

Contents

Pointer More Insights	1
Double Pointers - Integer array of array	1

Pointer More Insights

Double Pointers - Integer array of array

1. Compare the behavior of pointers for complete objects with that of pointers for primitive data types.
2. Gain insight into how this concept mirrors the address or reference of an entire string during our discussions..

```
int cstyle_array[2][3] = {{100,200,300}, {400,500,600}};
```

Decimal	Hex
100	0x64
200	0xC8
300	0x12C
400	0x190
500	0x1F4
600	0x258

----- IN Decimal -----

```
+-----+
| [100] || [200] || [300] |
+-----+
| [400] || [500] || [600] |
+-----+
```

----- IN Hexadecimal -----

```

+----- SECOND LEVEL POINTEER (*cstyle_array) - length 4-bytes
|
V
SECOND LEVEL
(**cstyle_array) +-----+
length 12-bytes --> | [64]-[00]-[00]-[00] || [C8]-[00]-[00]-[00] || [2C]-[01]-[00]-[00] |
|0x16f0b9fc0|      +-----+
TOP LEVEL          | [90]-[01]-[00]-[00] || [F4]-[01]-[00]-[00] || [58]-[02]-[00]-[00] |
+-----+
```

1. We obtain a pointer `cstyle_array` `-->` `(0x16f0b9fc0)` that refers to the first element in the array. This pointer knows that if we increment it, it will point to the second element in the array, which is the value ``400``.

2. Dereferencing the double pointer:

Double dereferencing it will give us the first value which is `100`

```
*(my_array) --> `100`
```

3. Increment the double reference

```
**(&cstyle_array + 1) -> 400
```

- This one will increment the double pointer by 1-cell, at top level, means when we dereferencing this double pointer it will give us the second value in top level (rows)

4. Here, we will convert the double pointer to a single pointer with a size of 1 byte, which is of type char. We will then increment it by 4 bytes. After that, we will cast it again to an integer with a size of 4 bytes and dereference it.

```
*((int*)((char*)&cstyle_array + 4)) -> 200
```

5. Here, we obtain the second level pointer by first dereferencing the double pointer and then casting it to a char with a size of 1 byte. We increment it by 4 bytes. After that, we cast it back to an integer pointer and finally dereference it.

```
*(int*)((char*)(*cstyle_array) + 4) -> 200
```

6. Here, we obtain the first level pointer by dereferencing our double pointer `cstyle_array`, then we increment it by 1 byte and its size is 4 bytes. After that, we cast it back to an int and dereference it.

```
*(int*)(((*cstyle_array) + 1) -> 200
```

7. This is analogous to Point 6. We know that the second-level pointer `*cstyle_array` occupies 4 bytes in length because we incremented it. When we dereference it again, it returns the second element at the second level.

```
*((*cstyle_array) + 1) -> 200
```