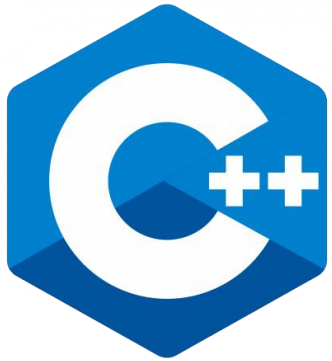


C++ Programming

Trainer : Pradnyaa S. Dindorkar

Email: pradnya@sunbeaminfo.com

We did…..

1. namespace
2. cin and cout
3. complex class
4. Modular Approach
5. Constant

1. References
2. Difference between Pointers and reference
3. Sum function and Copy Constructor
4. New and delete
5. Difference between New and malloc
6. Shallow Copy and deep copy
7. static
8. Friend function

Reference

- Reference is derived data type.
- It alias or another name given to the existing memory location / object.
  - Example : int a=10; int &r = a;
  - In above example a is referent variable and r is reference variable.
  - It is mandatory to initialize reference.
- Reference is alias to a variable and cannot be reinitialized to other variable
- When ‘&’ operator is used with reference, it gives address of variable to which it refers.
- Reference can be used as data member of any class

Reference

- We can not create reference to constant value.
  - int &num2 = 10; //can not create reference to constant value
- Reference is internally considered as constant pointer hence referent of reference must be variable/object.

```
int main( void )
{
        int num1 = 10;
        int &num2 = num1;
        cout<<"Num2 : "<<num2<<endl;
        return 0;
}
```

pass arguments to function, by value, by address or by reference.

- **In C++, we can pass argument to the function using 3 ways:**

1. By Value
2. By Address
3. By Reference

- If variable is passed by reference, then any change made in variable within function is reflected in caller function.
- Reference can be argument or return type of any function

Difference between Pointers and reference

Pointer

- It is a variable that points to another variable.
- To access the value of a variable with the help of a pointer, we need to do dereferencing explicitly.
- We can create a pointer to pointer
- We can create a pointer without initialization. Create a NULL pointer.

Eg int n=5; int* ptr=&n;

Reference

- It is an alias / secondary name to an already existing memory.
- No need of dereferencing to access a value of a variable with ref.
- We can't create a reference to reference.
- We can't create a ref without initialization NULL ref can't be created.

Eg : int n=5; int& ref=n;

Sum function and Copy Constructor

- Copy constructor is a single parameter constructor hence it is considered as parameterized constructor
- Example: sum of two complex number

  Complex sum(const Complex &c2)

Copy constructor

Complex c2(c1)

or

Complex c2=c1

C1 → 7 + j 6

C2 → 7 + j 6

Write a function in complex class to add 2 complex numbers

C1 → 7+j6

\+

C2 → 3+j2

C3 → 10+j8

Dynamic Memory Allocation

To allocate memory dynamically then we should use **new** operator

To deallocate that memory we should use **delete** operator.

Dangling pointer :- The pointer which contains, address of deallocated memory.

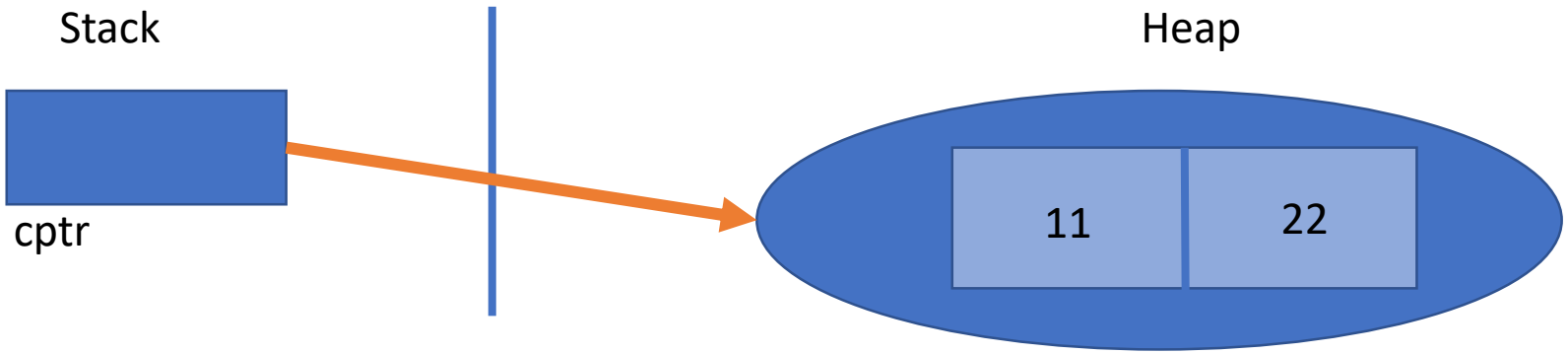
Memory leakage :- When we allocate space in memory, and if we loose pointer to reach to that memory then such wastage of memory is called memory leakage.

```
int main()
{
        int *ptr = new int;             // allocate memory
        *ptr = 125;                     //Dereferencing
        cout<<"Value:"<<*ptr<<endl;  //Dereferencing
        delete ptr;                     // deallocate memory
        ptr = NULL;
        return 0;
}
```

Heap Based object

```
Complex *cptr=new Complex (11,22) ;
delete cptr;
```

- By using new we are allocating dynamic memory for complex class object .
- Object get created on heap section hence this object is call Heap based or dynamic object.
- Cptr is complex type pointer which is holding the address of that dynamic object.



Difference between malloc and new

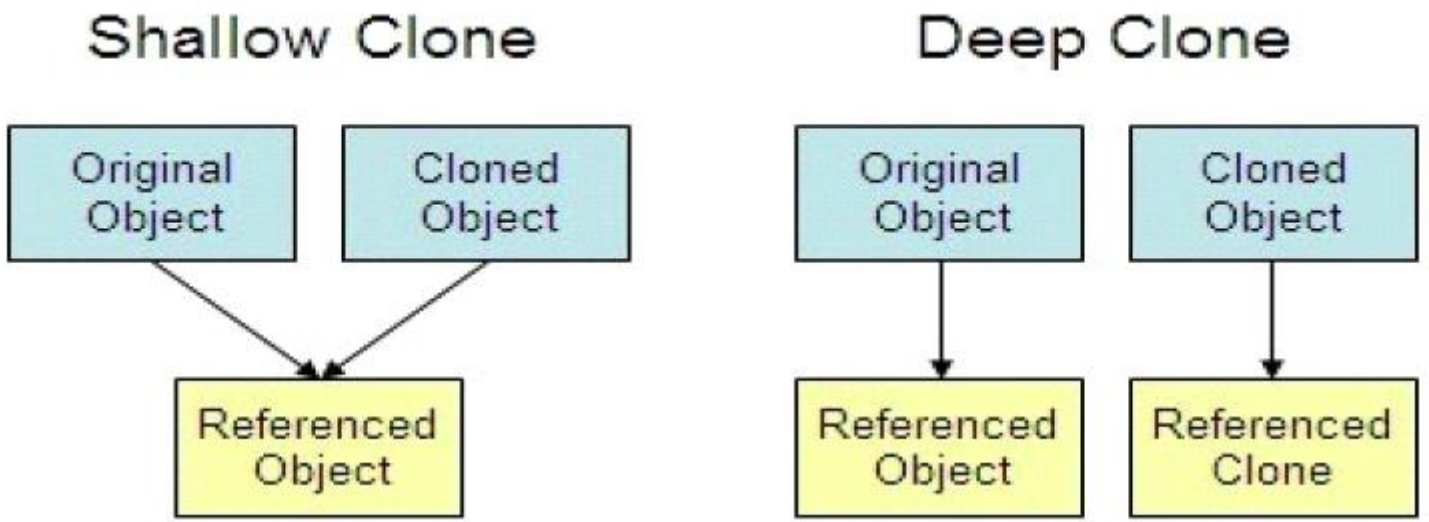
new	malloc
new is an operator.	malloc is a function.
new returns a typecasted operator, so no need to do explicit typecasting.	malloc returns void pointer,malloc returns void pointer,cast it explicitly, before use.
We must mention the datatype while allocating the memory with new.	malloc accepts only the exact no. of bytes required, so no need to mention datatype.
When memory is allocated with new, constructor gets called for the object.	When memory gets allocated by malloc function, constructor function does not gets called.

Difference between malloc and new

new	malloc
Memory allocated by new is released by the operator delete.	Memory allocated by malloc is released by function free.
Destructor is called when memory is released with delete.	Destructor is NOT called when the memory is released with free.
To release memory for array syntax is delete[ ]	To release memory for array syntax is free(ptr);
In case new fails to allocate memory, it raises a Run time exception called as bad_alloc.	In case malloc fails to allocate memory it returns a NULL.

Object Copying

- In object-oriented programming, “object copying” is a process of creating a copy of an existing object.
- The resulting object is called an object copy or simply copy of the original object.
- Methods of copying:
  - Shallow copy
  - Deep copy



Types of Copy

The copy constructor is used to initialize the new object with the previously created object of the same class.

- **Shallow Copy**
  - The process of copying state of object into another object.
  - It is also called as **bit-wise** copy.
  - When we assign one object to another object at that time copying all the contents from source object to destination object as it is. Such type of copy is called as shallow copy.
  - Compiler by default create a shallow copy. Default copy constructor always create shallow copy.

- **Deep Copy**
  - Deep copy is the process of copying state of the object by modifying some state.
  - It is also called as **member-wise** copy.
  - When class contains at least one data member of pointer type, and class contains user defined destructor then when we assign one object to another object then copy constructor is called.
  - At that time instead of copy base address allocate a new memory for newly created object and then copy contain from memory of source object into memory of destination object. Such type of copy is called as deep copy.

Thank You