QAC++ 6.3.0
requires-expressions

QAC++ 6.2.0
#warning (C++23)
#elifdef / #elifndef (C++23)

QAC++ 5.4.0
Layout-compatibility and pointer-interconvertibility traits (C++20)
Default member initializers for bit-fields (C++20)

QAC++ 5.3.0
consteval (C++20)
constinit (C++20)
explicit(bool) (C++20)
Three-way comparison operator <=> (C++20)
using enum (C++20)
TryCatch blocks in constexpr functions (C++20)

QAC++ 5.2.0
std::is_constant_evaluated (C++20)
Feature-Test macros (C++20)

QAC++ 5.1.0
__VA_OPT__ (C++20)

QAC++ 5.0.0
Guaranteed Copy Elision (C++17)
Dynamic Allocation of Over-aligned Data (C++17) (C++20)
Allow lambda-capture (C++20)
Designated initializers (C++20)
Range-based for statements with initializer (C++20)
Nested inline namespaces (C++20)
Deprecating volatile (C++20)
Length Modifier (C++11)

QAC++ 4.8.0
lambda capture of "*this" (C++17) (C++20)
constexpr lambda (C++17)
aggregates with base classes (C++17)
constant evaluation for all non-type template arguments (C++17)

QAC++ 4.7.0
Structured binding declarations (C++17) (C++20)
Enum direct list initialization (C++17)
In-class explicit specializations (C++17)
UTF-8 character literals (C++17)
Non-type template parameters of placeholder type (C++17)
The use of 'typename' in a template template parameter (C++17)
A using declaration with a declarator list (C++17)
A using declaration with a pack expansion (C++17)

QAC++ 4.6.0
Init statement (C++17) (C++20)
Nested Namespace Definitions (C++17) (C++20)
Inline Variables (C++17)
Exception Specification is Part of the Function Type (C++17)
Class Template Argument Deduction (C++17)
constexpr if Statement (C++17)

Fold Expressions (C++17)
Deduction guides (C++14)
Generalized Constant Expressions (C++14)
_Pragma operator (C++11)

QAC++ 4.3.0
designated initializers (C++20)
__has_include (C++17)
__has_include_next (Langage Extension)
Hexadecimal Floating Point Literals (C++17)
Universal character names (Langage Extension)
thread local storage (C++11)
Diagraph (C++11)
Trigraph (C++17)
Variable Templates (C++14)
constexpr for static member and non-member functions (C++11)
Contextual Implicit Conversions (C++14)
Non-static Data Member Initializers for Aggregate Initialization (C++14)
Sized Deallocation Functions (C++14)
Static Assertions (C++11) (C++17)
decltype of Call Expression with Incomplete Return Type (C++11)
Unrestricted Unions (C++11)

QAC++ 4.1.0
Raw String Literals (C++11)
Generic Lambdas (C++14)
Digit Separator (C++14)
Binary Literals (C++14)
Standard Layout Types (C++11)
Initializer Lists (C++11)
Non Static Data Member Initializers (C++11)
Lambdas (C++11)
Return Type Deduction (C++14)
decltype(auto) (C++14)
nullptr (C++11)
Statement Expressions (Langage Extension)

QAC++ 3.2.2
Extern Template (C++11)
Default Arguments (C++11)
Extension to this (C++11)
Unnamed Namespace Linkage (C++11)
Generalized Attributes (C++11)
User-defined literals (C++11)
noexcept (C++11)
Template aliases (C++11)
Generalized Constant Expressions (C++11)
Alignment Support (C++11)
Inheriting constructors (C++11)
Delegating constructors (C++11)
Range-based for (C++11)

QAC++ 3.1
Rvalue references (C++11)
Variadic templates (C++11)
=default (C++11)
=delete (C++11)
Override (C++11)
auto (C++11)

trailing-return-type (C++11)
std::make_shared (C++11)

QAC++ 3.0
Strongly typed/scoped enums (C++11)
The auto keyword (C++11)
The decltype keyword (C++11)
Explicit conversion operators (C++11)
Inline namespaces (C++11)
Defaulted Functions (C++11)
Deleted Functions (C++11)