Best introduction to C++ template metaprogramming? [closed]

Asked 16 years, 3 months ago Modified 5 years, 11 months ago Viewed 73k times

123

Closed. This question is seeking recommendations for software libraries, tutorials, tools, books, or other off-site resources. It does not meet Stack Overflow guidelines. It is not currently accepting answers.

We don't allow questions seeking recommendations for software libraries, tutorials, tools, books, or other off-site resources. You can edit the question so it can be answered with facts and citations.

Closed 7 years ago.

Improve this question

Static metaprogramming (aka "template metaprogramming") is a great C++ technique that allows the execution of programs at compile-time. A light bulb went off in my head as soon as I read this canonical metaprogramming example:

```
#include <iostream>
using namespace std;

template< int n >
struct factorial { enum { ret = factorial< n - 1 >::ret * n }; };

template<>
struct factorial< 0 > { enum { ret = 1 }; };

int main() {
    cout << "7! = " << factorial< 7 >::ret << endl; // 5040
    return 0;
}
```

If one wants to learn more about C++ static metaprogramming, what are the best sources (books, websites, on-line courseware, whatever)?

c++ templates metaprogramming

Share
Improve this question
Follow

edited Sep 22, 2010 at 4:41
Georg Fritzsche
98.9k ● 26 ● 196 ● 236

asked Sep 21, 2008 at 21:50
jwfearn
29.5k ● 28 ● 100 ● 123

50 lightbulb went 'off' or 'on' ? – horseyguy May 4, 2010 at 18:58

43 Off. Definitely off. – Thomas Eding Aug 30, 2012 at 19:05

8 Answers

Sorted by: Highest score (default)

124

[Answering my own question]

The best introductions I've found so far are chapter 10, "Static Metaprogramming in C++" from *Generative Programming, Methods, Tools, and Applications* by Krzysztof Czarnecki and Ulrich W. Eisenecker, ISBN-13: 9780201309775; and chapter 17, "Metaprograms" of *C++ Templates: The Complete Guide* by David Vandevoorder and Nicolai M. Josuttis, ISBN-13: 9780201734843.

Todd Veldhuizen has an excellent tutorial here.

A good resource for C++ programming in general is *Modern C++ Design* by Andrei Alexandrescu, ISBN-13: 9780201704310. This book mixes a bit of metaprogramming with other template techniques. For metaprogramming in particular, see sections 2.1 "Compile-Time Assertions", 2.4 "Mapping Integral Constants to Types", 2.6 "Type Selection", 2.7 "Detecting Convertibility and Inheritance at Compile Time", 2.9 "`NullType` and `EmptyType`" and 2.10 "Type Traits".

The best intermediate/advanced resource I've found is *C++ Template Metaprogramming* by David Abrahams and Aleksey Gurtovoy, ISBN-13: 9780321227256

If you'd prefer just one book, get *C++ Templates: The Complete Guide* since it is also the definitive reference for templates in general.

Share

Improve this answer

Follow

edited Jan 7, 2019 at 7:41

community wiki
12 revs, 2 users 92%
jwfearn

2 Link to Todd Veldhuizen's piece has gone stale. – John R. Strohm Dec 6, 2012 at 16:51

1 I believe its this: www10.informatik.uni-erlangen.de/~pflaum/pflaum/ProSeminar/... – 0fnt Jan 30, 2013 at 14:12

2 link fixed now, thanks! – jwfearn Feb 15, 2013 at 21:01

The link is broken again! – Paul Sep 8, 2015 at 6:09

1 Thank you @jwfearn, I read the article, although I found this more interesting: youtube.com/watch?v=Am2is2QCvxY (it has two parts). It is a talk by Walter E. Brown: Modern Template Metaprogramming: A Compendium – Paul Sep 8, 2015 at 18:42

25

Andrei Alexandrescu's Modern C++ Design book covers a lot of this and other tricks for speedy and efficient modern C++ code and is the basis for the Loki library.

Also worth mentioning is the Boost libraries, which heavily use these techniques and are usually of very high quality to learn from (although some are quite dense).

Share
Improve this answer
Follow

edited Mar 4, 2014 at 17:13

Dan Forbes
2,824 ● 4 ● 32 ● 62

answered Sep 21, 2008 at 21:56

Alaric
839 ● 8 ● 10

12

Modern C++ Design, a brilliant book and design pattern framework by Alexandrescu. Word of warning, after reading this book I stopped doing C++ and thought "What the heck, I can just pick a better language and get it for free".

Share Improve this answer Follow

answered Sep 21, 2008 at 21:56

Florian Bösch
27.7k ● 12 ● 51 ● 53

6 "What the heck, I can just pick a better language and get it for free" Ummm, what do you mean? I'm especially confused by "for free". And what other language did you had in mind? – Pop Catalin Sep 21, 2008 at 22:16

13 C++ template metaprogramming gives you all kind of new capabilities like passing types or list of types as arguments etc. Most of these capabilities are present in dynamically typed languages like python, with nicer syntax. – Florian Bösch Sep 23, 2008 at 11:04

1 If you're looking for a language that you can write correct programs quickly, Python is better than C++. If you need some of the other things C++ provides, adding Lisp-like capability to C++ may be better than trying to C++-ify another language. – David Thornley Mar 26, 2009 at 20:03

7

Two good books that spring to mind are:

- Modern C++ Design / Andrei Alexandrescu (It's actually 7 years old despite the name!)
- C++ Templates: The Complete Guide / Vandevoorde & Josuttis

It's quite an in-depth field, so a good book like one of these is definitely recommended over websites. Some of the more advanced techniques will have you studying the code for some time to figure out how they work!

Share Improve this answer Follow

answered Sep 21, 2008 at 21:59

Nik

1,364 ● 8 ● 7

5

Modern C++ is one of the best introductions I've read. It covers actual useful examples of template metaprogramming. Also take a look at the companion library Loki.

Share Improve this answer Follow

answered Sep 21, 2008 at 21:54

Eclipse

45.5k ● 20 ● 116 ● 172

5

There won't be a large list of books, as the list of people with a lot of experience is limited. Template metaprogramming started for real around the first C++ Template Programming Workshop in 2000, and many of the authors named so far attended. (IIRC, Andrei didn't.) These pioneers greatly influenced the field, and basically what should be written is now written. Personally, I'd advice Vandevoorde & Josuttis. Alexandrescu's is a tough book if you're new to the field.

Share Improve this answer Follow

answered Sep 21, 2008 at 22:40

MSalters

179k ● 11 ● 164 ● 368

4

Veldhuizen's original papers were good. If you up for a whole book, then there's Vandevoorde's book "C++ Templates Complete Guide". And when you're ready for the master's course, try Alexandrescu's Modern C++ Design.

Share Improve this answer Follow

answered Sep 21, 2008 at 21:56

eduffy

40.2k ● 14 ● 98 ● 93

google Alexandrescu, Modern C++ Design: Generic Programming and Design Patterns Applied

4

Share

Improve this answer

Follow

edited Sep 21, 2008 at 23:01

answered Sep 21, 2008 at 21:53

Maxim Ananyev

690 ● 6 ● 13