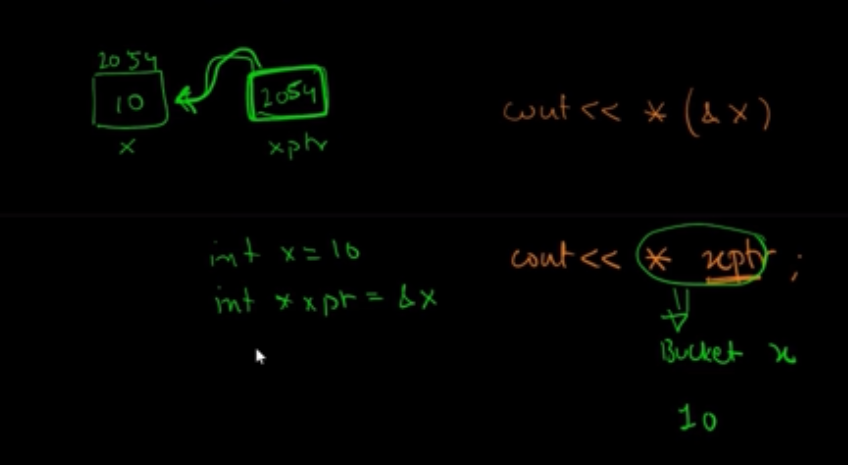
**Address of Operator**:

* It is denoted by &.
* When we use it on a variable C++ shows the hexadecimal address of it.
* Only exception is **character** variables. It prints the character even if we write the address of operator. (This happens due to operator overloading, Cout has this logic that if it gets a character variable address it still prints the character.)
* One way to solve this, we can type cast it to (void\*). It is a bit of hack as we explicitly tell Cout that the address is not of a character type.

**Dereferencing Operator:**

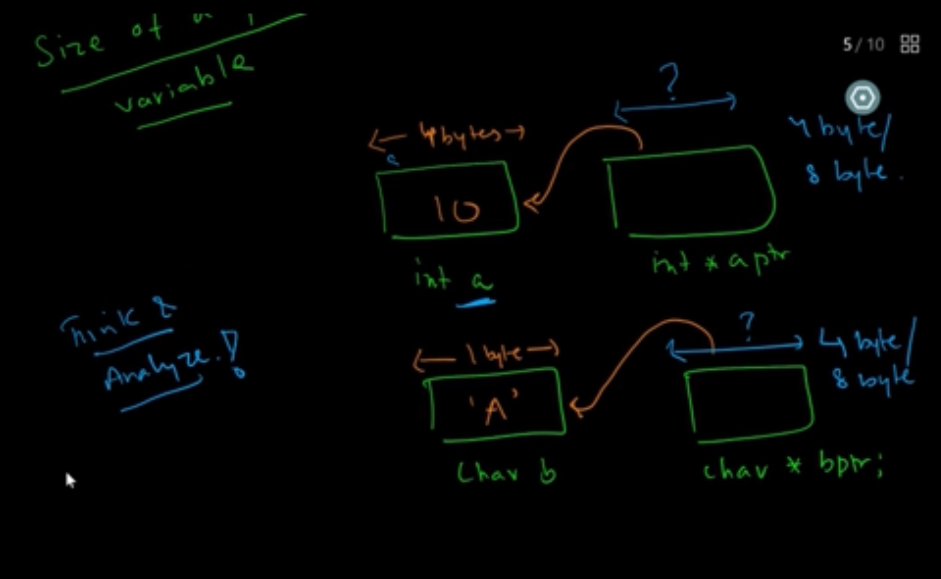
* It is denoted by \*, it can be used for multiplication and declaration of pointer as well.
* By dereferencing a pointer, we can access the variable in which they are pointing to.

****

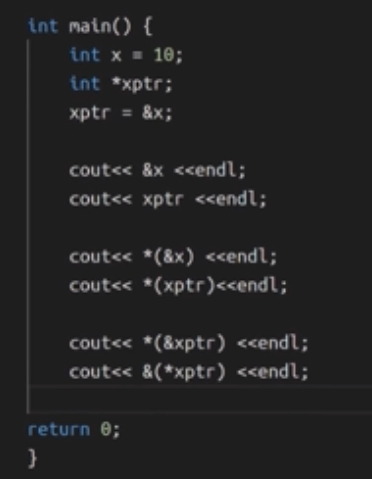
**We can make combination of \* and & with pointer to have interesting operations (do some exercise time to time for practice remembering)**

**Pointer:**

* Pointer is a variable that stores the address of another variable.
* Format is **Data\_type \*variable\_name**
* Example int \*a = &x; (Declaration and assignment)

****

* The size of pointer depends on the system, for example it can be 4-20 bytes in a x86 machine.

****

**Pass by reference using pointers:**

