

**Data Types, Variables and Constants**

**Data Types:**

Data is differentiated into various types. Every type of data element assumes a particular set of values. There are four data types in C language.

|  |  |
| --- | --- |
| Char | A character in the character set |
| Int | A signed or unsigned number having no special character (may have integer values -32767 to + 32767) |
| Float | A signed or unsigned number having decimal point |
| Double | A double precision floating-point number |

The actual storage allocation of each data type depends on the particular computer

The qualifiers short and long can be applied to some of the above data types.

**Variables:**

In C symbolic names called variables are used in place of actual memory locations. Every variable in C has a type and it must be declared before it is used.

**Rules of defining a variable:**

It should begin with a letter or underscore.

It can contain only letters, underscores or digits.

Special characters are not allowed.

A variable name cannot be a keyword.

A variable length should not be more than 31 characters.

**Let’s learn how to declare a variable.**

<data type> <variable name>, . . . .;

Int a;

Defines a memory location of size 4 bytes and of integer data type.

Int a, b, c;

Variables of the same type can be declared in the same line separated by commas

**Constants**

An integer constant is a number that has an integer value

An integer constant can also be specified as decimal, octal and hexadecimal notation.

A Floating-point constant is a number that has a real value, floating point constant is a number written with a decimal point.

1.0 1 .1 0. .0

**A Character constant consists of a single character enclosed within apostrophes**

Such as ‘0’ ‘m’ ‘K’ ‘+’

**String constants (check back slash)**

It consists of zero or more characters enclosed within double quotations marks. A string can be continued by putting a backslash (l) at the end of the line, compiler automatically places a null character (‘l0’) at the end of each string.