

Object Oriented Programming

Pointers

Mr. Usman Wajid

usman.wajid@nu.edu.pk



National University
of Computer & Emerging Sciences

What is a Pointer?

Pointer

It is special type of variable that holds the address of a conventional (simple) variable as a value

- **Basic Syntax:**

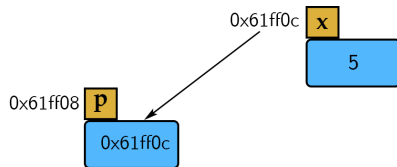
```
<data-type> * <identifier> = &<identifier>;
```

Integer Pointer Example:

```
int main() {  
  
    int x = 5;  
    int * p = &x;  
  
    cout<<"value of x: "<<x<<endl;  
    cout<<"Address of x: "<<&x<<endl;  
    cout<<"value of p: "<<p<<endl;  
    cout<<"Address of p: "<<&p<<endl;  
    cout<<"value of *p: "<<*p<<endl;  
}
```

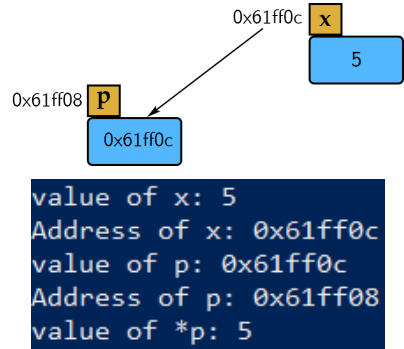
Integer Pointer Example:

```
int main() {  
  
    int x = 5;  
    int * p = &x;  
  
    cout<<"value of x: "<<x<<endl;  
    cout<<"Address of x: "<<&x<<endl;  
    cout<<"value of p: "<<p<<endl;  
    cout<<"Address of p: "<<&p<<endl;  
    cout<<"value of *p: "<<*p<<endl;  
}
```



Integer Pointer Example:

```
int main() {  
  
    int x = 5;  
    int * p = &x;  
  
    cout<<"value of x: "<<x<<endl;  
    cout<<"Address of x: "<<&x<<endl;  
    cout<<"value of p: "<<p<<endl;  
    cout<<"Address of p: "<<&p<<endl;  
    cout<<"value of *p: "<<*p<<endl;  
}
```



Char Pointer Example:

```
int main() {  
  
    char c = 'a';  
    char * p = &c;  
  
    cout<<"value of c: "<<c<<endl;  
    cout<<"Size of c: "<<sizeof(c)<<endl;  
    cout<<"value of *p: "<<*p<<endl;  
    cout<<"Size of p: "<<sizeof(p)<<endl;  
}
```

Char Pointer Example:

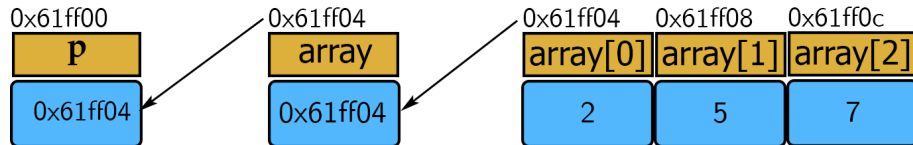
```
int main() {  
  
    char c = 'a';  
    char * p = &c;  
  
    cout<<"value of c: "<<c<<endl;  
    cout<<"Size of c: "<<sizeof(c)<<endl;  
    cout<<"value of *p: "<<*p<<endl;  
    cout<<"Size of p: "<<sizeof(p)<<endl;  
}
```

```
value of c: a  
Size of c: 1  
value of *p: a  
Size of p: 4
```

Pointer Vs Array

An Array name without indices

It points to the address of the first index of the array



Pointer Vs Array Example:

```
int main() {  
    int array[3] = {2, 5, 7};  
    int * p;  
    cout<<"&array = "<<&array<<endl;  
    cout<<"array = "<<array<<endl;  
    cout<<"&array[0] = "<<&array[0]<<endl;  
    cout<<"array[0] = "<<array[0]<<endl;  
    cout<<"&array[1] = "<<&array[1]<<endl;  
    cout<<"array[1] = "<<array[1]<<endl;  
    cout<<"&array[2] = "<<&array[2]<<endl;  
    cout<<"array[2] = "<<array[2]<<endl;  
    cout<<"\np = array"<<endl; p = array;  
    cout<<"\np = "<<p<<endl;  
    cout<<"*p = "<<*p<<endl;  
    cout<<"\np++; "<<endl; p++;  
    cout<<"p = "<<p<<endl;  
    cout<<"*p = "<<*p<<endl;  
    cout<<"\np++; "<<endl; p++;  
    cout<<"p = "<<p<<endl;  
    cout<<"*p = "<<*p<<endl;  
}
```

```
&array = 0x61ff04  
array = 0x61ff04  
&array[0] = 0x61ff04  
array[0] = 2  
&array[1] = 0x61ff08  
array[1] = 5  
&array[2] = 0x61ff0c  
array[2] = 7
```

```
p = array  
&p = 0x61ff00
```

```
p = 0x61ff04  
*p = 2
```

```
p++  
p = 0x61ff08  
*p = 5
```

```
p++  
p = 0x61ff0c  
*p = 7
```

The `new` Keyword

The `new` Keyword

It is used to store a value in the memory that can be accessed using a pointer only. It is also used to allocate the size an array size dynamically during the run time.

① Single Pointer Syntax:

```
<data-type> * <identifier> = new <data-type>;
```

② Array Pointer Syntax:

```
<data-type> * <identifier> = new <data-type>[<array-size>];
```

The new Keyword Example 1

```
int main() {  
    int * x = new int;  
  
    *x = 1 ;  
  
    cout<<"\n&x = "<<&x<<endl;  
    cout<<"x = "<<x<<endl;  
    cout<<"*x = "<<*x<<endl;  
}
```

```
&x = 0x61ff08  
x = 0x1096d60  
*x = 1
```

The new Keyword Example 2

```
int main() {  
    int * array = new int[3];  
  
    array[0] = 2;  
    array[1] = 5;  
    array[2] = 7;  
  
    cout<<"&array = "<<&array<<endl;  
    cout<<"array = "<<array<<endl;  
    cout<<"&array[0] = "<<&array[0]<<endl;  
    cout<<"array[0] = "<<array[0]<<endl;  
    cout<<"&array[1] = "<<&array[1]<<endl;  
    cout<<"array[1] = "<<array[1]<<endl;  
    cout<<"&array[2] = "<<&array[2]<<endl;  
    cout<<"array[2] = "<<array[2]<<endl;  
}
```

```
&array = 0x61ff0c  
array = 0xff6d48  
&array[0] = 0xff6d48  
array[0] = 2  
&array[1] = 0xff6d4c  
array[1] = 5  
&array[2] = 0xff6d50  
array[2] = 7
```

The new Keyword Example 3

```
int main() {  
    int * array = new int;  
  
    *array = 2;  
    cout<<"array = "<<array<<endl;  
    cout<<"*array = "<<*array<<endl;  
  
    array++;  
    *array = 5;  
    cout<<"array = "<<array<<endl;  
    cout<<"*array = "<<*array<<endl;  
  
    array++;  
    *array = 7;  
    cout<<"array = "<<array<<endl;  
    cout<<"*array = "<<*array<<endl;  
}
```

```
array = 0x12c6d48  
*array = 2  
array = 0x12c6d4c  
*array = 5  
array = 0x12c6d50  
*array = 7
```