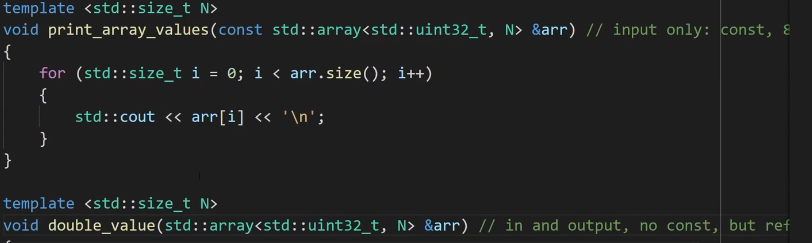
* **Structs:**
* Only use when I want to store data of different types, without the need of methods (if I need methods class is better)
* **Const:**
* Means the variable/object cannot be modified after initialization (Read only)
* The value can be set at runtime or compile time.
* **Constexpr**:
* Stronger guarantee: the value must be computable at compile time (when possible).
* Ensures the expression is a constant expression.
* Useful for array sizes, template parameters, switch cases, etc.
* **Anonymous namespace:**
* A modern C++ way to defining a function private to a single source file. (works like global variables) A screen shot of a computer code

  AI-generated content may be incorrect.
* **Template:** is used to initialize arrays of size “N” (not a prefixed size)
* **Reference to array (pointer):**
* 
* Reference here is made in both cases but const for input only (read only) and in and output
* Use a pointer when you have to – otherwise use reference
* False of a pointer == **nullptr**, true of a pointer != nullptr (use in ifs)
* **lvalues**: An expression that refers to a specific object in memory (has an identifiable address).
* **rvalues**: A temporary value or a value that does not persist beyond the expression (no permanent address).