

Краткий план обзора стандартов C++ - dumb

стандарт 1998 - стандарт 2003

- технические правки. непрерывность `std::vector`
- smb://192.168.1.19/disk_d/docs/standards/iso14882-C++98.pdf
- smb://192.168.1.19/disk_d/docs/standards/INCITS+ISO+IEC+14882-2003.pdf

стандарт c++0x

- рабочие материалы комитета: <http://www.open-std.org/jtc1/sc22/wg21/>
- тенденции (оптимизация, шаблоны, boost)
- изменения в языке (редакция n2369 smb://192.168.1.19/disk_d/docs/standards/c++-drafts/n2369.pdf)
 - лексика:
 - ключевые слова: `alignas`, `alignof`, `char16_t`, `char32_t`, `constexpr`, `decltype`, `static_assert`; - 2.11:1
 - символные и строковые литералы `u/U` - ISO 10646 (http://unicode.org/faq/unicode_iso.html, <http://www.iso.org/PubliclyAvailableStandards>); - 2.13.2, 2.13.4;
 - объявления:
 - новый смысл `auto` (`auto i=a.begin();`); - 7.1.6.4:2;
 - спецификация выравнивания (`T alignas(T) alignas(A) buffer[N];`); - 5.3.6, 7.1.7;
 - спецификация базового типа `enum` (`enum E int {...}`); `scoped-enum` (`enum class E int {...}`); - 7.2;
 - `constexpr` - константные функции; - 7.1.5;
 - `decltype`; - 7.1.6.2:4; (пример)
 - `alias-declaration` (`using Int=int;`); - 7.1.3:2;
 - `rvalue-references`; - 8.3.2; (пример)
 - `=default` в специальных функциях (конструкторы, деструкторы, операторы присваивания) - 8.4:9;
 - `=delete` для явного запрета ссылки на функцию (вызова, взятия адреса) - 8.4:10;
 - классы:
 - делегирование конструкторов; - 12.6.2:2;
 - шаблоны:
 - угловые скобки интерпретируются по контексту; - 14.2:3; (пример)
 - `variadic templates` (`template<typename T...> class tuple;`); - 14.5.3; (пример!)
 - `template aliases` (`template<typename T> using Vec=vector<T, Alloc<T>;`); - 14.5.7;
 - изменение требований к значениям шаблонных параметров по-умолчанию (пример) - 14.1:9 (в сравн. с 2003), 14.1:11
 - препроцессор:
 - разрядность препроцессорной целочисленной арифметики - `intmax_t` (было `long`); - 16.1:4;

- variadic macros; - 16.3;
- оператор `_Pragma`; - 16.9;
- изменения в библиотеке
 - general utilities:
 - изменился набор requirements - появились новые концепции; - 20.1.1;
 - type traits вошли в стандарт (`std::is_...<T>::value`); - 20.4;
 - tuples вошли в стандарт (`template<class ...Type> tuple`); - 20.3;
 - `auto_ptr` стал `unique_ptr`; - 20.6.5;
 - `smart_pointers` вошли в стандарт; - 20.6.6;
 - containers:
 - появился `array` (`template<typename T,size_t N> class array`); - 23.2.1;
 - появился `bitset` (`template<size_t N> class bitset`); - 23.3.5;
 - появились `unordered_[multi]map` и `set` (`template<typename Key, T, Hash=hash<Key>, Pred=equal<Key>,Alloc>`); - 23.4;
 - numerics library:
 - появились серьезные методы работы с [псевдо]случайными величинами (`<random>`); - 26.4;
 - regular expressions:
 - вошли в стандарт (`<regex>`); - 28;

компиляторы с поддержкой 0x

g++4.3

Статус поддержки стандарта: http://gcc.gnu.org/gcc-4.3/cxx0x_status.html

Доступен для всех по имени g++-4.3 на многострадальном 192.168.81.5.
Исходники: <ftp://192.168.81.7/distrib/gcc>
При компиляции указывайте опцию `-std:c++0x`

From:
<http://wiki.osll.ru/> - **Open Source & Linux Lab**

Permanent link:
http://wiki.osll.ru/doku.php/etc:users:yuri_v_katkov:max_std0x

Last update: **2016/08/08 20:53**

