

C++ Club UK Meeting 116

Gleb Dolgich

2020-11-05

- mailing2020-10
 - Reddit

Select papers

- P1206R1 `ranges::to`: A function to convert any range to a container
- P2214R0 A Plan for C++23 Ranges
- P2226R0 A proposal for an idiom to move from an object and reset it to its default constructed state
- P2237R0 Metaprogramming

Named Parameters in C++20

Peter Dimov

- [Reddit](#)

C++ in Visual Studio Code reaches version 1.0

Julia Reid

- [Reddit](#)

C++ Talk Index

Website

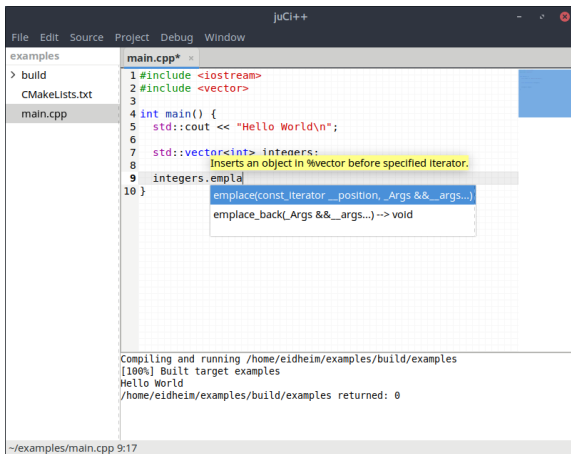
- [Reddit](#)

The terrible **size_t**

Reddit

juCi++: a lightweight, cross-platform IDE

- GitLab
- Installation guide



The screenshot displays the juCi++ IDE window. The title bar reads "juCi++". The menu bar includes "File", "Edit", "Source", "Project", "Debug", and "Window". On the left, a project explorer shows a tree structure with "examples" as the root, containing "build", "CMakeLists.txt", and "main.cpp". The "main.cpp" file is selected and open in the main editor. The code in the editor is as follows:

```
1 #include <iostream>
2 #include <vector>
3
4 int main() {
5     std::cout << "Hello World\n";
6
7     std::vector<int> integers;
8     // Inserts an object in %vector before specified iterator.
9     integers.emplace
10 }
```

A tooltip is visible over the `emplace` call, showing the following signatures:

```
emplace(const_iterator __position, _Args &&__args...)
emplace_back(_Args &&__args...) -> void
```

Below the editor, a console window shows the output of the compilation and execution:

```
Compiling and running /home/eidheim/examples/build/examples
[100%] Built target examples
Hello World
/home/eidheim/examples/build/examples returned: 0
```

The status bar at the bottom indicates the current file and line: `~/examples/main.cpp 9:17`.

Figure 1: Screenshot

Library: cpp-lazy

GitHub

Cpp-lazy is a fast and easy lazy evaluation library for C++14/17/20.

Lazy evaluation is an evaluation strategy which holds the evaluation of an expression until its value is needed. In this library, all the iterators are lazy evaluated.

This library is not a replacement for `ranges::v3` but rather a (smaller) alternative.


Library: Crypto3

- Home page
- Boost mailing list announcement
- GitHub
- Reddit

Library: AAA - Auxiliary Arithmetic Algorithms

- [GitHub](#) (MIT)
- [Docs](#)

Daisy Hollman's deduction trick



Daisy Hollman
@The_Whole_Daisy

Cute C++ trick of the day: C++17 deduction guides and class template argument deduction make it easier than ever to use the "rule of zero" for constructors, even for classes with relatively specific template parameters to deduce: godbolt.org/z/bvGWMq
pic.twitter.com/2NxZDCBJvL

```
1 #include <vector>
2 #include <initializer_list>
3 template <class T, class U, class V>
4 struct Foo {
5     T bar;
6     std::vector<U> baz;
7     V foofoo;
8 };
9 template <class T, class U, class V>
10 Foo(T, std::initializer_list<U>, V) -> Foo<T, U, V>;
11 int main() {
12     auto test = Foo{42, {3.14, 2.718, 1.618}, 'x'};
13 }
```

116 Likes

18 Retweets

13 Oct 2020 at 17:17 via Twitter Web App

Figure 2: Deduction guides

How it started/How it's going



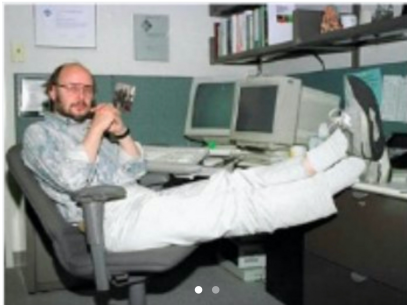
S is for Shafik who stared into the void too long

@shafikyagmour

How it started:

How it's
going:

#cplusplus pic.twitter.com/cINZFGUJLN



92 Likes

9 Retweets

12 Oct 2020 at 05:10

via **TweetDeck**



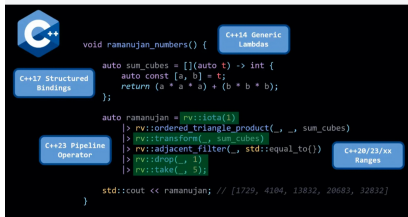
S is for Shafik who stared into the void too long

@shafikyagmour

How it started:

How it's
going:

#cplusplus pic.twitter.com/cINZFGUJLN



92 Likes

9 Retweets

12 Oct 2020 at 05:10

via **TweetDeck**

Halloween logic

