# Data Types (1.4.1)

| Data Type | Description |
|---|---|
| **Integer** | A whole number |
| **Real/float** | A number with a fractional part |
| **Boolean** | Either True or False |
| **Character** | A letter, number or special character typically represented in ASCII |
| **String** | Anything enclosed in either double or single quotation marks |

Sign and Magnitude: MSB represents the sign of the number (0 if positive, 1 if negative). The rest of the digits are used to represent the number itself. Can give you the wrong answer when using operations. More difficult to implement since the MSB and the rest of the binary number represent data in different data types (character and integer).

Two's Complement: The MSB has a place value of –2^(n-1) (0 if positive, 1 if negative). The rest of the headings have place values from 2^(n-2), 2^(n-3).....2, 1 (same number of possible numbers as normal). Operations can be completed successfully with it. Only one data type used. The range of TC is from –2^(n-1) to 2^(n-1)-1, where n is the number of bits.

The bits can be split into two sections: the part before the binary point (the mantissa) and the part afterwards (exponent). You must specify the number of bits for each part. The part afterwards can have the headings of ½, ¼, 1/8, 1/16.... etc. In this way you can represent some decimal numbers as fixed-point binary numbers.

Issues with fixed-point binary numbers

- Lower range
- Higher accuracy
- Some decimal numbers (such as a 1/3, 1/5, 1/7...), cannot be expressed perfectly

Normalisation: Provides a consistent way of storing floating-point binary numbers so there is only one way of representing any given number.

There is an infinite number of possible ways to represent any given number using floating point binary notation. All positive normalised numbers start with 01. All negative normalised numbers start with 10.

Advantages:

We are storing numbers with the highest possible degree of accuracy.

Disadvantages:

Normalisation provides extra computational steps before the number can be stored. Therefore, the execution of a program will take longer. However, the increase in time is extremely small, therefore, the advantages far outweigh the disadvantages.

To increase range, we must increase the size of the exponent. Since the total number of bits is fixed, the size of the mantissa decreases, decreasing accuracy. Therefore, we have a trade-off between range and accuracy.

| Name of Shift | Logical Left Shift | Logical/Unsigned Right Shift | Arithmetic Right Shift | Circular Right Shift | Circular Left Shift |
|---|---|---|---|---|---|
| **Explanation of how shift Works** | Shift the MSB into the carry bit<br>Move a 0 into the LSB | Shift the LSB into the carry bit. Move a 0 into the MSB | Shift the LSB into the carry bit. If the MSB is a 1, move a 1 into the MSB, else, move a 0 | Shift the carry bit into the MSB, and move the LSB into the carry bit | Shift the carry bit into the LSB, and move the MSB into the carry bit |

| Effect/Use | Multiplying by 2 | Dividing by 2 | Multiplying by 2; can be used on negative numbers | | |
|---|---|---|---|---|---|

| Bitwise Mask | AND | OR | XOR |
|---|---|---|---|
| What the mask does | Used to examine a bit without changing the other bits | Used to set a bit without changing the other bits | Used to toggle a bit without changing the other bits |

Character Set: A defined, standard list of characters recognised by the computer hardware and software, with each character being represented by a single, unique binary number.

Number of possible characters we can store = 2 ** (number of bits for each character)

Every computing device should use the same binary number to represent each character so that different people can use different computers to communicate with each other.

ASCII

- 7 bit (2 ** 7 = 128)
- Covers English + American + some European language characters
- Advantage: Small file size
- Disadvantage: Small Character Set, does not include characters from every written language, nor from every historical script nor emojis

Extended ASCII

- 8 bits (2 ** 8 = 256)
- Covers ASCII + more characters from some foreign languages + some graphic symbols
- Advantage: Small file size
- Disadvantage: Small Character Set, does not include characters from every written language, nor from every historical script nor emojis

Unicode

- Extension of ASCII
- Includes Characters from every written language, Historical Scripts, Emojis
- Advantage: Very large character set
- Disadvantage: Large file size
- 16-bit
- Writing in 16 bits is slow, painful, and prone to error
- Hexadecimal is used instead: 1 nibble = 1 hex digit, 16 binary bits = 4 hex digits

When two files are stored, one with ASCII and one with Unicode, the Unicode file will take up much more space, even when the files look identical.