constexpr comparison operators for std::array

Tristan Brindle (tcbrindle@gmail.com)

1 Introduction

The interface for `std::array` was made mostly `constexpr` in C++17. However, this did not include the comparison operators, presumably because they are specified in terms of `std::equal()` and `std::lexicographical_compare()`, which only recently became `constexpr` in the draft International Standard.

This paper corrects this surprising omission by adding the `constexpr` keyword to the declarations of the `array` comparison operators.

2 Proposed wording

Changes are relative to [N4741].

In section 26.3.2 [array.syn], change

```
template<class T, size_t N>
  constexpr bool operator==(const array<T, N>& x, const array<T, N>& y);
template<class T, size_t N>
  constexpr bool operator!=(const array<T, N>& x, const array<T, N>& y);
template<class T, size_t N>
  constexpr bool operator< (const array<T, N>& x, const array<T, N>& y);
template<class T, size_t N>
  constexpr bool operator> (const array<T, N>& x, const array<T, N>& y);
template<class T, size_t N>
  constexpr bool operator<=(const array<T, N>& x, const array<T, N>& y);
template<class T, size_t N>
  constexpr bool operator>=(const array<T, N>& x, const array<T, N>& y);
```

References

[N4741] Richard Smith. Working Draft, Standard for Programming Language C++. http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2018/n4741.pdf, 2018 (accessed 2018-04-20).