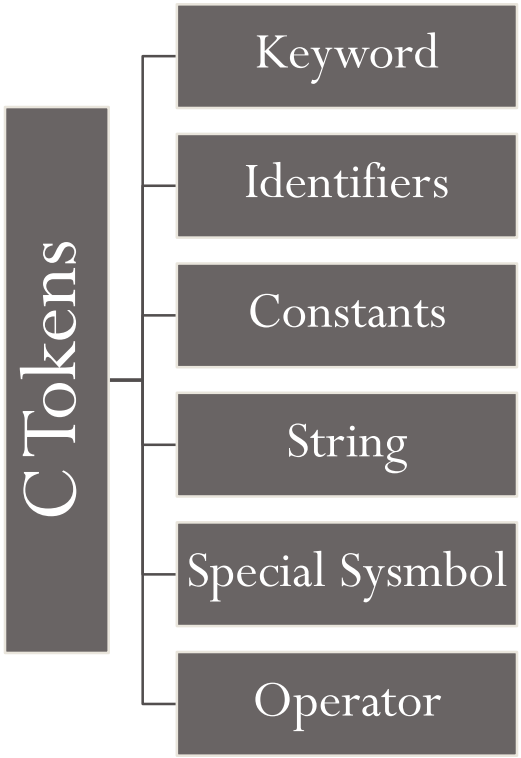
Chapter 2-
Data types, Constants and Variable

C TOKENS

- For understanding C tokens first we have to understand the words or sequence of characters that are used or supported by C. In C programming different character set are used which are the following type:
- Letter: All alphabets. Uppercase A - Z and lowercase a – z.
- Digit: All decimal digits 0…….9
- Special Characters: Comma (,). Semicolon (;), Colon (:), Question mark (?), dollar sign ($), back slash (\), slash (/), exclamatory sign (!), vertical bar (|), Percentage sign (%).
- White Spaces: Blank spaces, New Line, Carriage Return, Form Feed, Horizontal tab.

Types of Tokens

In a text each of the individual words and punctuations marks are called tokens. In C programming each of the smallest individual unit is called token. C tokens are classified in six types as follows:



Keyword

In programming C there is a list of words that have specific meaning and task and the meaning of these words cannot be changed called keyword. Keywords are the building blocks of program instructions. All keywords in C are written in lower case. **Some of the C keywords** are given below:

int	float	double	char
while	for	auto	extern
case	const	default	goto
if	do	static	switch

Identifiers

Identifiers are the name of the variables and sub programs or functions. These are user-defined names means written by users, but there are some rules that must be maintained when writing identifiers. Rules are as follows:

- **First Character must be an character or underscore**
- **Must consist of only letter, digit or underscore.**
- **Only first 31 characters are significant. That means name can be 31 character long maximum.**
- **Cannot use keyword.**
- **Whitespaces are not allowed.**

Constants

In programming C constants refers to the fixed values that can't be changed during the execution of a program. Example: Value of pi 3.1416. The C constants are classified as:

INTEGER CONSTANTS

- Integer constants refer to the numeric constants values without having fractional or exponential values. Integer constants in C can be divided into three types:
- Decimal: Consist of digits 0 to 9 and can be signed + or -. Example: 100, -20, +50 etc.
- Octal: This constants type holds values 0 to 7. Example: 007, 702 etc.
- Hexadecimal: These types of integer constants are rarely used. Values contains 0 to 9 and A to F. Example: 0x7F, 0x6A etc.

Constants

STRING CONSTANTS

- String Constants is sequence of characters enclosed in double quotation. Example: “Hello”, “1971” etc.

ESCAPE SEQUENCES

- In c there are some back slash character constants in output functions those are known as Escape sequences. Example: for output in new line ‘\n’ is added. The lists of characters are:

Constants

Constant	Meaning
'\a'	Audible Alert
'\b'	Back Space
'\f'	Form Feed
'\n'	New Line
'\r'	Carriage Return
'\t'	Horizontal Tab
'\v'	Vertical Tab
'\''	Single Quote
'\"'	Double Quote
'\?'	Question Mark
'\\'	Back slash
'\0'	Null

DATA TYPES

In programming C the basic data types are:

- Integer
- Character
- Float
- Void

Each of them is used in programming C for declaring different types of variable for holding different values. As like for storing price of a fish we have a variable price. But what type of data it will hold we have to define. As price of a fish can be fractional so its data type will be floating. Now we will learn brief about all the types.

DATA TYPES

- **INTEGER TYPE**

Integer types refers to the numeric values without fractional or exponential values In C integers are declared with the keyword **"int".** Size or range of integer data types depends on the computer, it differs depending on word length.

There are some extension of integer for controlling the size and storage space. They are short int, long int, signed int, unsigned int. The size of them will be discussed later.

DATA TYPES

- **FLOATING**

Floating type refers to the numeric values having fractional and exponential numbers with six digit of precision. The floating types are defined with keyword **“float”** in C.

When we need more precession for perfect value than we have use double data types which is an extension of floating type defined with keyword **“double”.**

Double data types have 14 digit precision. For larger value there is another type called long double which defining keyword is **“long double”**with 19 digit of precision.

DATA TYPES

- **CHARACTER**

For declaring single character data character data type is used. Keyword for defining character data type is **“char”.** Character can be signed or unsigned. Unsigned character ranges from 0 to 255 and signed character ranges from -128 to 127.

- **VOID TYPES**

Void types has no values. For defining different functions we need to use this type. If any sub program or function does not return any value then its type is defined as void. The keyword for defining void is **“void”**.

DATA TYPES

The **size of different data types** are given below:

Data Type	Size	Range
char	8 bits	0 – 255 or -128 – 127
unsigned char	8 bits	0 – 255
signed char	8 bits	-128 – 127
int	16 bits	-32768 – 32767
signed int	16 bits	-32768 – 32767
unsigned int	16 bits	0 – 65535
short int	8 bits	0 – 255
long int	32 bits	-2147483648 - 2147483647
float	32 bits	1.2E-38 to 3.4E+38
double	64 bits	2.3E-308 to 1.7E+308
long double	80 bits	3.4E-4932 to 1.1E+4932

VARIABLES

Variable is a container that holds value of a specific type. Variable store values. Actually, the name of the variable is a reference of a memory location for storing data.

The value of a variable can be changed during the execution of a program. Variable name is defined by programmer. Example: Salary, interest etc.

VARIABLES

- There are some rules for writing the name of a variable :
- First digit must be letter or underscore.
- Name length maximum 31 characters long.
- Names are case sensitive. That means uppercase and lower case are significant.
- White spaces are not allowed
- As variable is an identifier, all the rules for identifier also applicable for variables.
- Example: Some **valid variable name**: amount, salary, sum, average etc.
- Some **invalid variable name**: 123, 1name, 2ndplace, %percent etc.

DECLARING VARIABLE

- General format for declaring a variable is: **data-type variable-name;**
- Example: Declaring a integer data named amount: int amount;
- For declaring more than one variable: **data-type variable1,variable2, … ,variableN;**
- Examples: int x,y,z;
- double area;
- float perimeter;

SAMPLE PROGRAM

Sample Program 1: Adding two integer number.

```
/* adding two integers number */
#include <stdio.h>
int main()
{
        int x, z, sum;
        x = 5;
        y = 5;
        sum = x + y;
        printf ("Sum of the Values %d", sum);
}
```

Output of the Program will be:
Sum of the Values 10

SAMPLE PROGRAM

Sample Program 2: Calculating area of a circle

```
/* Calculating area of a circle */
#include <stdio.h>
#define PI 3.1416
int main()
{
        float radius, area;
        radius = 2.5;
        area = PI * radius*radius;
        printf ("Area of the circle is: %f", area);
}
```

- Output of the Program will be:
- Area of the circle is: 19.635