

Contents	1
Notes by Standard at a Glance	1
Notes belonging to C++11	1
Notes belonging to C++17	1
Notes belonging to C++20	1
1 Tips and Tricks with Templates	1
Note 1: Know the name	1
Note 2: Templates can have type and non-type parameters	1
Note 3: There are no implicit conversions for template parameters	1
Note 4: Alias template for clean Template Meta-Programming (TMP)	1
Note 5: Variable template for clean TMP	1
Note 6: The trailing-return-type with decltype and void()	1
Note 7: Use declval when you need to construct a type for testing during compile-time	1
Note 8: What void_t does	1
Note 9: Keep that array's size	1
Note 10: There is no else if in C++	1
Note 11: More useful than it appears: always_false	1
Note 12: Prefer auto as non-type template parameter (NTTP) to reduce redundancy	1
Note 13: Block template argument deduction	1
Note 14: Fold expressions and the comma operator	1
Note 15: Poor men's fold expressions	1
Note 16: From an array to a pack	1
Note 17: From an array to a pack with templated lambda	1
Note 18: Create objects in place for direct use	1
Note 19: Control the resulting code size	1
Acronyms	1
Bibliography	1
Index	1