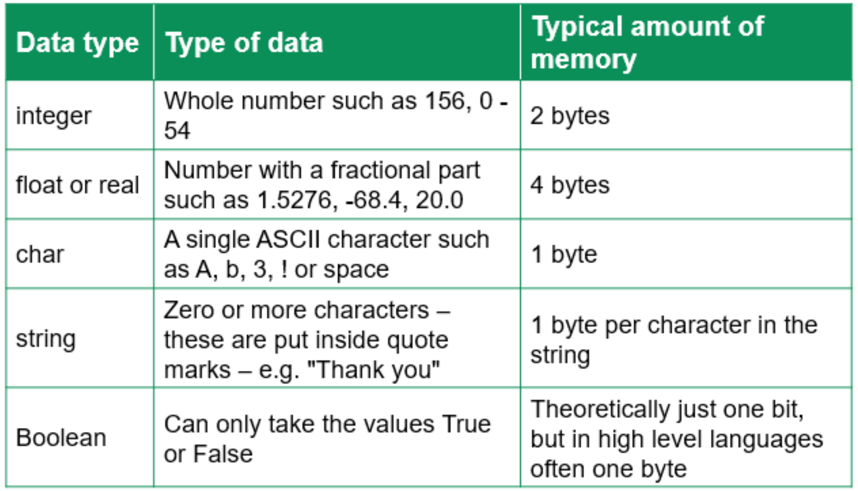
Programming Basics

Keyword	Definition
Variable	A value that can change throughout the program
Constant	A value that stays the same throughout the program and cannot be changed when the program is running
Data type	The kind of values that can be used in a data item
Identifier	A suitable label given to a variable or constant to indicate what it contains
Sequence	The statements are executed one by one in the order they are written
Selection	An IF statement
Iteration	FOR loop, WHILE loop or REPEAT UNTIL loop
Array	A data structure that allows you to hold many items of data
Record	A data structure consisting of a number of fields of different types

Definitions of Data Types



Data type	Type of data	Typical amount of memory
integer	Whole number such as 156, 0 -54	2 bytes
float or real	Number with a fractional part such as 1.5276, -68.4, 20.0	4 bytes
char	A single ASCII character such as A, b, 3, ! or space	1 byte
string	Zero or more characters – these are put inside quote marks – e.g. "Thank you"	1 byte per character in the string
Boolean	Can only take the values True or False	Theoretically just one bit, but in high level languages often one byte

Types of Iteration Statement

WHILE loop – condition controlled, where the condition is tested at the start of the loop.

FOR loop – counter controlled, repeats a specified number of times.

REPEAT .. UNTIL loop – condition controlled, where the condition is tested at the end of the loop, this loop will always be executed at least once.

Iteration using an Array

```
FOR i ← 0 TO LEN(usernames)
    OUTPUT username[i]
ENDFOR
```

Swapping items in an Array

```
names ← ["Sam", "Ron", "Tom", "Bob", "Mo"]
numItems ← LEN[names]
FOR i ← 0 TO numItems - 2
  FOR j ← 0 TO numItems – i – 2
    IF names[j] > names[j + 1] THEN
      temp ← names[j]
      names[j] ← names[j + 1]
      names[j + 1] ← temp
    ENDIF
  ENDFOR
ENDFOR
```

Two dimensional Array

```
FOR s ← 0 TO 2
    studentTotal ← 0
    FOR m ← 0 TO 4
        studentTotal ← studentTotal + mark[s][m]
    ENDFOR
    OUTPUT studentTotal
ENDFOR
```

Programming Basics

Boolean Expressions

Comparison operators	Meaning	Pseudocode example	Result	Notes
=	Equal to	5 = 5	True	Some languages use a double ==
≠	Not equal to	5 ≠ 5	False	Most languages use !=
>	Greater than	5 > 5	False	
≥	Greater than or equal to	5 ≥ 5	True	Most languages use >=
<	Less than	5 < 5	False	
≤	Less than or equal to	5 ≤ 5	True	Most languages use <=

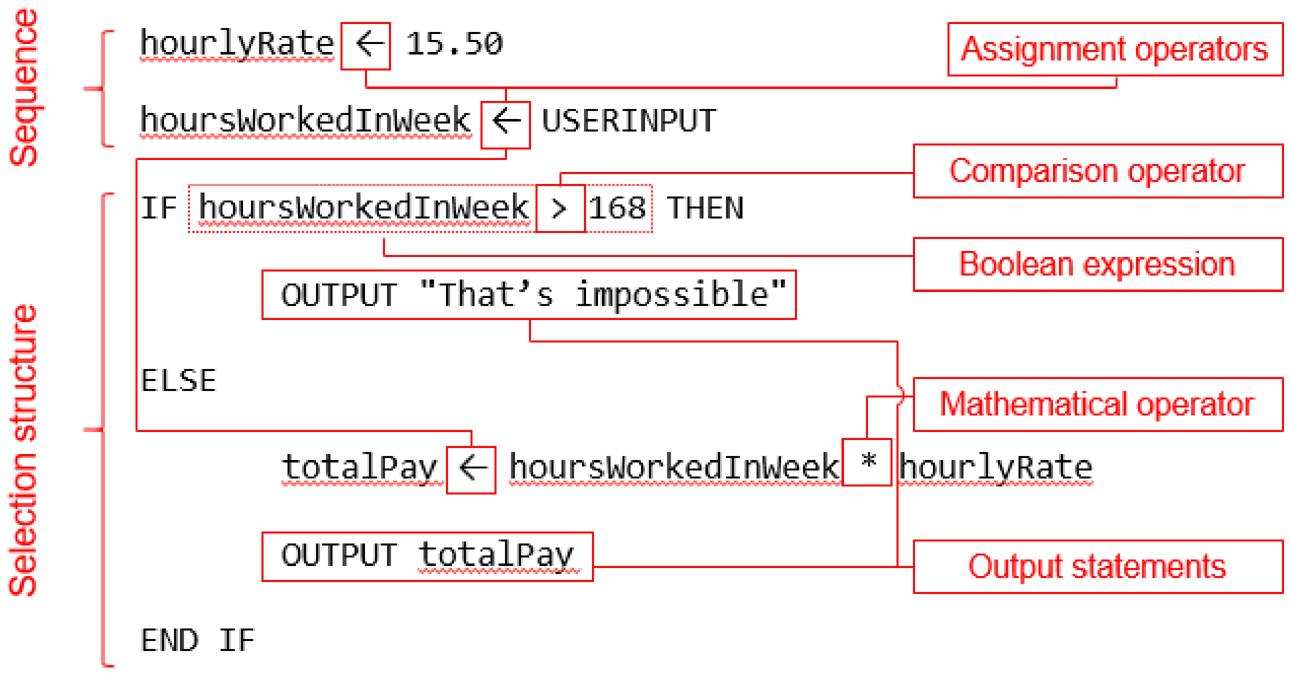
Complex Boolean Expressions

Operator	Description
AND	Returns TRUE if both conditions are TRUE
OR	Returns TRUE if either of the conditions are TRUE
NOT	A TRUE expression becomes FALSE and vice versa

String Handling functions

Function	Example	Result
LEN(str)	word ← "Algorithm" OUTPUT LEN(word)	9
SUBSTRING(start, end, str)	OUTPUT(3,6,word)	"orit"
POSITION(str, char)	POSITION(word, 'r')	4

Sequence and Selection



```
hourlyRate ← 15.50
hoursWorkedInWeek ← USERINPUT
IF hoursWorkedInWeek > 168 THEN
    OUTPUT "That’s impossible"
ELSE
    totalPay ← hoursWorkedInWeek * hourlyRate
    OUTPUT totalPay
END IF
```

Record Structure

```
RECORD film
    filmID : integer
    title : string
    yearReleased : integer
ENDRECORD
```

This record has the elements filmID, title and yearReleased.

Remember - Python does not support records but you do need to know about them.