Object Oriented Programming using C++

Pointer and Arrays

Pointer

- Pointers are variables that store the memory addresses of other variables of same data type.

Ex:

If we have a variable ***var*** in our program, ***&var*** will give us its address in the memory.

Var=10

1000

```
#include <iostream>
using namespace std;
int main()
{
    // declare variables
    int var1 = 3;
   int var2 = 24;
    int var3 = 17;
cout << "Address of var1: "<< &var1 << endl; // print address of var1
   cout << "Address of var2: " << &var2 << endl; // print address of var2
     cout << "Address of var3: " << &var3 << endl; // print address of var3
}
```

O/P:

Address of var1: 0x7fff5fbff8ac

Address of var2: 0x7fff5fbff8a8

Address of var3: 0x7fff5fbff8a4

- *Here, 0x at the beginning represents the address is in the hexadecimal form.*
- *Notice that the first address differs from the second by 4 bytes and the second address differs from the third by 4 bytes. This is because the size of an int variable is 4 bytes in a 64-bit system.*

Declaration of pointer

Syntax :

*data_type * pointer_name;*

Ex:

```
int var,*p;
P=&var;
```

```
#include <iostream.h>
using namespace std;
int main()
{
int var1 = 3;
    int var2 = 24;
    int var3 = 17;
int *p1,*p2,*p3;
p1=&var1;
P2=&var2;
P3=&var3
cout << "Address of var1: "<< p1 << endl;
 cout << "Address of var2: " << p2 << endl;
  cout << "Address of var3: " << p3 << endl;
}
```

O/P:

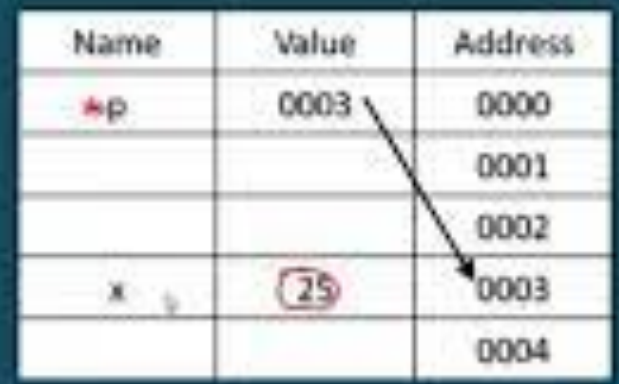
```
Address of var1: 0x7fff5fbff8ac
Address of var2: 0x7fff5fbff8a8
Address of var3: 0x7fff5fbff8a4
```

Note: You may not get the same results when you run the

Using the * Operator

The Contents of or Dereference operator allows us to get the value stored at the address held by the pointer.

```
int x = 25;
int* p = &x;
cout << *p << endl;
25
```



X=25
&x=0003
P=0003
*p=25
&p=0000

Applications of Pointer

- Working on the original variable.
- With the help of pointers, we can create data structures (linked-list, stack, queue).
- Returning more than one values from functions.
- Searching and sorting large data very easily.
- Dynamically memory allocation.
- Less time in program execution.

How to access value using Pointers?