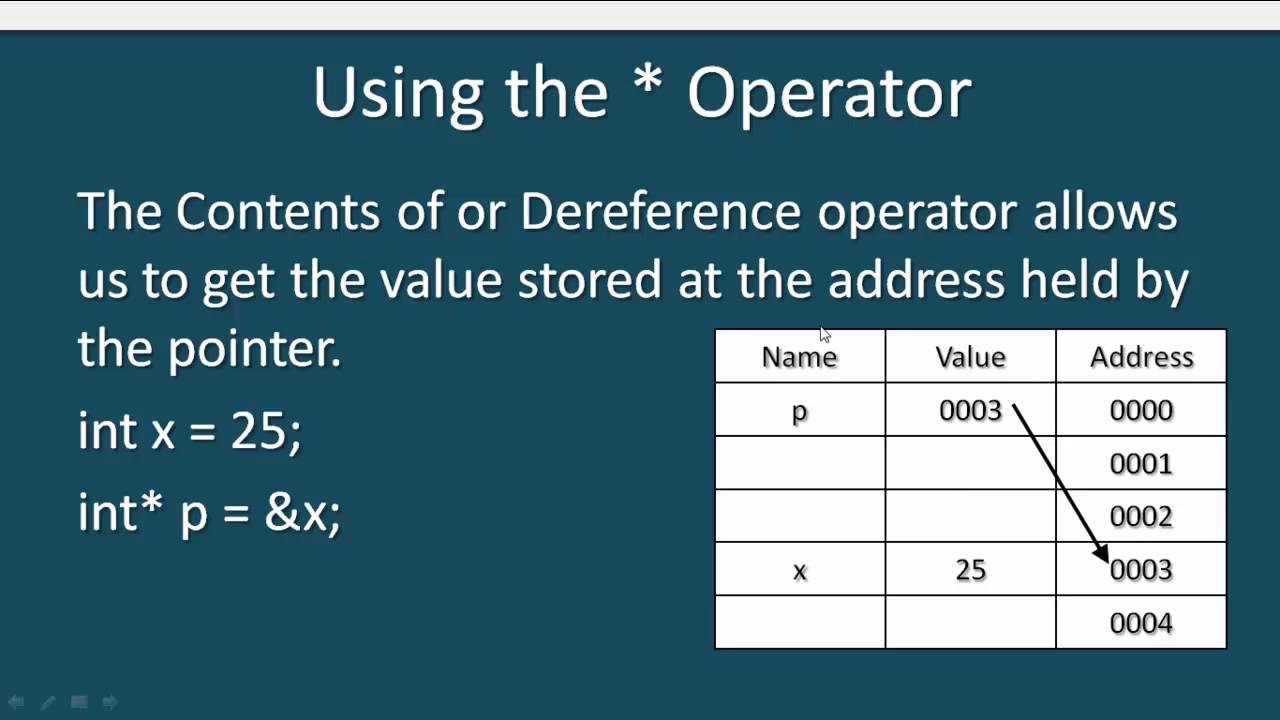
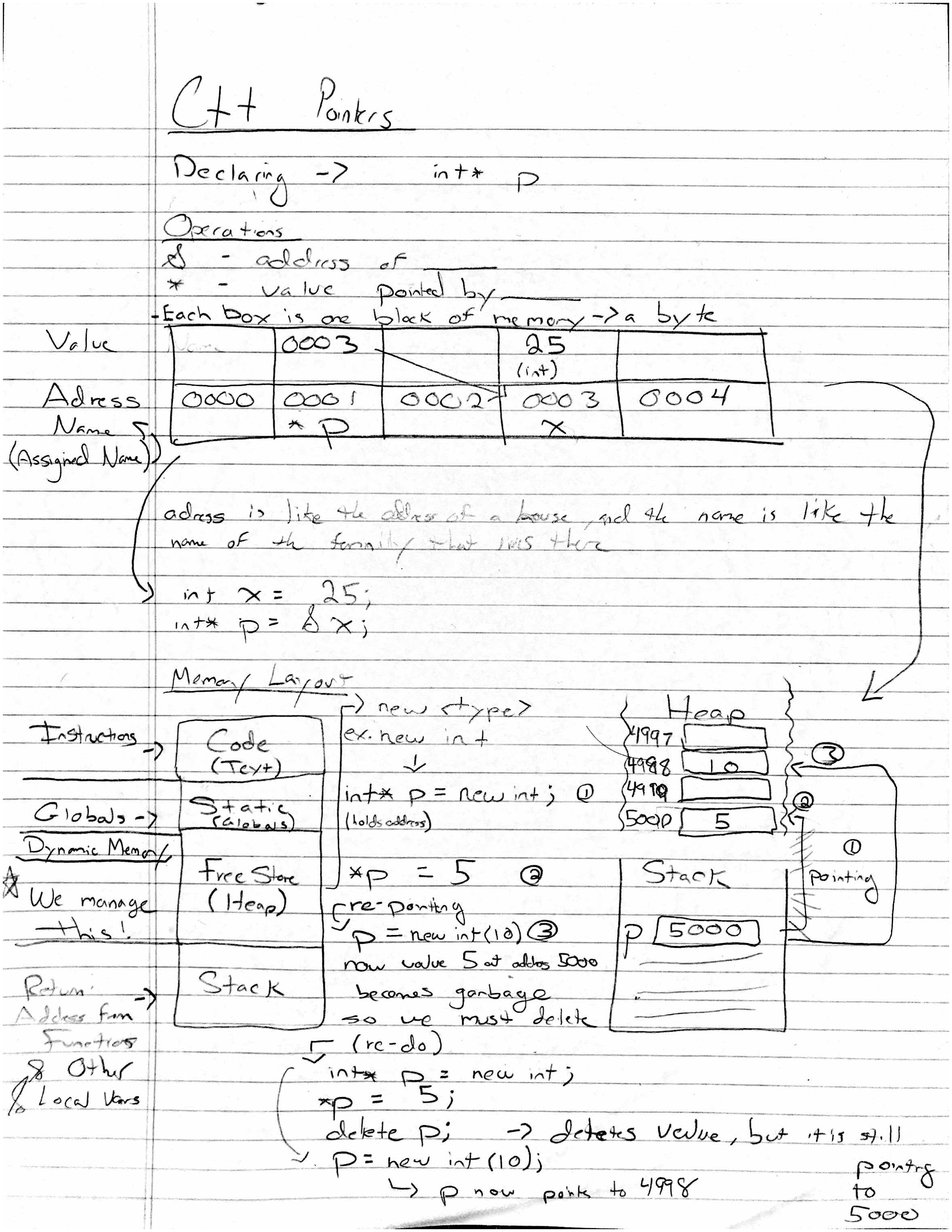
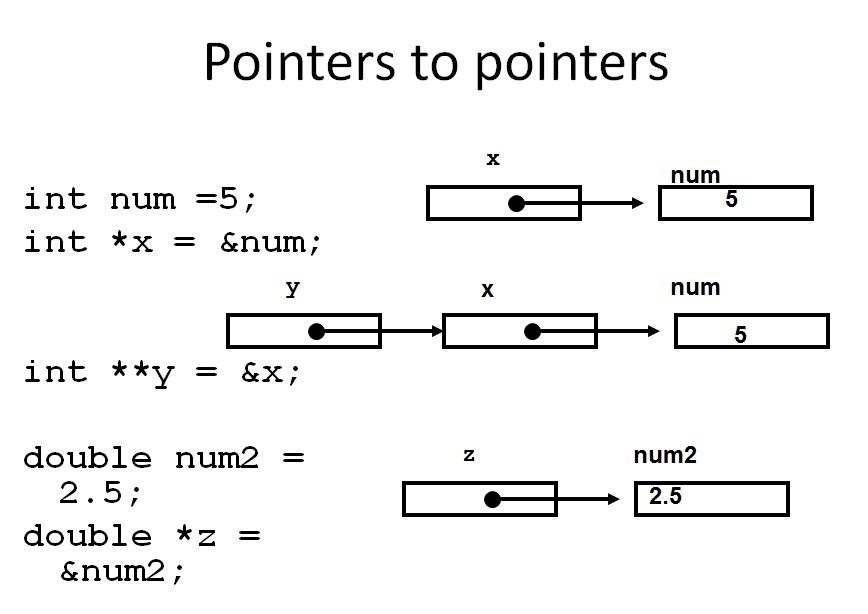
**Pointers**

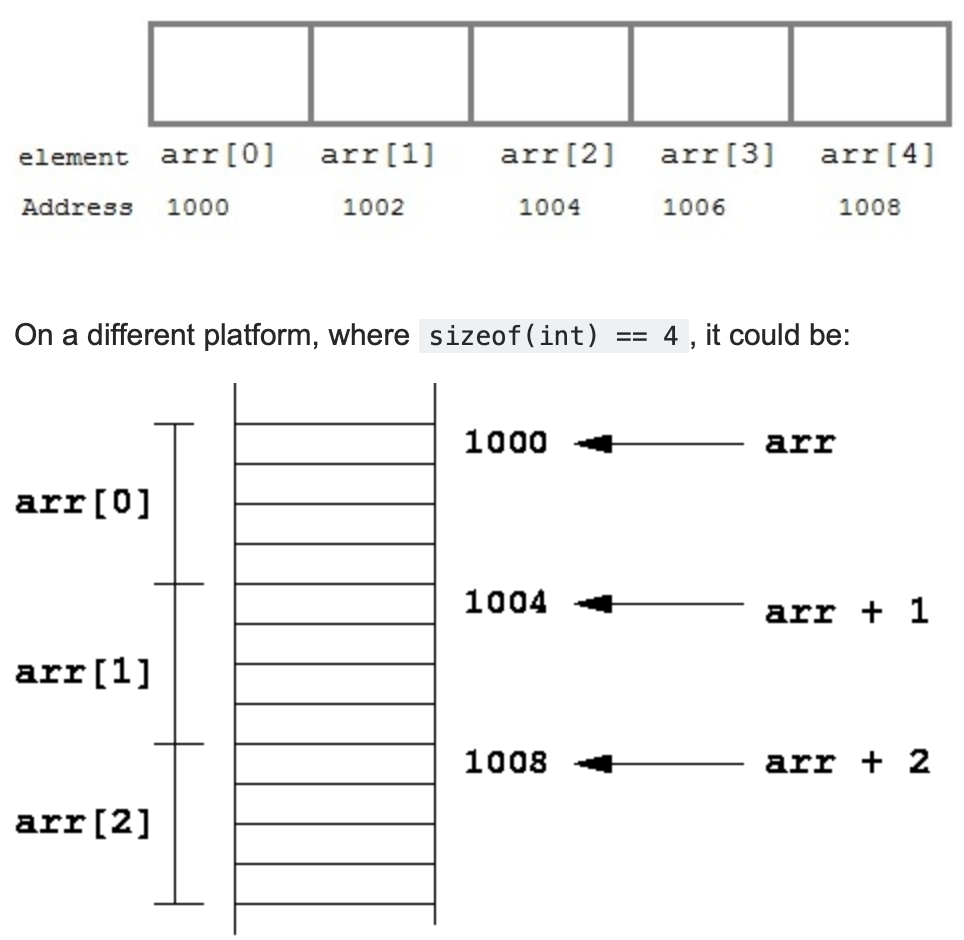
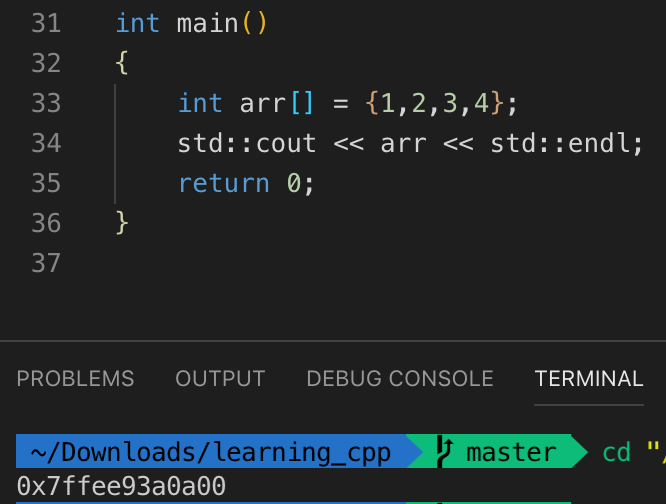
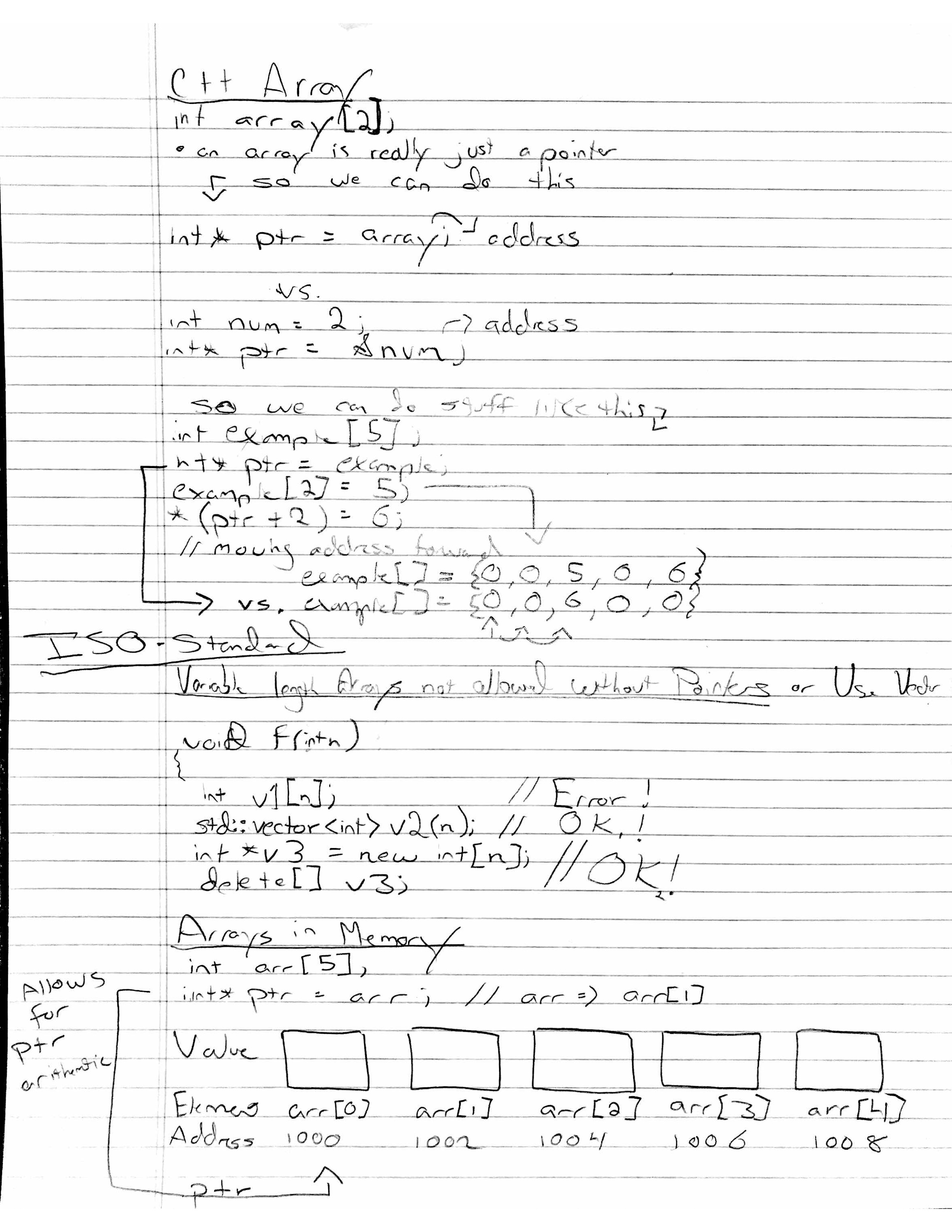
* ****They are stored in memory like other variables, except they store another address for the element value rather than actual value like a char for instances. Note that pointers also have their own address because they are stored in memory too. Here we work with the Heap!



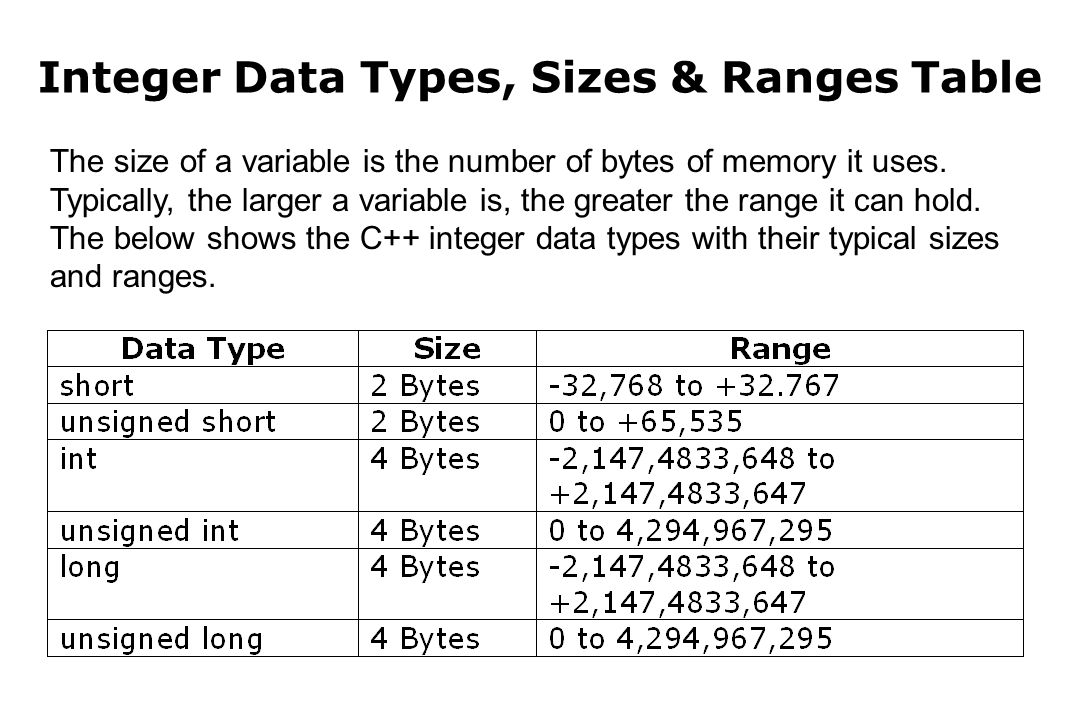
**More on Pointers**

* add info

**Arrays**

* ****Arrays elements are stored in separate but continuous memory locations. Byte size of the data type also matters because it determines the spacing memory address, and the actual blocks of memory provided.

**Basics**

Fundamentals

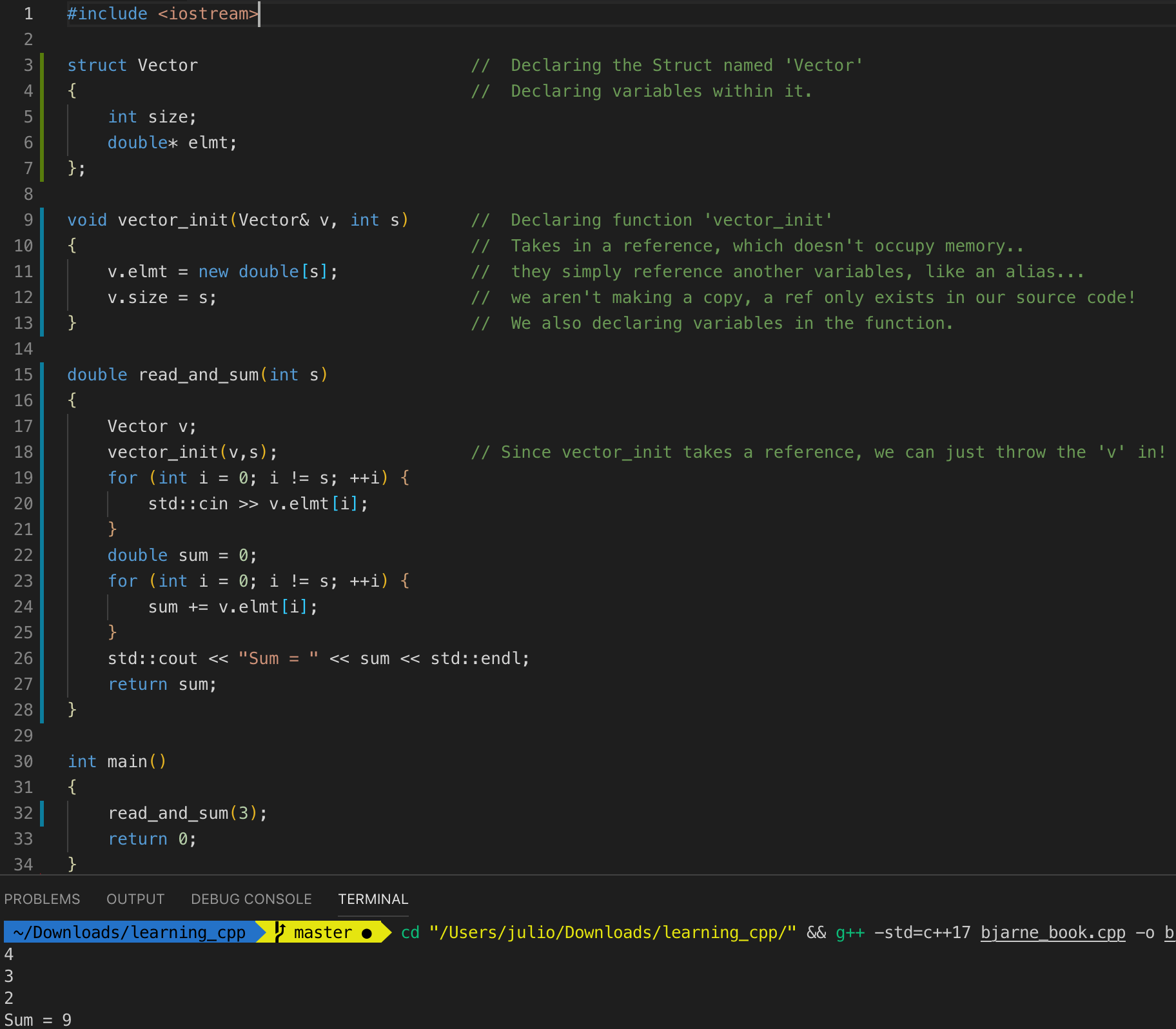
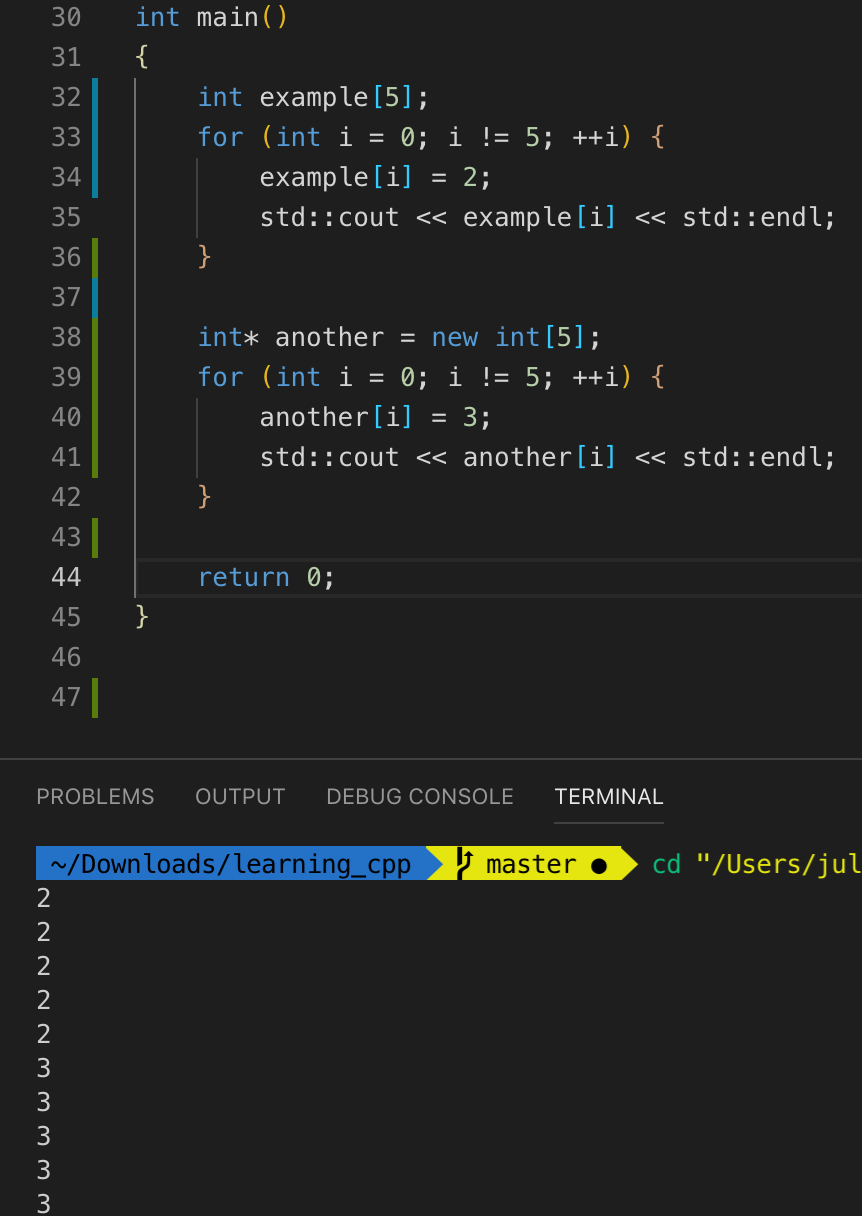
* Type – defines a set of possible values and set of operations for an object.
* Object – is some memory unit that holds a value of some type.
* Value – a set of bits interpreted according to a type... ex. char ‘c’, int 12.
* Variable – is a named object.

Logical

* && is equivalent to ‘and’.
* | | is equivalent to ‘or’.

**Overloading**

* Function overloading when you define multiple functions with the same name.

****