

ELEMENTS OF C

Every language has some basic elements & grammatical rules. Before starting with programming, we should be acquainted with the basic elements that build the language.

Character Set

Communicating with a computer involves speaking the language the computer understands. In C, various characters have been given to communicate.

Character set in C consists of;

Types	Character Set
Lower case	a-z
Upper case	A-Z
Digits	0-9
Special Character	!@#\$%^&*
White space	Tab or new lines or space

Keywords

Keywords are the words whose meaning has already been explained to the C compiler. The keywords cannot be used as variable names because if we do so we are trying to assign a new meaning to the keyword, which is not allowed by the computer.

There are only 32 keywords available in C. Below figure gives a list of these keywords for your ready reference.

KEYWORDS

auto	do	goto	signed	unsigned
break	double	if	sizeof	void
case	else	int	static	volatile
char	enum	long	struct	while
const	extern	register	switch	
continue	float	return	typedef	
default	for	short	union	

Identifier

In the programming language C, an identifier is a combination of alphanumeric characters, the first being a letter of the alphabet or an underline, and the remaining being any letter of the alphabet, any numeric digit, or the underline.

Two rules must be kept in mind when naming identifiers.

1. The case of alphabetic characters is significant. Using "INDEX" for a variable is not the same as using "index" and neither of them is the same as using "InDeX" for a variable. All three refer to different variables.
2. As C is defined, up to 32 significant characters can be used and will be considered significant by most compilers. If more than 32 are used, they will be ignored by the compiler.

Data Type

In the C programming language, data types refer to a domain of allowed values & the operations that can be performed on those values. The type of a variable determines how much space it occupies in storage and how the bit pattern stored is interpreted. There are 4 fundamental data types in C, which are- *char*, *int*, *float* & *double*. *Char* is used to store any single character; *int* is used to store any integer value, *float* is used to store any single precision floating point number & *double* is used to store any double precision floating point number. We can use 2 qualifiers with these basic types to get more types.

There are 2 types of qualifiers-

Sign qualifier- signed & unsigned

Size qualifier- short & long

The data types in C can be classified as follows:

Type	Storage size	Value range
char	1 byte	-128 to 127
unsigned char	1 byte	0 to 255
int	2 or 4 bytes	-32,768 to 32,767 or -2,147,483,648 to 2,147,483,647
unsigned int	2 or 4 bytes	0 to 65,535 or 0 to 4,294,967,295
short	2 bytes	-32,768 to 32,767
unsigned short	2 bytes	0 to 65,535

long	4 bytes	-2,147,483,648 to 2,147,483,647
unsigned long	4 bytes	0 to 4,294,967,295

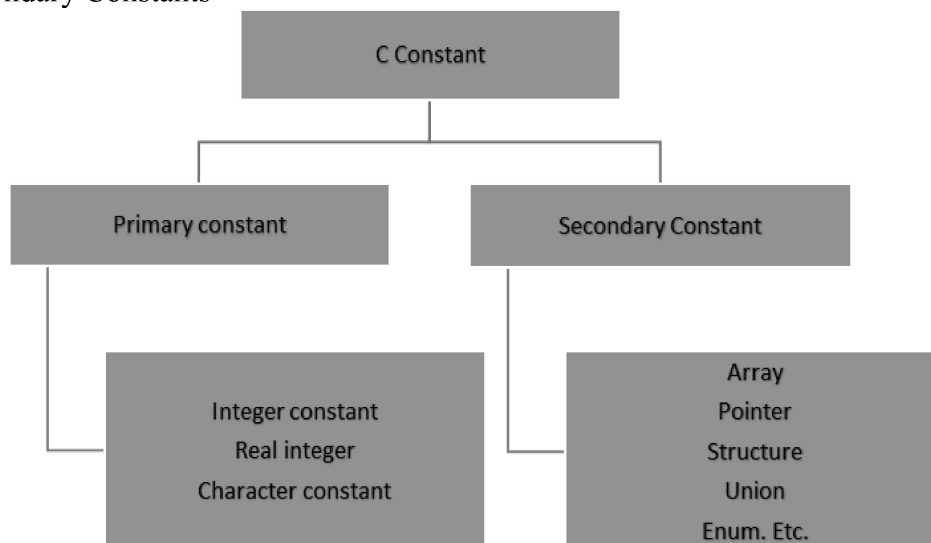
Type	Storage size	Value range	Precision
float	4 bytes	1.2E-38 to 3.4E+38	6 decimal places
double	8 bytes	2.3E-308 to 1.7E+308	15 decimal places
long double	10 bytes	3.4E-4932 to 1.1E+4932	19 decimal places

Constants

A constant is an entity that doesn't change whereas a variable is an entity that may change.

C constants can be divided into two major categories:

- Primary Constants
- Secondary Constants



Here our only focus is on primary constant. For constructing these different types of constants certain rules have been laid down.

Rules for Constructing Integer Constants:

An integer constant must have at least one digit.

- It must not have a decimal point.
- It can be either positive or negative.

- c) If no sign precedes an integer constant it is assumed to be positive.
- d) No commas or blanks are allowed within an integer constant.
- e) The allowable range for integer constants is -32768 to 32767.

Ex.: 426, +782, -8000, -7605

Rules for Constructing Real Constants:

Real constants are often called Floating Point constants. The real constants could be written in two forms—Fractional form and Exponential form.

Rules for constructing real constants expressed in fractional form:

- a) A real constant must have at least one digit.
- b) It must have a decimal point.
- c) It could be either positive or negative.
- d) Default sign is positive.
- e) No commas or blanks are allowed within a real constant.

Ex. +325.34, 426.0, -32.76, -48.5792

Rules for constructing real constants expressed in exponential form:

- a) The mantissa part and the exponential part should be separated by a letter e.
- b) The mantissa part may have a positive or negative sign.
- c) Default sign of mantissa part is positive.
- d) The exponent must have at least one digit, which must be a positive or negative integer.
Default sign is positive.
- e) Range of real constants expressed in exponential form is -3.4×10^{38} to 3.4×10^{38} .

Ex. +3.2e-5, 4.1e8, -0.2e+3, -3.2e-5

Rules for Constructing Character Constants:

- a) A character constant is a single alphabet, a single digit or a single special symbol enclosed within single inverted commas.
- b) The maximum length of a character constant can be 1 character.

Ex.: 'M', '6', '+'