**CPP BASIC CONCEPTS:-**

**Computer language : Used for communicate with computer.**

**Types of coputer language :**

1. **Machine level language *: language present in 0 and 1 forms.***

1. **Assembly language : *Human readable command such as add ,mov , add ,sub.***
2. **High level language : *Language like English language.***

**Object oriented language :**

**Language depend on object.**

**Characteristics:**

* **Top down approach**
* **Focus on function,data**
* **Divded into small block**
* **Data transfer easy.**

**Data members : member variable in the object**

**Method (Function) : set of instruction**

**Data encapsulation : Combining data member and member function together in one unit .**

**Data abstraction :**

**Hiding the implementation details from user.**

**Inheritance :**

**Accessing the properties of another class**

**Polymorphism :**

**Same function have different behaviour data type.**

**Const keyword : constant variable does not change .**

**Data types : specify the type of value.**

1. **Primitive Data Types :**

* **Primitive data types - includes byte, short, int, long, float, double, boolean and char**
* **Primitive types are predefined (already defined)**

1. **Non-Primitive Data Types:**

* **Non-primitive data types - such as**[**String**](https://www.w3schools.com/java/java_strings.asp)**,**[**Arrays**](https://www.w3schools.com/java/java_arrays.asp) **,** [**Structures**](https://www.geeksforgeeks.org/difference-c-structures-c-structures/) **,** [**Enumeration**](https://www.geeksforgeeks.org/enumeration-enum-c/) **,** [**Pointers**](https://www.geeksforgeeks.org/pointers-in-c-and-c-set-1-introduction-arithmetic-and-array/)**,**[**Union**](https://www.geeksforgeeks.org/union-c/)**and**[**Classes**](https://www.w3schools.com/java/java_classes.asp)**.**

***Structure :- It is user defined data type used to store elements of different data types together.***

***Struct keyword is used to create structure.***

***Operators :***

* ***Arithmetic Operators :* (+ , - , \* , / , % , ++ , -- )**
* ***Relational Operators :* (== , != , > , < , >= , <= )**
* ***Logical Operators :* ( && , || , ! )**
* ***Bitwise Operators :* ( & , | , ^ , ~ , << , >>)**
* ***Assignment Operators :* ( = , += …………….)**

**Control structure :**

1. **If statement**
2. **Switch statement**
3. **For loop**
4. **While loop**
5. **Do while loop**
6. **Break and continue**

**Access specifier :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Access specifier** | **Own class** | **Derived class** | **From object** |
| **Private** | **Yes** | **No** | **No** |
| **Protected** | **Yes** | **Yes** | **No** |
| **Public** | **Yes** | **Yes** | **Yes** |

***Inline function :- It is a function that replaces function call with actual code of function.***

***It cannot be used in case if the code in function is big.***

***All function cannot be inline.***

**Namespace :- It is used to define scope in the program .There are two types of scope in the program**

**1)Global scope : *It is applicable for entire program.and not bound to block of code.***

**2)Local Scope *: It is applicable to function or block of code.***

***Constructor : It is a special member function gets automatically called during creation of object of the class .***

***Syntax: function name()***

***There are different types of constructor:***

***1)Copy Constructor***

***2)Parameterized Constructor***

***Destructor :- It is used to destroy objects in c++***

***It is used for deallocation of memory at the time of termination of program..***

***Syntax : ~function()***

***Member functions :- These are the function present inside the class.***

***Data Member :- These are the variables present inside the class.***

***Class :- It is a user defined data type which has data members and member functions inside it.***

***It is blueprint for creating objects.***

***Object :- It is a instance of a class which is used to access the data member and member function of class.***

***Function :- It is a block of code performing a specific task.***

***It is used for reducing the complexity of program.***

***It is used for promoting the code reusability of code.***

***Call by value :- Called functions create copy of variables and copy the value passed to it.***

***Function does not have access to original value.***

***If we want to modify the value of variable then call by value is not sufficient.***

***Call by reference :- We pass the address of variable instead of value so any changes in variable will reflect.***

[***Friend function***](https://www.geeksforgeeks.org/friend-class-function-cpp/)***: A friend class can access private and protected members of other class in which it is declared as friend.***

***The function which is declared inside the class it is not a member of class but can access the private and protected members of the class.***

***Link :-*** [***https://www.geeksforgeeks.org/friend-class-function-cpp/***](https://www.geeksforgeeks.org/friend-class-function-cpp/)

***Virtual Function :- If the same function is present in base class and derived class . Then to resolve that ambiguity of which function to call virtual function is used.***

***We create pointer in base class which is used to access the function in derived class***

***We use virtual keyword to function present in base class to resolve ambiguity.***

***Pointer :- It is a derived data type . It stores address of the another variable . It is used in dynamic memory allocation .***

***Languages that support pointer :- c++ , c***

***Languages that support pointer :- python , java***

***Different types of pointer in c++.***

***1)Null pointer :- The pointer does not point to any memory location i.e- stores null value***

***(int \*ptr=NULL;).***

***2)Void pointer :- Void pointer can hold address of any data type . This pointer is not associated with any data type(void \*ptr)***

***3)Wild pointer :- Uninitialized pointers are known as wild pointers because they point to some arbitrary memory location and may cause a program to crash or behave badly.***

***4)Dangling pointer :- A pointer pointing to the memory location that has been deleted.***

***5)This pointer :-***

***Link:-*** [***https://www.geeksforgeeks.org/dangling-void-null-wild-pointers/***](https://www.geeksforgeeks.org/dangling-void-null-wild-pointers/)

***Static data member :-*** ***Static variables in a Function: When a variable is declared as static, space for it gets allocated for the lifetime of the program. Even if the function is called multiple times, space for the static variable is allocated only once and the value of variable in the previous call gets carried through the next function call.***

***Link :-*** [***https://www.geeksforgeeks.org/static-keyword-cpp/***](https://www.geeksforgeeks.org/static-keyword-cpp/)

***Static member function :- Static member functions are allowed to access only the static data members or other static member functions***

***Static functions in a class: Just like the static data members or static variables inside the class, static member functions also does not depend on object of class.***

***To invoke the static members we use the class name and the scope resolution operator.***

***Link :-*** [***https://www.geeksforgeeks.org/static-keyword-cpp/***](https://www.geeksforgeeks.org/static-keyword-cpp/)

***Object-oriented programming – As the name suggests uses objects in programming.***

***It provides:-***

***Class***

***Objects***

***Encapsulation***

***Abstraction***

***Polymorphism***

***Inheritance***

***Dynamic Binding***

***Message Passing***