# Foundations 2: KEYBOARD FAMILIARISATION

We will now explain the functions of some of the keys on the computer. Those who are experienced in using computers may skip this section.

**[ENTER]**

There are two **[ENTER]** keys. Either of these keys enter the information that you have typed into the computer. After the **[ENTER]** key is pressed, a new line is started on the screen. Each instruction that you type in to the computer should be followed by pressing the **[ENTER]** key.

From now on, we will show **[ENTER]** as meaning press the **[ENTER]** key after each instruction or program line..

**[DEL]**

This key is used to delete a character to the left of the cursor on the screen (for example a letter or a number) which is not required.

Type in a b cd and you will see that the letter d is positioned to the left of the cursor. If you decide that you do not want the letter d, press **[DEL]** once and you will see the d removed. If you press **[DEL]** and continue to hold it down, the letters a b c will also be removed.

**[SHIFT]**

There are two **[SHIFT]** keys. If you press either of these and hold it down whilst typing a character, a capital letter or upper case symbol will appear on the screen.

Type in the letter a then hold down the **[SHIFT]** key and type in the letter a again. On the screen you will see:

a A

Now type in a few spaces by holding down the space bar. Try the following using the number keys which are on the top line of the keyboard, above the letter keys. Type in the number 2, then hold down the **[SHIFT]** key and type in the number 2 again. On the screen you will see:

2"

You can now see what happens when the **[SHIFT]** key is held down whilst pressing a character key. Experiment by typing any of the character keys, either on their own, or together with the [SHIFT] key.

[CAPS LOCK]

This has a similar operation to [SHIFT] except that you only have to press it once. From then on each letter that you type in will be in capitals, although the number keys will not be shifted. Press [CAPS LOCK] once, then type in:Type in:

```
abcdef123456
```

On the screen you will notice that although all the letters are shifted to capitals, the numbers have not been shifted to symbols. If you wish to type in a shifted symbol while [CAPS LOCK] is in operation, simply hold down the [SHIFT] key before pressing a character key. Press the following keys while holding down the **[SHIFT]** key:

*ABCDEF123456*

On the screen you will see:

```
ABCDEF!"#$%&
```

If you wish to return to small (lower case) characters again, press [CAPS LOCK] key once again.

If you wish to type in capital letters *and* shifted upper case symbols without having to constantly hold down the **[SHIFT]** key, this can be carried out by holding down the [CTRL] key, then pressing [CAPS LOCK] key once. Now type in:

```
abcdef123456
```

On the screen you will see:

```
ABCDEF!"#$%&
```

It is still possible to type in numbers while [CTRL] and [CAPS LOCK] are in operation, by using the number keys to the right of the main keyboard.

Holding down the [CTRL] key and pressing [CAPS LOCK] once will return you to the mode that you were previously in - CAPS LOCK or lower case. If you have returned to the CAPS LOCK mode, simply press [CAPS LOCK] once again to return to the lower case mode..

**[CLR]**

This key is used to clear a character within the cursor.

Type in A B C D E F G H. The cursor will be positioned to the right of the last letter typed (H). Now press the cursor left key **[←]** four times. The cursor will have moved four places to the left, superimposed over the top of the letter E.

Notice how the letter E is still visible within the cursor. Press the [CLR] key once and you will see that the letter E has been cleared and the letters F G H have each moved one space to the left with the letter F now appearing within the cursor. Now press the [CLR] key again and hold it down. You will see how the letter F is cleared followed by the letter G and H.

## [ESC]

This key is used to **[ESC]**ape from a function that the computer is in the process of carrying out. Pressing the [ESC] key once will cause the computer to temporarily pause in its function, and will continue again if any other key is pressed.

Pressing the [ESC] key twice will cause the computer to completely **[ESC]**ape from the function which it is carrying out. The computer is then ready for you to type in some more instructions.

Now press the [ESC] key twice..

IMPORTANT

When you reach the right hand edge of the display by entering more than 40 characters on the screen, the next character will automatically appear on the next line at the left edge of the screen. This means that you should NOT press [ENTER] as those of you accustomed to typewriters might press a carriage return towards the right edge of a page.

The computer does this automatically for you, and will react to an unwanted [ENTER] by printing an error message - usually a Syntax error, either there and then, or when the program is run.

## Syntax Error

If the message: Syntax error appears on the screen, the computer is telling you that it does not understand an instruction that you have entered. For example type in:

```
printt [ENTER]
```

On the screen you will see the message:

```
Syntax error
```

The message appears because the computer has not understood the instruction printt

If you type a mistake in the line of a program, such as:

```
10 printt "abc" [ENTER]
```

The Syntax error message will not appear until the instruction is processed by the computer when the program is run.

Type in:

run [ENTER]

On the screen you will see:

Syntax error in 10
10 printt "abc"

This message tells you in which line the error has occurred, and displays the program line together with the editing cursor so that you can correct the mistake.

Press the cursor right key [→] until the cursor is over the t in printt. Now press the [CLR] key to remove the unwanted t, then press the [ENTER] key to enter the corrected line into the computer.

Type in: run [ENTER] . ..and you will see that the computer has accepted the instruction, and has printed abc

# An introduction to AMSTRAD BASIC keywords

In Chapter 8 you will find an illustrated *description of* all the BASIC keywords found in the AMSTRAD BASIC. We will introduce some of the more commonly used BASIC keywords in this section.

## CLS

Type in: cls (clear screen). You can use either upper case (capital) or lower case (small) letters. Then press [ENTER]. You will notice that the screen clears and the word Ready with the cursor ■ will appear at the top left of the screen.

## PRINT

This is used whenever you want characters, words or figures in a program to be printed on the screen. Type in the following instruction line:

print "hello" [ENTER]

On the screen you will see:

hello

The quotation marks "" are used to tell the computer what should be printed. hello appeared on the screen as soon as the [ENTER] key was pressed. Type in cls [ENTER] to clear the screen.

## RUN

The previous example showed a single line program. Most programs have many lines. In front of each line, a number is first typed in. These numbers tell the computer the order in which to run the program. When **[ENTER]** is pressed, the line is stored in the memory until the program is run. Type:

10 *print* "he 1 lo" **[ENTER]**

Notice that when **[ENTER]** was pressed, h e 1 L o was not printed on the screen. To do this, the word r u *n* has to be typed in. Type:

run **[ENTER]**

**You** will now see h e l t o on the screen. Note that iustead of continually typing in p r i n t, *you* can use the ? symbol, for example:

10 ? "he 1 lo" **[ENTER]**

## LIST

After a program has been stored in the memory, it is possible to check what has been typed in by listing the program. Type:

L i s t **[ENTER]**

and on the screen you will see:

10 **PRINT "hel** *lo"*

which is the program stored in the memory.

Notice how the word P R I NT is now in capitals. This means that the computer has accepted P R I N T as a known BASIC keyword.
Type in c 1 s **[ENTER]** to clear the screen, Note that although the screen is cleared when you type in c 1 s **[ENTER], your** program is not erased from the computer's memory.

## GOT0

The G 0 T 0 keyword tells the computer to go from one line to another in order to either miss out a number of lines or to form a loop. Type this in:

10 *print* "hello" **[ENTER]**
**20** goto *10* **[ENTER]**

Type:

run **[ENTER]**

and you will see he llo printed continuously on the screen, one under another on the left side. To stop this program running, press **[ESC]** once. To start it again, press any other key. To stop it running so that other instructions can be typed in, press **[ESC]** twice.

Type in

```
c l s [ENTER]
```

to clear the screen.

To see the word he L Lo printed continuously on each line, one next to another filling the whole of the screen, type in the previous program but with a semi-colon ; after the quotation marks " Type:

```
10 print "het lo"; [ENTER]
20 goto 10 [ENTER]
r u n [ENTER]
```

Note that the semicolon , tells the computer to print the next character immediately following the previous one. Escape from this program by pressing [ESC] twice. Now type in line 10 again, but this time use a comma , instead of a semi-colon ;

```
10 print "hello", [ENTER]
run [ENTER]
```

You will now see that the comma , tells the computer to print the next character (or group of characters) 13 columns away from the beginning of the first group of characters. This feature is useful for displaying information in separate columns. Note, however, that if the number of characters in a group exceeds 12, the next character will be displaced forward by another 13 columns.

Again, to escape from this program press [ESC] twice. To clear the computer's memory completely, hold down the [SHIFT] [CTRL] and [ESC] keys in that order, and the computer will reset.

# INPUT

This command is used to let the computer know that it is expecting something to be typed in, for example, the answer to a question. Type the following:

```
10 input "what is your age"; age [ENTER]
20 print "you look younger than"; age "years
   old." [ENTER]
r u n [ENTER]
```

On the screen you will see:

```
what is your age?
```

Type in your age then [ENTER] If your age was 18, the screen would then show:

```
you Look younger than 18 years old.
```

This example shows the use of the i n p u t command and a number variable. The word a g e was put into the memory at the end of line 10 so that the computer would associate the word a g e with whatever numbers were typed in and would p r i n t these numbers where the word a g e is on line 20. Although we used the expression age in the above for the age variable, we could have just as easily used a letter, for example b.

*Reset* the computer to clear the memory. ([CTRL] [SHIFT] and [ESC] keys). If you had wanted an input made up of any characters (letters or letters and numbers), the dollar sign $ must be used at the end of the variable.

Type in the following program: (Note that in line 20 you must put a space after the o in h e 1 1 o and before the m in my).

```
10 input "what is your name"; name$ [ENTER]
20 print "hello "; name$" my name is Arnold" [ENTER]
run [ENTER]
```

On the screen you will see:

what *is your name?*

Type in your name then *[ENTER]*

If your name was Fred you will see on the screen:

*hello Fred my name is Arnold*

(As a point of interest, Arnold was the codename for the AMSTRAD CPC464 during its development.)

Although we used *name$* in the above for the name string variable, we could have just as easily used a letter, for example a $. Now we will combine the above 2 examples into one program.

Reset the computer by pressing [CTRL] [SHIFT] and [ESC]. Type in the.following:

```
5 cls [ENTER]
10 input "what is your name"; a$ [ENTER]
20 input "what is your age"; b [ENTER]
30 print "I must say "; a$ " you dont look";
   b"years old" [ENTER]
run [ENTER]
```

In this program we have used 2 variables, a$ for the name and b for the age. On the screen you will see:

what *is your name?*

*Now* type in your name (e.g. Fred) then [ENTER]. You will then be asked:

what *is your age?*

*Now* type in your age (e.g.18), then [ENTER]. If your name were Fred and your age 18, on the screen you will see:

*I must say Fred you dont look 18 years old*

# EDITING A PROGRAM

If any of the lines in the program had been typed incorrectly, resulting in a Syntax error or other error message, it would be possible to edit that line rather than type it out again. To show this you can type the previous program out incorrectly.

Type in the following:

```
5 clss
10 input"what is you name"; a$ [ENTER]
20 input"what is your age"; b [ENTER]
30 print"1 must say"; a$" you dont Look";
b"years of age" [ENTER]
```

There are 3 mistakes in the above program:

In line 5 we typed in c lss instead of cls.
In line 10 we typed in y o u instead of your.
In line 30 we forgot to put a space between s a y and the quotation marks ".

There are 3 methods of editing a program. The first is to simply type in the new line again. When a line is retyped and entered it replaces the same numbered line currently in the memory, Secondly, there is the edit method and finally there is the copy cursor method.

## EDIT METHOD

To correct the mistake in line 5, type:

```
edit 5 [ENTER]
```

Line 5 will then be printed with the cursor superimposed over the 5. To edit out the extra s in c L s s, press the cursor right key [→] until the cursor appears over the last s, then press the [CLR] key. You will see the s disappear. Now press;

[ENTER] and line 5 will now be corrected in the memory. Type in:

```
list [ENTER]
```

to check line 5 is now correct.

## COPY CURSOR METHOD

To correct the mistakes in line 10 and 30, hold down [SHIFT] key then press the cursor up key [↑] until the copy cursor is positioned over the very beginning of line 10. You will notice the main cursor has not moved, so there are now two cursors on the screen. Now press the [COPY] key until the copy cursor is positioned over the space between you and n a me. You will notice that line 10 is being re-written on the last line and the main cursor stops at the same place as the copy cursor. Now type in the letter r . This appears on the bottom line only.

The main cursor has moved but the copy cursor stayed where it was. Now press the **[COPY]** key until the whole of line 10 is printed. Press [ENTER] and this new line 10 will be stored in the memory. The copy cursor disappears and the main cursor positions itself under the new line 10. To correct the second mistake, hold down **[SHIFT]** and press cursor up key **[↑]** until the copy cursor appears over the very beginning of line 30.

Press [COPY] until the copy cursor is positioned over the quotations next to say. Now press the space bar once. A space will be inserted on the bottom line. Hold down the [COPY] key until the whole of line 30 is printed. Then press [ENTER]. You can now list the program to check that it is corrected in the memory by typing: list [ENTER].

Now reset the computer by pressing [CTRL] **[SHIFT]** and [ESC] keys.

# IF THEN

We will now extend the previous program with the use of the if and then comands.

Type in the following : Notice that we have added two symbols. **<** means less than and is next to the **M** key, **>** means greater than and is next to the **<** (less than) key.

```
5 cls [ENTER]
10 input "what is your name";a$ [ENTER]
20 input "what is your age";age [ENTER]
30 if age < 13 then 60 [ENTER]
40 if age < 20 then 70 [ENTER]
50 if age > 19 then 80 [ENTER]
60 print"So "a$" You are not quite a teenager
   at"age"yearsold":end [ENTER]
70 print "So "a$" you are a teenager at"
   age"yearsold":end [ENTER]
80 print "Oh wet l";a$" You are no longer
   a teenager at"age"years old" [ENTER]
```

To check this program is correct, type:

```
list [ENTER]
```

Now type in :

```
run [ENTER]
```

Now answer the questions prompted by the computer and see what happens.

You can now see what effect the if and then commands have in a program. We have also added the word end at the end of line 60 and 70. The keyword end is used literally to end the running of a program. If end wasn't there in line 60, the program would continue to run, and carry out lines 70 and 80.

Likewise, if *end* wasn't there in line 70, the program would continue to run, and carry out line 80. The colon : before the word *end* separates it from the previous instruction. Colons : can be used to enable you to put two or more instructions on one program line. We have also added line 5 to clear the screen at the start of the program. We will do this from now on in the following programs to make things look neater.

Other keywords associated with I F and THEN include E LS E, 0 R and G 0 T 0. The use of these words in the I F command will become clearer as you work through this guide.

Reset the computer by pressing [CTRL] [SHIFT] and [ESC] keys.

## FOR TO NEXT

We will now show the use of the F 0 R, T 0 and N E X T commands. For this example we will show a way the computer can print the 12 times table (1x12, 2x12, 3x12, etc) Type the following, note that the * symbol means multiply.

```
5 cls [ENTER]
10 for a = 1 to 20 [ENTER]
20 print a"* 12 ="a*12 [ENTER]
30 next a[ENTER]
run [ENTER]
```

You will notice the columns look untidy, so type in:

```
5 cls [ENTER]
10 for a = 1 to 9 [ENTER]
20 print a" * 12 ="a*12 [ENTER]
30 next a [ENTER]
40 for a = 10 to 20 [ENTER]
50 print a"* 12 ="a*12 [ENTER]
60 next a [ENTER]
run [ENTER]
```

Try this program for multiplication tables using other numbers. For example, if you wanted to see the 17 times table, replace the number 12 in lines 20 and 50 with the number 17. Finally, reset the computer using the [CTRL][SHIFT] and [ESC] keys. Note that it is possible to specify the value of each step taken in the FOR TO NEXT command by use of the STEP command. For further information, study the description of F 0 R in Chapter 8.

## SIMPLE ARITHMETIC

Your CPC464 can be used as a calculator quite easily.

To understand this carry out the following examples. We will use the ? symbol instead of typing print in this section. The answer will be printed as soon as the [ENTER] key is pressed.

## ADDITION

(use [SHIFT] and ; keys for plus)

Type:

```
? 3 + 3   [ENTER]
 6
```

(note that you do not type in the equals sign =)

Type:

```
? 8 + 4   [ENTER]
 12
```

## SUBTRACTION

(use unshifted - key for minus)

Type:

```
? 4 - 3   [ENTER]
 1
```

Type:

```
? 8 - 4   [ENTER]
 4
```

## MULTIPLICATION

(Use [SHIFT] and : keys for multiply (* means ×)

Type:

```
? 3 * 3   [ENTER]
 9
```

Type:

```
? 8 * 4   [ENTER]
 32
```

## DIVISION

(use unshifted ? key for divide (/ means ÷)

Type:

```
? 3 / 3   [ENTER]
 1
```

Type:

```
? 8 / 4
 2
```

## SQUARE ROOT

To find the square root of a number type in `sqr( )`. The number you want the square root of should be typed inside the brackets.

Type:

`?sqr(16)` **[ENTER]** (this means $\sqrt{16}$)
4

Type:

`?sqr(100)` **[ENTER]**
*10*

## EXPONENTIATION

(use unshifted **£** for exponentiation)

Exponentiation is when a number is raised to a power of another.
For example 3 squared ($3^2$),3 cubed ($3^3$) etc. Type:

? 3↑3 **[ENTER]** (this means $3^3$)
27

Type:

`?8↑4` **[ENTER]** (this means $8^4$)
4094

## CUBE ROOT

**You** can quite easily calculate cube roots by using a similar method to the last example under square roots.

To find the cube root of 27 ($\sqrt[3]{27}$). Type:

`?27↑(1/3)` **[ENTER]**
**3**

To find the cube root of 125

Type:

`?125↑(1/3)` **[ENTER]**
**5**

## MIXED CALCULATIONS (+,-,*,/,MOD,\)

Mixed calculations are understood by the computer, but they are calculated within certain priorities.

First priority is given to multiplication and division, then addition and subtraction. These priorities apply only to calculations containing only these four operations. The priorities will be further explained later and will include exponentiation etc.

If the calculation was:

```
3+7-2*7/4
```

You may think this would be calculated as

```
3+7-2 *7 /4
=     8  *7 /4
=        56/4
=        14
```

In fact it is calculated as

```
3+7-2*7/4
=3+7-14/4
=3+7-3.5
=10-3.5
=6.5
```

Prove this by typing in this calculation as it is written:

Type:

```
?3+7-2*7/4 [ENTER]
 6.5
```

**You** can change the way the computer calculates this by adding brackets. The computer will deal with the calculation inside brackets prior to the multiplication etc, outside the brackets. Prove this by typing in the calculation including brackets.

Type:

```
?(3+7-2)*7/4 [ENTER]
 14
```

## FURTHER EXPONENTS

If **you** want to use very large or very small numbers in calculations, it is sometimes useful to use scientific notation. The letter E is used for the exponent of numbers to the base 10. You may use either lower case e or upper case **E.**

For example 300 is the same as $3x10^2$. In scientific notation, this is **3E2**. Similarly, 0.03 is the same as $3x10^{-2}$). In scientific notation this is **3E-2**. Try the following examples.

1. 30x10

You can type in:

```
?30*10 [ENTER]
 300
```

**or you can** type:

```
?3e1*1e1 [ENTER]
 300
```

**2. 3000x1000**

Type:

```
?3e3*1e3 [ENTER]
 3000000
```

**3. 3000x0.001**

Type:

```
?3e3*1e-3 [ENTER]
 3
```