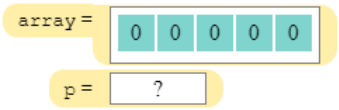
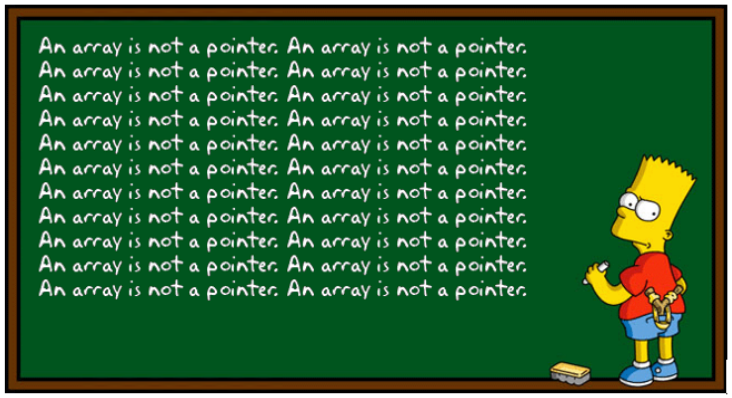
Pointers & Arrays

Because an array name is the same as the address of its first element, you can use it as **a pointer value**. The crucial **difference between arrays and pointers** in C++ comes into play when variables are declared.

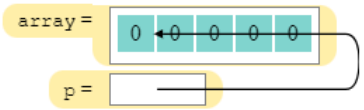
```
int array[5] = {0};
int *p;
```

The distinction between these is **memory allocation**. The first reserves five **int** values (on the stack or in the static storage area); the second **reserves space for a pointer**. The name array is an **address**, not a pointer.





If you define an array, **you have storage to work with**; if you declare a pointer variable, that variable **is not associated with any storage** until you initialize it. The simplest way to initialize a pointer to an array is to **assign the array name to the pointer variable**:




```
int array[5] = {0};
int *p = array;
```

Now, the pointer **p** contains the same address used for **array**.