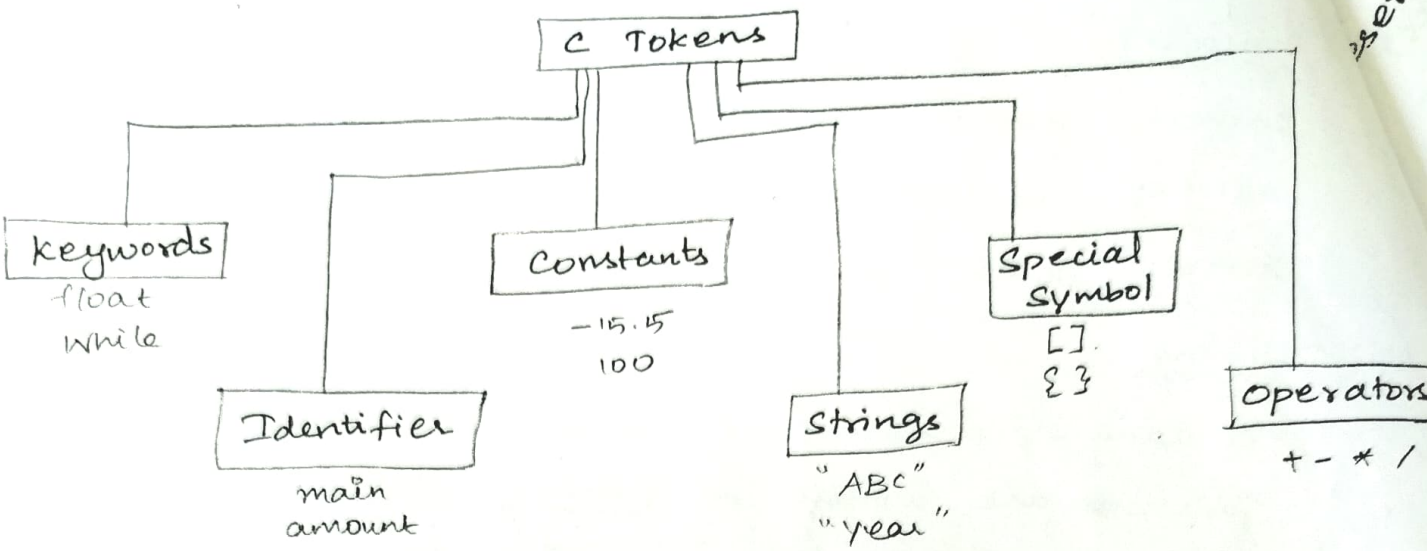



Identifier

* In c language, every word is classified into either a keyword / identifier.

* Names given to various pgm elements such as variables, functions & arrays.

Rules - Identifier

(i) Consists of letters & digits in any order (Considered as letter ↓)

(ii) must begin with a letter / character / Underscore (_)

(iii) Uppercase & lowercase letters are allowed.

(iv) Identifier can be of any length (most 'c' compiler recognizes 1st 31 characters).

(v) No Space & Special Symbols are allowed.

(vi) Identifier cannot be a keyword.

Examples.

STDNAME, TOT_MARKS, _TEMP, Y2K.

Not allowed: IREC, STD NAME.

word.

Reserved words - standard & predefined meaning in 'C' which cannot be changed & they are building blocks for program structure.

Examples.

auto, break, case, default, char, do, float, int, void

Variables

* Variable is an identifier which is used to represent some specific type of information within a particular portion of program.

Variable ← different values at different times during the execution.

Rules for naming the variables.

(i) Variable name - combination of 1to 8 alphabets, digits / underscore.

(ii) first character - Alphabet / Underscore (_).

(iii) Length of variable cannot extend upto 8 characters long, & some can recognize upto 31 characters long.

(iv) No commas / blank spaces allowed within a variable name.

(v) No special symbol, an underscore (_) can be used in variable name.

Variable Declaration.

* After designing variable names, Declare them in pgm & this declaration tells the compiler what variable name & what type of data it can hold.

Syntax: data_type $v_1, v_2, v_3 \ldots v_n$;

data_type is the type of data ←

$v_1, v_2, v_3 \ldots v_n$ — list of variables.

Example: int code; char sex; float price;

char name[10] (array of characters)

Initializing Variables.

* Initialization - Assignment operator (=).
* Initialization can be done while declaring variables.

Syntax: Variable = constant; / datatype variable = const;

Example: int i = 5; char c = 's';

float f = 29.77;

Users-defined variables: C provides a feature to declare a variable of type of user-defined. It allows users to define an identifier that represents existing data type & this can later be used to declare variables.

Type Declaration ⇒ Syntax: typedef data-type identifier;

User defined type declaration.

↳ identifier refers to new name given to the datatype.

Example: typedef int marks;

marks m1, m2, m3;

Enumerated Data type. (User-defined data type) define own datatype where its variable can take value.

Syntax: enum identifier {value1, value2, .... valuen}

User-defined enumerated Datatype

Enumerated Constant

Example: enum day {mon, tue, .... sun}

enum day w-st, w-end;

w-st = mon;

w-end = sun;

enum ... have only one value from constant

ype of Variables.

Availability of variables within the progeam.

Two types of scopes - local & Global

① Local Variables.

Variables defined inside function blk/ compound stmt is called local variables.

Ex:
```
function()
{
    int i, j;      → local variables
    /* body of function */
}
```

② Global / External Variables.

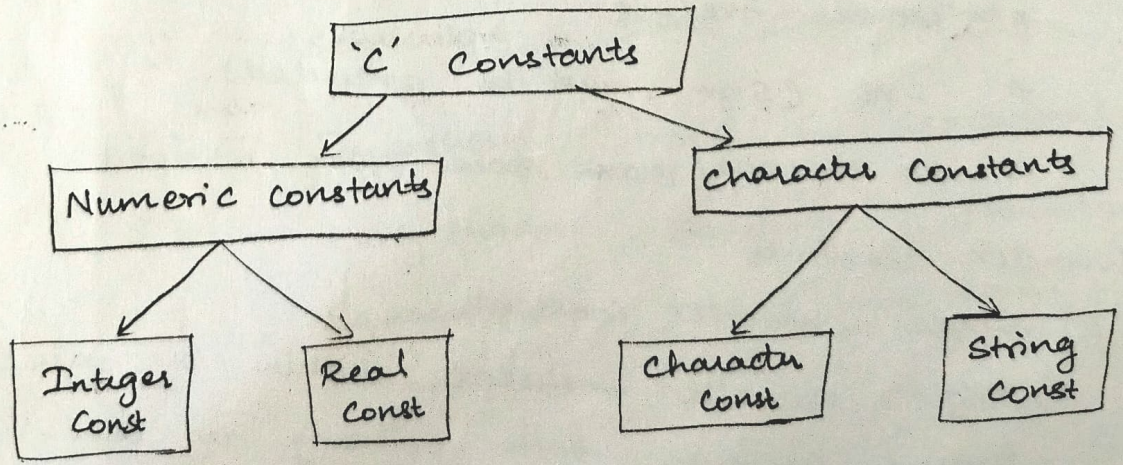
Variables declared before function main(). these variables are available for all functions inside the pgm.

Ex:
```
int a, b = 2;      → Global variables.
main()
{
    ...
    fun();
}

fun()
{
    int sum;
    sum = a+b;
}
```

Constants.

Item whose value cannot be changed during execution is called constants.



Numeric Constants.

① Integer Constant. formed with sequence of digits

3 types: Decimal Number - 0 to 9
(10, 153, -321)
Octal Number - 0 to 7
(052, 0541)
Hexadecimal Number - 0 to 9, A, B, C, D, E, F.
(0x4, 0x8F, 0xBCF)

Ex: marks = 90; discount = 15;

Rules:

* Decimal point is not allowed.
* can either be +ve/-ve. -ve (sign must be preceded) -51
* Must have atleast one digit.
* No commas / Blank spaces are allowed.
* Range (-32,768 to 32,767).

② Real Constants. It is made up of sequence of numeric digits with presence of a decimal point.

To represent quantities that vary continuously such as distance, height, temperature, etc..

Ex: distance = 126.0;
height = 5.6;

Rules:

* Must have one digit
* Must have decimal point
* Either +ve/-ve.
* -ve (sign must be preceded).
* No commas/ Blank spaces are allowed.

Character Constants.

① Single character constants.

Single character enclosed within a pair of single inverted commas both pointing to left.

Ex: 's', 'M', '3'.

(2) String constants.

Sequence of characters enclosed in double quotes, the characters may be letters, numbers, special characters & blank spaces. At the end of string '\0' is automatically placed.

Ex: "HI", "Nuni", "39.77", "50".

(3) Declaring a variable as constant.

When the value of some of the variables may remain constantly during execution of program, in such a situation this is done by using const keyword.

Syntax: const datatype variable = constant

keyword to declare constant

Example: const int dob = 3977;

Delimiters.

Symbols, which has some syntactic meaning & has got significance. Doesn't specify any operation.

| | | |
|---|---|---|
| # | Hash | pre-processor directive |
| , | Comma | seperate list of variables. |
| : | Colon | label Delimiters |
| ; | Semicolon | stmt Delimiters |
| () | Parenthesis | Used in Expression/fn. |
| {} | Curly braces | Blocking 'c' structure |
| [] | Square brats | Used with arrays. |