

Tutorial 12 - Arrays & Pointer Arithmetic in C++

Pointers in C++

What is a Pointer?

- A **pointer** is a data type that holds the address of another variable.
 - **Operators:**
 - `&` → **Address of** operator.
 - `*` → **Dereference** operator (value at address).
-

Basic Pointer Example

- **Code:**

```
1 int a = 3;
2 int* b = &a; // Pointer b stores the address of a
3 cout << "Address of a: " << &a << endl;
4 cout << "Address stored in b: " << b << endl;
5 cout << "Value at address b: " << *b << endl;
6
```

- **Key Points:**
 - `b` holds the address of `a`.
 - `*b` gives the value at the address stored in `b` (value of `a`).
-

Pointer to Pointer

- A **Pointer to Pointer** stores the address of another pointer.
- **Code:**

```
1 int** c = &b; // c holds the address of pointer b
2 cout << "Address of b: " << &b << endl;
3 cout << "Address stored in c: " << c << endl;
4 cout << "Value at address c: " << *c << endl;
5 cout << "Value at value_at(value_at(c)): " << **c << endl;
6
```

- **Key Points:**
 - `*c` gives the address stored in `b`.
 - `**c` gives the value at the address stored in `b` (value of `a`).
-

Summary

- Pointers are essential for managing memory and accessing data indirectly.
- The **address-of** operator (`&`) retrieves the address of a variable.
- The **dereference** operator (`*`) retrieves the value at a pointer's address.
- Pointer-to-pointer allows access to multi-level memory references.