to happen, except that the cursor has disappeared. Press RUN/STOP , and the message 'BREAK IN 1Ø' will appear.

The command GET A$ in line 1Ø tells the computer to take the first character that is typed and store it in the address A$. If you are not pressing a key when the computer comes to this command, it assumes you want A$ to be empty, and moves on to the next command.

This all happens in a flash, so the second half of line 1Ø is needed to make the computer keep looking until a key is pressed.

You try

Run the program again. Press the keys f1, f3, f5 and f7 in turn.

The 'You try' exercise changes the screen colour to black, white, red and cyan as the function keys are pressed.

You try

Change lines 2ØØ, 3ØØ, 4ØØ and 5ØØ to produce different colours. Add some extra lines to the program so that pressing f2, f4, f6 and f8 will change the colour of the screen border.

Make a note

The CHR$ codes for the function keys are as follows:

f1 CHR$(133)
f3 CHR$(134)
f5 CHR$(135)
f7 CHR$(136)
f2 CHR$(137)
f4 CHR$(138)
f6 CHR$(139)
f8 CHR$(14Ø)

Real-time clock

Another special feature of the Commodore 64 is its real-time clock.

You try

Type NEW then press [RETURN] .
Type in the following program.

```
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø INPUT TI$
3Ø PRINT " [SHIFT] [CLR/HOME] "
4Ø PRINT TAB(17) TI$
5Ø PRINT CHR$(145);
6Ø GOTO 4Ø
```

Type RUN then press [RETURN] .

TI$ is the address where the computer stores the time in hours, minutes and seconds. CHR$(145) is the code which tells the computer to go back up one line. This stops the display from scrolling down the screen.

When you run this program a question mark and the cursor will appear on the screen. To set the time, type in the hours, minutes and seconds as three two-figure numbers in a row. For example, if you want to set the time at 8.5Ø a.m., type in Ø85ØØØ then press [RETURN] . The time will appear at the top of the screen, and you will be able to watch the seconds pass by.

You try

Type NEW then press [RETURN] .
Type in the following program.

```
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø TI$="ØØØØØØ"
3Ø IF TI$>"ØØØØ1Ø" THEN PRINT
TAB(12) "YOUR TIME IS UP!":END
4Ø GOTO 3Ø
```

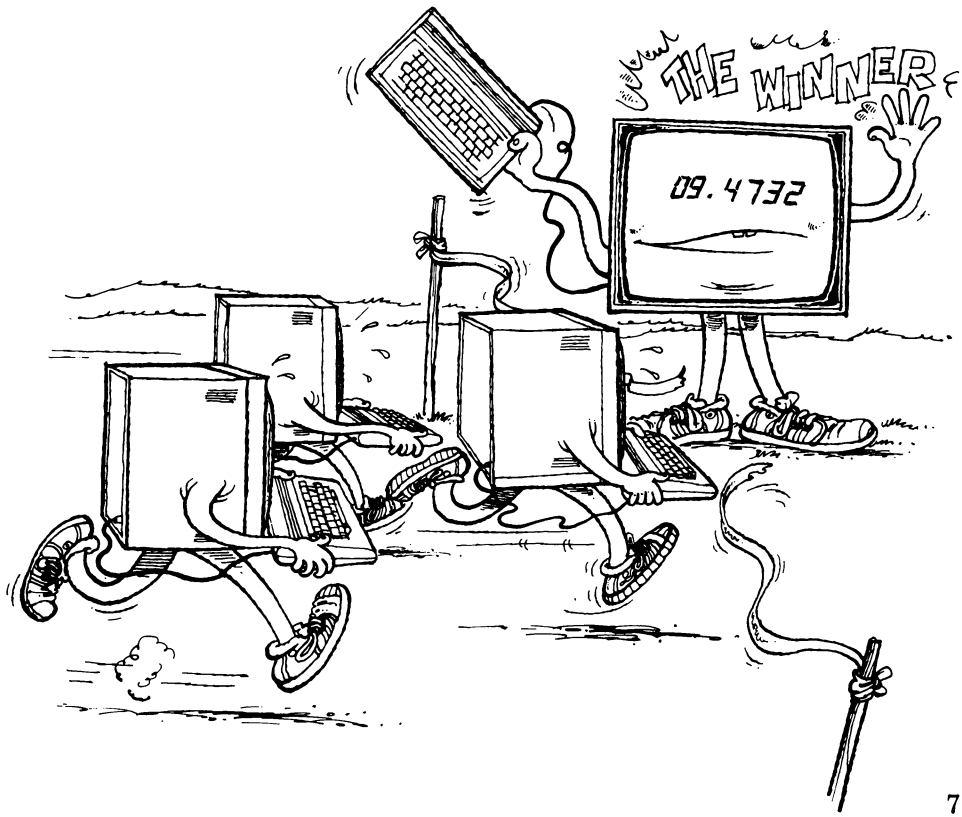
Type RUN then press [RETURN] .

The screen will go blank and then the message will appear after ten seconds. The program shows you how to reset the computer's internal clock and use it to decide when a particular command should be obeyed.

You try

Type NEW then press [RETURN] .
Type in the following program.
1Ø PRINT " [SHIFT] [CLR/HOME] "
2Ø PRINT TAB(3) "PRESS SPACE BAR TO START AND STOP"
3Ø GET K$: IF K$="" GOTO 3Ø
4Ø TI$="ØØØØØØ"
5Ø PRINT " [SHIFT] [CLR/HOME] "
6Ø GET K$: IF K$="" GOTO 6Ø
7Ø PRINT TAB(12)
TI/6Ø"SECONDS"
Type RUN then press [RETURN] .

When you run this program you will need to press the space bar to start the stopwatch, and then press it again to stop the stopwatch. TI (in line 7Ø) counts the time in 1/60ths of a second. It is set to Ø when you switch the computer on, or when you reset TI$.



Project 4 – Reaction timer

The idea of this game is to measure how quickly you or your friends can read a letter flashed on the screen and press the right key. The program can be written as follows.

1. Make the computer explain the game, using PRINT.

2. Clear the screen and reset the timer to ØØØØØØ.

3. Make the computer select a letter at random. There are 26 letters, and the CHR$ code for the first letter, A, is 65. So the formula you need is
L=INT(RND(1)*26)+65

4. Get the computer to print CHR$(L) on the screen.

5. Using GET A$, have the computer read which key the player presses. If it is the wrong key, the computer should keep on looking. When the correct key is pressed, the message 'WELL DONE!' should be printed on the screen together with the number of seconds the player has taken.

Improvements and variations

A. The computer could print the letter at a random position and after a random delay so that the player does not know when or where to expect it.

B. You could rub out the letter after a short delay. Using a FOR . . . NEXT loop which counted down from quite a long delay to shorter and shorter delays, you could see how quickly a letter can be flashed on and off the screen and still be read correctly.

C. You could adapt the program so that it becomes a typing tutor. In this case it might be better to prepare a series of sentences and put them in DATA statements. Then they could be printed in turn. Using INPUT instead of GET, the computer could compare what is typed with the sentence it has printed, and say whether it is right or wrong. You could also limit the amount of time allowed to type in the sentence. This limit could be made shorter and shorter as you became a better typist.

Project 5 – Street scene

The idea of this project is to draw a picture on the screen. It is up to you what you draw – the main steps will always be the same. But let us imagine a street with houses, a tree and a car.

1. The first thing to do is draw your picture on a piece of squared paper. Graph paper is best, because you will need plenty of squares. If you are going to draw on the screen using PRINT, then the columns will be marked from Ø to 39 and the rows from Ø to 24. If you are going to POKE your picture onto the screen, you will need two grids just like the ones shown on page 72. Or you may decide to use a mixture of PRINTs and POKEs, in which case each square must be numbered three different ways.

2. Colour your picture on the paper just as you want it to appear on the screen.

3. Now start programming. First select the background colour that you want. It saves trouble if you choose for the background the colour you are using most. Blue may be the commonest colour if you are going to have plenty of sky. A line saying
POKE 53281, colour number
followed by a line clearing the screen will do the trick.

4. If you are using PRINT, you should start at the top left-hand corner of the screen (if you only use POKE, it does not matter where you start). Choose suitably shaped graphics characters and build up your picture line by line, remembering to insert commands changing the PRINT colour at the right places. By the way, you can print solid squares by printing reverse-video spaces.

5. It saves a great deal of work if you can make the computer print regular shapes using FOR . . . NEXT loops. But if your picture is irregular, you will have to work things out yourself one square at a time.

6. When your picture is complete, you should stop the computer from ending the program and printing READY by adding a line such as
1ØØØ GOTO 1ØØØ
which will set up an endless loop. You can then escape from the program by pressing RUN/STOP .

Improvements and variations

It would make things much more interesting if you brought movement or animation into the picture. Of course, nothing will really move, but you can make it appear that a car is driving along the street, for example.

In order to do this, you need to draw your car in one position, then rub it out and draw it again in another position. You will find that this can be done much more quickly if you use POKE rather than PRINT.

The basic idea is to get the computer to do the work by using variables to refer to screen positions. For the sake of simplicity, imagine moving just one wheel of your car, but remember that the whole car can be defined by referring to one part. In our example, the car's back wheel is to start at position 1785. If you call this position P, then the front wheel might be at position P+6.

1. Set up a FOR . . . NEXT loop to count from P=1785 to the last position that you want the wheel to reach. You probably will not want to move it one position at a time, so use STEP 3, say.

2. POKE P,1Ø5 (1Ø5 is the POKE number for a solid circle), and POKE 54272+P with the colour code that you want. POKE the rest of the car onto the screen by adding or subtracting from P.

3. Set up an empty loop to keep the car on the screen for a suitable amount of time (trial and error will show how long this should be), then POKE 54272+P with the colour code for the background colour. This makes the wheel disappear. Do the same for the rest of the car.

4. NEXT count of the main loop. The computer will automatically work out where everything should be drawn next.

Pictures on the Commodore 64 are usually animated by using sprites. These are very useful objects, but they are a little more complicated to program. The **Commodore 64 User Manual** introduces sprites on page 68. If you are not using sprites, you must be careful that your moving object

does not rub out anything important as it goes along. So make sure that your car drives over a plain background!

Other ideas for computer programs

1. A game of chance which uses random numbers
2. A spelling test
3. A list of questions on a particular subject
4. A card game
5. A questionnaire
6. A coloured diagram
7. A science experiment
8. A cash register
9. An anagram finder
10. A language translator
11. A passage in which missing words have to be filled in
12. A pattern designer
13. A list of names and addresses
14. A plan or scale drawing
15. A word game
16. An alphabetical sorter
17. A number game with questions and answers
18. A foreign money exchange
19. A cartoon
20. A maze
21. A rocket launcher
22. A TV advert
23. A guessing game

You may need to look back at some of the programs in the book or ones which you have written yourself.
It may also be helpful to check certain things in the reference sections from time to time.

FINDING OUT

Ms O.C. Termup

My advice

Always check the terms you use just as you would check the meaning of unusual words with a dictionary. Most of the terms are fairly easy to understand, but there are a lot of them. If you are unsure then check the meaning and use. All the terms you are likely to need are contained in the eight parts of the reference section. So first find out which section the term is in and then look it up in that section. The sections are

Commands	Operators
Statements	Reserved variables
Number functions	Other functions
Word functions	Symbols

It would be very useful if you made your own reference book to contain all the terms you come across. You need not put them in separate sections, but it will make things much easier if they are in alphabetical order.

Olive C. Termup

Reference sections

The terms which you have used in the book will all be found in the reference sections, but you will also find some new terms. These are included because they may be useful when you are writing your own programs.
If you are writing a program which examines words, then you might find some of the word function terms very useful. In a mathematics program some of the number function terms will be very useful.
The reference sections only give a brief outline of the meaning and use of the terms and if you wish to find out more about them you will have to look them up in other books. The list below is given for this purpose.

BASIC Made Easy for Your Commodore 64
Garry Marshall
Arrow Books

Getting the Most from Your Commodore 64
Simon Potter
Penguin Books

How to Program the Commodore 64
Robert Young
Interface Publications

Learning to Use the Commodore 64 Computer
William Turner
Gower Publishing

Commodore 64 Programmers' Reference Guide
Commodore

There are also a number of computer magazines which can be very helpful. Quite regularly they list programs which you might find interesting and you may well be able to adapt these programs to your own needs. Some suitable magazines are

Commodore Horizons
Computer and Video Games
Personal Computer News
Personal Computing Today
Your Commodore
Your Computer

Here is a list of all the terms that the Commodore 64 computer understands. The ones in heavy type are those which are likely to be most useful to you. These are the ones described more fully in the following pages. Wait to use the other terms until you have learnt more about BASIC.

1. Commands

CONT
LIST
LOAD
NEW
RUN
SAVE
VERIFY
GET
GET#
GOSUB
GOTO
IF
INPUT
INPUT#
LET
NEXT
ON
OPEN
POKE
PRINT
PRINT#
READ
REM
RESTORE
RETURN
STEP
STOP
SYS
THEN
TO
WAIT

2. Statements

CLOSE
CLR
CMD
DATA
DEF FN
DIM
END
FOR

3. Number functions

ABS
ATN
COS
EXP
INT
LOG
PEEK
RND
SGN
SIN
SQR
TAN
USR

4. Word functions

ASC
CHR$
LEFT$
LEN
MID$
RIGHT$
STR$
VAL

5. Operators

AND
NOT
OR

6. Reserved variables

ST
TI
TI$

7. Other functions

FRE
POS
SPC
TAB

8. Symbols

%
$
+
−
*
/
↑
<
>
<=
>=
<>
=
π
.
;
,
" and "
:

Commands

CONT is used to restart a program. The program will be restarted from where it was stopped.

Commodore 64 User Manual 114

LIST lists a program. (See page 35.)
LIST lists the entire program.
LIST 1Ø lists line 1Ø only.
LIST 4Ø–8Ø lists lines from 4Ø to 8Ø inclusive.
LIST –5Ø lists lines up to and including 5Ø.
LIST 2ØØ– lists lines after (and including) 2ØØ.

Commodore 64 User Manual 115

LOAD loads a program from a cassette or disk.
LOAD"WATER" will load a program called WATER.

Commodore 64 User Manual 115

NEW sets up the computer for a new program to be typed or loaded in. It removes any existing program.

Commodore 64 User Manual 115

RUN tells the computer to run through a program and carry out the instructions in it.

Commodore 64 User Manual 116

SAVE saves a program onto a cassette or disk.
SAVE"MISSILE" saves the program presently in memory and entitles it MISSILE.

Commodore 64 User Manual 116

VERIFY checks that a program has been saved correctly.
VERIFY"WORDS" checks that the program called WORDS has been saved correctly.

Commodore 64 User Manual 116

Statements

CLR clears all the variables written in a program.

Commodore 64 User Manual 117

DATA is used with READ to supply data for a program. (See page 47.)
DATA 1,2,3,4,5
DATA APPLE,BANANA,CARROT,DATE

Commodore 64 User Manual 118

DEF FN is used to define a function.

Commodore 64 User Manual 118

DIM allows groups of words or numbers to be put into the computer.
DIM X(2Ø) allows numbers to be put into the computer with addresses X(Ø), X(1), X(2) . . . X(2Ø).
DIM X$(2Ø) allows words to be put into the computer with addresses X$(Ø), X$(1), X$(2) . . . X$(2Ø).

Commodore 64 User Manual 118

END tells the computer that the program has reached the end.

Commodore 64 User Manual 119

FOR is used with TO, STEP and NEXT to repeat the same lines many times. If STEP is left out, the computer assumes a step of 1. (See page 46.)
FOR X=Ø TO 1Ø STEP 2:NEXT X
gives X the values Ø, 2, 4, 6, 8, 1Ø taken in order.

Commodore 64 User Manual 119

GET checks the keyboard and, if a key is pressed, stores the letter or symbol represented by the key in the address supplied. (See page 74.)

```
1Ø GET A$:IF A$="" THEN GOTO 1Ø
2Ø IF A$="Y" THEN PRINT "YES":END
3Ø GOTO 1Ø
```

Commodore 64 User Manual 47, 119

GOSUB is a program instruction to go to a subroutine which starts at a given line number. (See page 54.)
GOSUB 55Ø instructs the program to go to a subroutine which starts at line number 55Ø. RETURN is used at the end of the subroutine.

Commodore 64 User Manual 12Ø

GOTO is a program instruction to go to a specified line number, skipping out any lines in between. (See page 52.)
GOTO 41Ø tells the computer to go to line 41Ø.

Commodore 64 User Manual 12Ø

IF is used with THEN to tell the computer to carry out a specified task if a condition holds true. (See page 50.)
IF A$="GOOD" THEN GOTO 6ØØ.

Commodore 64 User Manual 12Ø

INPUT allows words or numbers to be put into the computer. (See page 42.)
INPUT A (for numbers).
INPUT A$ (for words).

Commodore 64 User Manual 121

NEXT is used with FOR, TO and STEP. (See page 46.)

Commodore 64 User Manual 121

ON is used for options.
ON X GOTO 1Ø,2Ø,3Ø,4Ø means that if X=1 the computer goes to line 1Ø, if X=2 the computer goes to line 2Ø, and so on.
ON X GOSUB 1ØØ,2ØØ,3ØØ works in the same way, except that the computer goes to the appropriate subroutine.

Commodore 64 User Manual 122

POKE is used to put numbers into particular memory locations. This instruction is often used to alter the screen display and to create sounds. The first number following POKE is the memory location, and the second is the number being inserted. (See page 69.)
POKE 5328Ø,X changes the border colour to the colour given by the value of X.
POKE 53281,X changes the background colour.
POKE followed by a number between 1Ø24 and 2Ø23 and then a value for X will alter the screen memory map.
POKE followed by a number between 55296 and 56295 and then a value for X will alter the colour memory map.

Commodore 64 User Manual 6Ø, 79

PRINT is used for printing characters on the screen. (See pages 23 and 38).
PRINT 5,6 gives 5 6
PRINT 5;6 gives 5 6
PRINT 5,,,,6 gives 5
6
PRINT "WORDS" prints WORDS.

Commodore 64 User Manual 123

READ is used with DATA to read information. (See page 47.)
1Ø READ A$,B$,C$,D$,E$
2Ø DATA RED,GREEN,ORANGE,BLUE,PINK
3Ø PRINT B$
prints GREEN.

Commodore 64 User Manual 124

REM is used for putting a comment or REMark into a program. The computer ignores any writing after REM in the same line. (See page 54.)
REM This is a remark.

Commodore 64 User Manual 124

RESTORE resets the pointer which tells the computer the next piece of DATA to READ. After RESTORE, the pointer returns to the beginning of the list of data.

Commodore 64 User Manual 124

RETURN is used with GOSUB in order to return from a subroutine to the main body of a program. (See page 54.)

Commodore 64 User Manual 124

STEP is used with FOR, TO and NEXT to set the size of step with which the computer will count. (See page 46.)

Commodore 64 User Manual 119

STOP ends the running of a program.

Commodore 64 User Manual 125

THEN is used with IF. (See page 50.)

Commodore 64 User Manual 12Ø

TO is used with FOR, STEP and NEXT. (See page 46.)

Commodore 64 User Manual 119

Number functions

ABS gives the positive value of a number.
PRINT ABS(−7) gives 7.

Commodore 64 User Manual 125

ATN gives the angle whose tangent is known. The result will be in radians.

Commodore 64 User Manual 125

COS gives the cosine of an angle (which must be stated in radians).

Commodore 64 User Manual 126

EXP gives e (2.71827183) to any power.

Commodore 64 User Manual 126

INT gives the integer (whole number) part of a number.
(See page 48.)
INT 45.61 gives 45.
INT −17.22 gives −18.

Commodore 64 User Manual 126

LOG gives logarithms of numbers to base e.

Commodore 64 User Manual 126

PEEK reads a number from the memory location specified.
PRINT PEEK(5328Ø) AND 15 will print the number of the screen border colour.

Commodore 64 User Manual 126

RND(1) gives a random number between Ø and 1 (but not 1). It must be used with INT to produce whole numbers.
(See pages 48 and 55 for examples.)

Commodore 64 User Manual 126

SGN gives the sign of a number.
1 if positive
−1 if negative
Ø if zero.

Commodore 64 User Manual 127

SIN gives the sine of an angle (which must be stated in radians).

Commodore 64 User Manual 127

SQR gives the square root of a number.
PRINT SQR(16) prints 4.

Commodore 64 User Manual 127

TAN gives the tangent of an angle (which must be stated in radians).

Commodore 64 User Manual 127

Word functions

ASC gives the ASCII or CHR$ code of the character specified.
PRINT ASC("A") prints 65.

Commodore 64 User Manual 128

**CHR$** gives the character whose ASCII or CHR$ code is specified. (See page 67.)
PRINT CHR$(65) prints A.

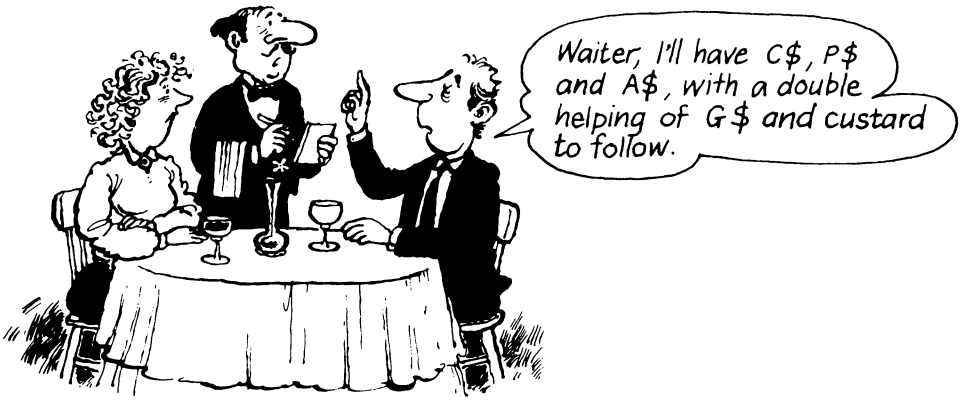
Commodore 64 User Manual 128

LEFT$ copies the left-hand part of a word. (See page 48.)
1Ø A$="FELLOW"
2Ø PRINT LEFT$(A$,4)
prints FELL.

Commodore 64 User Manual 128

LEN gives the number of characters in a word.
1Ø K=LEN ("BEAUMONT")
2Ø PRINT K
prints 8.

Commodore 64 User Manual 128



MID$ copies the middle part of a word. (See page 48.)
1Ø D$="TRIPLET"
2Ø PRINT MID$(D$,2,3)
prints RIP.

Commodore 64 User Manual 128

RIGHT$ gives the right-hand part of a word. (See page 48.)
1Ø H$="CRUMPET"
2Ø PRINT RIGHT$(H$,3)
prints PET.

Commodore 64 User Manual 128

Operators

AND is used to check whether two conditions hold. (See page 56.)
IF X>=Ø and X<1Ø THEN PRINT "DIGIT"

Commodore 64 User Manual 114

NOT is used with IF and THEN to test if something is not true.
IF NOT A=2 THEN PRINT "NO" will print NO if A does not equal 2.

Commodore 64 User Manual 114

OR is used to check whether one or another condition holds. (See page 58.)
IF A=3 OR B=6 THEN PRINT "YES" will print YES if either A=3 or B=6 (or both).

Commodore 64 User Manual 114

Reserved variables

TI counts the time, in 1/60ths of a second, since the computer was switched on, or since TI$ was reset to "ØØØØØØ". PRINT TI/6Ø displays the number of seconds. (See page 77.)

Commodore 64 User Manual 113

TI$ gives the time as a six-figure number. (See page 76.)

Commodore 64 User Manual 113

Other functions

FRE is used to find the amount of memory which is still available for use. PRINT FRE(Ø) displays this value.

Commodore 64 User Manual 129

POS gives the position of the cursor across the screen.

Commodore 64 User Manual 129

SPC prints a number of spaces on the screen.
PRINT SPC(2Ø) 4 prints 2Ø spaces before the 4.
Try the following:
PRINT SPC(Ø)"LEFT"
PRINT SPC(17)"MIDDLE"
PRINT SPC(35)"RIGHT"

Commodore 64 User Manual 129

TAB can only be used with PRINT. It is used to position words or numbers at particular points across the screen. (See page 39.)
PRINT TAB(9) will print spaces up to the ninth column.

Commodore 64 User Manual 129

Symbols

% is used to indicate that the variable marked takes only whole-number – integer – values.

$ is used to mark string variables – words – rather than numeric or integer variables.
See page 30.

+ is the sign for add.
See page 28.

– is the sign for subtract or take away.
See page 28.

***** is the sign for multiply or times.
See page 29.

/ is the sign for divide or share.
See page 29.

↑ is the sign for exponentiation. 5↑3 means 5*5*5.

< is the sign for 'is less than'.

> is the sign for 'is greater than'.

<= means 'is less than or equal to'. For example, 1<=2 and 1<=1.

>= means 'is greater than or equal to'. For example, 4>=2 and 4>=4.

<> means 'is not equal to'.

= is the sign for 'equals'. It is also often used to give a value to a variable.
See page 40.

π is the sign for PI, which equals 3.14159265. The circumference of a circle of radius R is 2*π*R.

. is used as a decimal point.

; is used with PRINT to make the printing carry on without leaving any spaces between words, and leaving only two spaces between numbers.
See page 25.

, is used with PRINT to space out words or numbers.
See page 25.

" and " are used at the start and finish of a word or group of words that the computer is to store in print.
See page 24.

: is used to separate different statements in the same program line or instruction.
See page 54.

A child's guide to the
COMMODORE 64

Here are five friendly experts, who'll help you to use your Commodore 64 microcomputer. See how to write your own programs – draw and animate pictures, tell jokes and even run a game park!

Pru Comet
"I'll help you understand the keyboard and learn the computer's language."

P.C. Truemo
"I'll help you make sure the computer understands you – and investigate why, when it doesn't."

Ms O.C. Termup
"I'm in charge of the reference section of this book. I'll help you look up terms and make your own reference file."

Mort Puce
"I'm an artist and I'll show you how to draw patterns and change colours with your computer."

Prof. O. Crumpet
"Ah! What a beautiful thing is a well written program! I'll show you how to plan it, write it, and run it."

Cambridge books for children

ISBN 0 521 31538 7