Programming Principle Algorithm(P.P.A)

Unit -1

Complete Detailed Lectures for BCA syllabus

➤ **Introduction about Algorithms & Problem**
➤ **Introduction to 'C' Language & History**
➤ **Functions & Blocks**
➤ **Language Fundamentals**
  ○ Character set
  ○ C Tokens
  ○ Keywords
  ○ Identifiers
  ○ Variables
  ○ Constant
  ○ Data Types
  ○ Comments.

Algorithms: Step by step procedure to solve a problem or develop a program.

Properties of Algorithms:

➢ **Finiteness:** Must terminate in a finite number of steps.
➢ **Definiteness:** Algorithms must be precisely stated.
➢ **Effectiveness:** Algorithms must be easy to convertible into programming language.
➢ **Input/Output:** Algorithms must take at least some input to produce desired output.
➢ **Completeness:** The algorithms must be complete in itself in order to solve the given problem.

Computational Problem: A computational problem is a problem that can be solved step-by-step with algorithms.

Components of Algorithms & Computational Problem

➤ **Programming**: Programming is the art of developing a solution to a computational problem
➤ **Code**: Instruction that we use to develop the algorithm.
➤ **Programming language**: The language in which the code is written is known as programming language.

Properties of Computational Problems: Input & Output

Examples(Sorting): Input Random String of P1,P2,P3...... P'n-1

: Output P1≤ P2 ≤P3 ....P'n-1

Introduction about C Language

➤ **Developed BY:** AT&T's Bell Laboratories of USA.
➤ **Foundation Year:** 1972
➤ **Writer:** Dennis Ritchie

Importance of C language :

➤ **Simple,Fast and Efficient** : The basic syntax style of implementing C language is very simple and easy to learn.

➤ **Portability:** C programs are machine-independent which means that you can run the fraction of a code created in C on various machines.

➤ **Extensibility:** We can add new features in existing codes or program.

➤ **Function-Rich Libraries:** C comes with an extensive set of libraries with several built-in functions that make the life of a programmer easy.

➤ **Dynamic Memory Management:** We can utilize and manage the size of the data structure in C during runtime

➤ **Modularity With Structured Language:** We can divide the code or program in different modules.

Functions in C language: Function is a set of statements or block of codes which only runs when it is called.
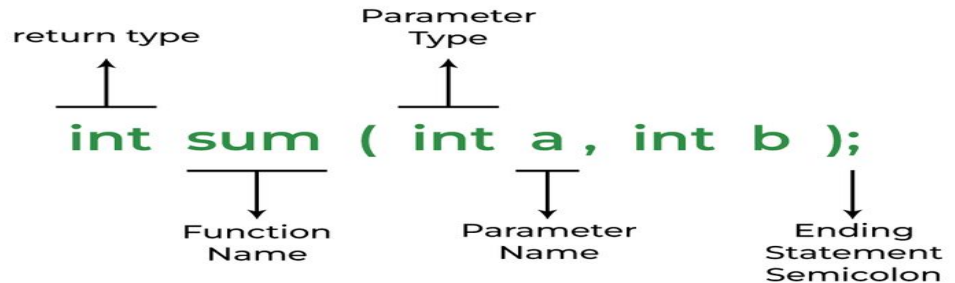
Type of functions in C language :

➤ **Used defined function:** We can design own function with set of logics.

➤ **Library function:** These are the predefined functions defined in system library.

Syntax of Functions in C : The syntax of function can be divided into 3 aspects:

1. Declaration 2. Definition 3. Calls

```
int sum ( int a , int b );
```

return type

Parameter Type

Function Name

Parameter Name

Ending Statement Semicolon

Block in C language: Set of statements written within the right and left braces.
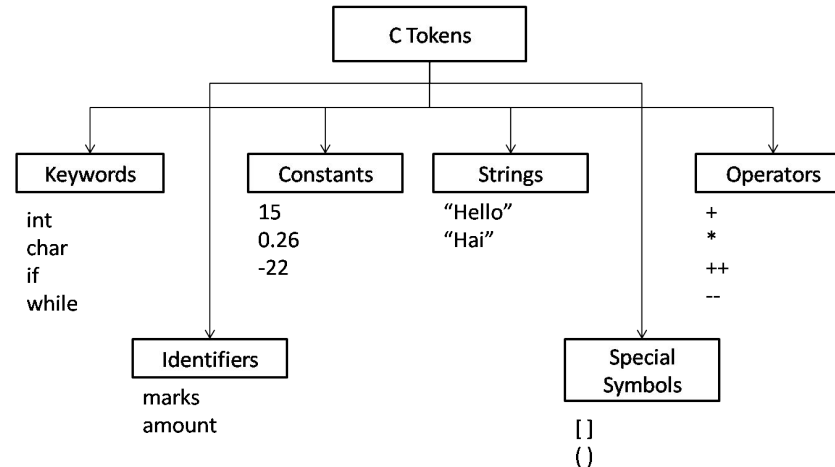
Example:

Directory for preprocessors	⟷

```c
#include<stdio.h>
void main() {
int a = 5;
printf("\n%d", a);{
int a = 2;
printf("\n%d", a);
} }
```

➤ **Character set** : Character used from from A to Z to number 0,9 special charter(+ − * / = % & # ! ? ^ " ' / | < > ( ) [ ] { } : ; . , ~ @ !)

➤ **C Tokens:** Combination of variables,operators, symbols etc that is having meaningful to the functioning of a compiler.

➤ **Keywords**: Predefined by system program that has its own property and behaviour.

Example: ⟶

```
int, main, void
```

➤ **Identifiers:** Identifiers are names given to various program elements such as variables, functions, and arrays

Examples: ⟶

```
int sum, average, A[10];
```

➤ **Variables**: Variables are containers for storing data values, like numbers and characters.
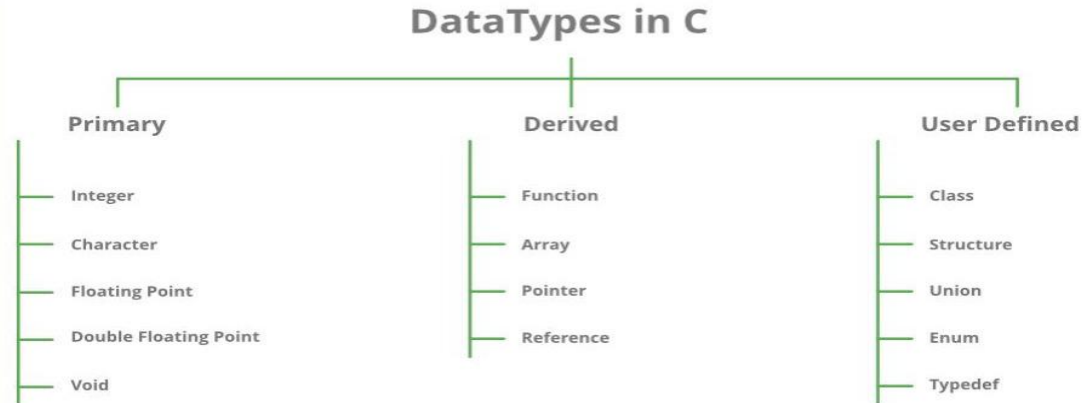
Example: int a=9;

CodeAspire

➢ **Constants**: A constant is an identifier whose value remains unchanged throughout the program

Syntax: ➡️

```
const datatype varname = value;
```

➢ **Data Types**: Type of data that variable can store like integer, character, floating, double, etc known as data type.

DataTypes in C

Primary	Derived	User Defined
Integer	Function	Class
Character	Array	Structure
Floating Point	Pointer	Union
Double Floating Point	Reference	Enum
Void		Typedef

CodeAspire

Comments: It is generally used to show the information about the program.

Types of Comments:

➢ **Single-line Comment in C**: A single-line comment in C starts with (//) double forward slash

   Example: "// Statement......"

➢ **Multi-line Comment in C:** The Multi-line comment in C starts with a forward slash and asterisk (/*) and ends with an asterisk and forward slash (*/)

   Example: "/* Statement */"

Thanks for Watching!
Stay Connected for upcoming videos

Ashish K. Verma
BRAND MANAGER

🌐 www.codeaspire.in

📞 0120-4786498

✉ info@codeaspire.in

LIKE
SUBSCRIBE
SHARE

For any enquiry, connect us on call or whatsapp +91-7289058852