What is the single most influential book every programmer should read? [closed]

Asked 16 years, 4 months ago Modified 12 years, 10 months ago

Viewed 1.1m times

1434

votes

As it currently stands, this question is not a good fit for our Q&A format. We expect answers to be supported by facts, references, or expertise, but this question will likely solicit debate, arguments, polling, or extended discussion. If you feel that this question can be improved and possibly reopened, visit the help center for guidance.

Closed 12 years ago.

Locked. This question and its answers are locked because the question is off-topic but has historical significance. It is not currently accepting new answers or interactions.

If you could go back in time and tell yourself to read a specific book at the beginning of your career as a developer, which book would it be?

I expect this list to be varied and to cover a wide range of things.

To search: Use the search box in the upper-right corner. To search the answers of the current question, use `inquestion:this`. For example:

```
inquestion:this "Code Complete"
```

resources

Share

edited Sep 26, 2011 at 15:39

community wiki
18 revs, 13 users 34%
Robert Harvey

8 Browsing this thread make me realize how ugly most programming related books are. Very good thread though! – Carl Bergquist Aug 5, 2009 at 12:09

23 Interesting this is, while the title reads "What is the single most influential book *every* programmer should read?", there are quite a few books suggested that deal with language specific topics. By definition, and by question as it was put, the books suggested here should deal with language agnostic topics, which proves most programmers have yet to learn how to read. – Rook Oct 2, 2009 at 19:54

19 If I could go back in time and tell myself to read something, it better be a newspaper or sports fact book that I carried with me. Anything else is a waste of good time travel. :-) – jmucchiello Nov 8, 2009 at 9:38

32 You know, if I wasn't worried about getting down voted a WHOLE lot I would trollishly go and suggest Twilight. "Its ALSO about people who are pale and avoid the sun!" – Jacob Bellamy Feb 12, 2010 at 0:20

3 Can someone cleanup answers by deleting repeated entries on books? Most of them area already down voted. – rao Mar 16, 2010 at 5:44

Comments disabled on deleted / locked posts / reviews |

214 Answers

Sorted by: Highest score (default)

Prev 1 … 4 5 6 7 8 Next

2 votes

I would say that "Beyond Code - Learn to Distinguish Yourself in 9 Simple Steps" is quite a good and motivational book. I doesn't cover technical issues, but it describes ways of working with people, being professional, ... For me, this is a book you can read again and again if you are in need of some pep talk. Besides that, it is cheap and very easy and enjoyable to read in 3 to 4 hours.

There is a little review over at my blog.

Share

edited Sep 2, 2011 at 6:48

community wiki

2 revs, 2 users 92%
Christophe Herreman

2
votes

My high school math teacher lent me a copy of Are Your Lights Figure Problem that I have re-read many times. It has been invaluable, as a developer, and in life generally.

Share

edited Sep 2, 2011 at 6:50

community wiki
2 revs, 2 users 67%
gcamp

1
vote

Implementation Patterns by Kent Beck.

alt text http://ecx.images-amazon.com/images/I/51JHn-6oNwL._SL500_AA240_.jpg

You can learn how to communicate people with programming.

Share

answered Aug 21, 2008 at 21:26

community wiki
popopome

1

Deitel and Deitel, "C++: How to Program" vote

XUnit Test Patterns

Share

answered Aug 25, 2008 at 16:53

community wiki
Mike Caron

1

vote

Code is Law - you are doing all this writing, editing, and thinking in [language of your choice] but WHY? What does you code MEAN? What will does it actually DO?

(I could have recommended a book on QA, but I didn't...)

Share

answered Sep 4, 2008 at 1:10

community wiki
benc

1

vote

Pro Spring is a superb introduction to the world of Inversion of Control and Dependency Injection. If you're not aware of these practices and their implications - the balance of topics and technical detail in Pro Spring is excellent. It builds a great case and consequent personal foundation.

Another book I'd suggest would be Robert Martin's Agile Software Development (ASD). Code smells, agile techniques, test driven dev, principles ... a well-written balance of many different programming facets.

More traditional classics would include the infamous GoF Design Patterns, Bertrand Meyer's Object Oriented Software Construction, Booch's Object Oriented Analysis and Design, Scott Meyer's "Effective C++'" series and a lesser known book I enjoyed by Gunderloy, Coder to Developer.

And while books are nice ... don't forget radio!

... let me add one more thing. If you haven't already discovered safari - take a look. It is more addictive than stack overflow :-) I've found that with my google type habits - I need the more expensive subscription so I can look at any book at any time - but I'd recommend the trial to anyone even remotely interested.

(ah yes, a little obj-C today, cocoa tomorrow, patterns? soa? what was that example in that cookbook? What did Steve say in the second edition? Should I buy this book? ... a subscription like this is great if you'd like some continuity and context to what you're googling ...)

Share

edited Sep 6, 2008 at 16:15

community wiki
2 revs
Luther Baker

1 +1 for Object Oriented Software Construction - a must read. – Tobias Langner Oct 31, 2009 at 22:48

1
vote

Here are two I haven't seen mentioned:

I wish I had read "Ruminations on C++" by Koenig and Moo much sooner. That was the book that made OO concepts really click for me.

And I recommend Michael Abrash's "Zen of Code Optimization" for anyone else planning on starting a programming career in the mid 90s.

Share

answered Sep 16, 2008 at 1:12

community wiki
AShelly

Can you narrow it down to one please? – Ruben Bartelink Mar 27, 2010 at 14:06

1
vote

Modern C++ Design by Andrei Alexandrescu

Share

answered Sep 16, 2008 at 13:54

community wiki
Ferruccio

1

vote

Writing Solid Code by Steve Maguire.

Share

answered Sep 16, 2008 at 13:59

community wiki

1

vote

"Object-Oriented Analysis and Design with Applications" by Grady Booch. I read this a long time ago and it showed me that there could be a methodology to developing Object Oriented Software. Since then many other books have had an impact on me but this one got me started.

Share

answered Sep 16, 2008 at 14:01

community wiki
David Dibben

1

vote

Mine is Test Driven Development by Example

Share

answered Sep 16, 2008 at 14:19

community wiki
jop

1
vote

Learning C# 2005, by Jesse Liberty & Brian MacDonald (O'Reilly).

ISBN 10: 0-596-10209-7.

When I first made the jump from ASP classic procedural code to object-oriented C# code in VS2005, this book set me on the right path.

Share

answered Sep 16, 2008 at 15:46

community wiki
Kelly Adams

1
vote

Software Tools by Brian W. Kernighan and P. J. Plauger by a wide margin had the most effect on me.

Share

answered Sep 16, 2008 at 15:55

community wiki
Bryan Oakley

1
vote

Inside the C++ Object Model by Stan Lippman. It made C++ finally "click" for me, before it was all "magic". This book gave me a different frame of mind when approaching a new programming language.

Share

answered Sep 16, 2008 at 16:15

community wiki
Eddie Velasquez

1
vote

Literate Programming by Donald Knuth, it's a great book on code structure.

Share

answered Sep 16, 2008 at 16:27

community wiki
agamotto

1
vote

The Productive Programmer by Ford

I'm not quite through this one yet, but I'm already thrilled by some of the tips/tricks I've picked up to become more...well...productive.

Sure, there's plenty of the stuff we all already know (use the keyboard shortcuts, DRY, etc). But there's plenty of new stuff to go with it. And careful readers will quickly start to see how things can be combined for even greater effect.

Share

answered Sep 16, 2008 at 17:42

community wiki
Snibble

1
vote

Object Oriented Analysis and Design - by Grady Booch

Share

answered Sep 16, 2008 at 17:45

community wiki
Vaibhav

1
vote

"Thinking in C++" by Bruce Eckel

Share

answered Sep 16, 2008 at 17:53

community wiki
DavidGR

1
vote

Donald Norman, 'The Design of Everyday Things'

Not about programming, per se, but about how things in the world should *work* -- kind of the psychology of usability.

It's been invaluable for me in designing both end-user interfaces and APIs.

Share

answered Sep 16, 2008 at 18:55

community wiki
Nate

1

vote

How to think like a computer scientist: learning with python

May not be the most advanced book on the world but it made me understand programming concepts that I couldn't, especially object oriented topics.

Share

answered Sep 17, 2008 at 0:52

community wiki
Gustavo Rubio

1

vote

Agile Software Development with Scrum by Ken Schwaber and Mike Beedle.

I used this book as the starting point to understanding Agile development.

Share

answered Sep 17, 2008 at 4:44

community wiki
Jay Walker

1

vote

The Pragmatic programmer was pretty good. However one that really made an impact when I was starting out was :

Windows 95 System Programming Secrets"

I know - it sounds and looks a bit cheesy on the outside and has probably dated a bit - but this was an awesome explanation of the internals of Win95 based on the Authors (Matt Pietrek) investigations using his own own tools - the code for which came with the book. Bear in mind this was before the whole open source thing and Microsoft was still pretty cagey about releasing documentation of internals - let alone source. There was some quote in there like "If you are working through some problem and hit some sticking point then you need to stop and really look deeply into that piece and really understand how it works". I've found this to be pretty good advice - particularly these days when you often have the source for a library and can go take a look. Its also inspired me to enjoy diving into the internals of how systems work, something that has proven invaluable over the course of my career.

Oh and I'd also throw in effective .net - great internals explanation of .Net from Don Box.

Share

answered Sep 17, 2008 at 5:49

community wiki

Ian

1

vote

In recent years it has been 'The C++ Standard Library' by 'Nicolai M. Josuttis'. It's my bible.

alt text http://ecx.images-amazon.com/images/I/51BT5SKXTCL._SL500_AA240_.jpg

Share

answered Sep 17, 2008 at 7:45

community wiki

Rob

1

vote

The first book that made a real impact on me was *Mastering Turbo Assembler* by Tom Swan.

Other books that have had an impact was *Just For Fun* by Linus Torvalds and David Diamond and of course *The Pragmatic Programmer* by Andrew Hunt and David Thomas.

Share

answered Sep 17, 2008 at 8:58

community wiki

Jonas Engström

1

vote

If you are doing anything in Unix/Linux/MacOS etc, you must read Advanced Programming in the Unix Environment (also known by the acronym APUE), by the late W Richard Stevens. If you don't know how file descriptors work or what sessions are, or all the things you should do when you daemonize yourself (admit it, you don't), then this book will tell you.

You'll feel amatuerish for a bit afterwards, but if you want to consider yourself a professional programmer (in any language) in the Unix environment you need to read this.

Share

answered Sep 17, 2008 at 19:37

community wiki
Lot105

1

vote

Even though I had been programming rofessionally for years, Rocky Lhotka's "Business Objects" series about his CSLA framework was the book that opened my eyes.

His ideas he got me excited about software development patterns and theory again. It set me on the path of a new interest in learning how to be a better developer, and not just learning about the latest gee-whiz control or library. (Don't get me wrong, I still love a good technical book too - you gotta keep up!)

Share

answered Sep 20, 2008 at 19:17

community wiki
Doug L.

1

vote

I found the [The Algorithm Design Manual]() to be a very beneficial read. I also highly recommend [Programming Pearls]().

Share

answered Sep 20, 2008 at 19:30

community wiki
daniel

1

vote

recommended for Windows Programmer, Programming Windows

Share

answered Sep 21, 2008 at 12:09

community wiki
MovingSpotlight

1

vote

Anything by Edward Tufte: The Visual Display of Quantitative Information; Envisioning Information; Visual Explanations

Share

answered Oct 2, 2008 at 22:46

community wiki
Geoff Snowman

1

vote

OK, so the question is not "what's the best programming book", but "if you could tell yourself what to read in the beginning of your career"...

Probably one of "On Lisp" and SICP, plus one of CLRS or "Algorithms: a creative approach" by Udi Manber.

Introduction to Algorithms by Udi Manber http://vig-fp.prenhall.com/bigcovers/0201120372.jpg

The first two will teach lots of programming techniques, patterns, and really open up one's mind to his/her own creativity; the other two are different. They're more theoretical, but also very important, focusing on design of correct and efficient algorithms (and requiring substantially more math).

I see lots of people recommending the three first books when the subject of "good programming books" pops up, but the last one (by Manber) is a great book, and few people know it. It's a shame! Manber focuses on the incremental development of algorithms through theorem proving using induction.

Share

answered Oct 18, 2008 at 15:39

community wiki
sergio_petralia

-1 Pick one, stick in a pic and let people vote on it
– Ruben Bartelink Mar 27, 2010 at 13:51

@sergio: Agreed about Manber's book. Please do what Ruben says and I'll upvote. Let me know if you decide to do that.
– Neil G Jul 6, 2011 at 1:19

Prev 1 … 4 5 6 7 8 Next