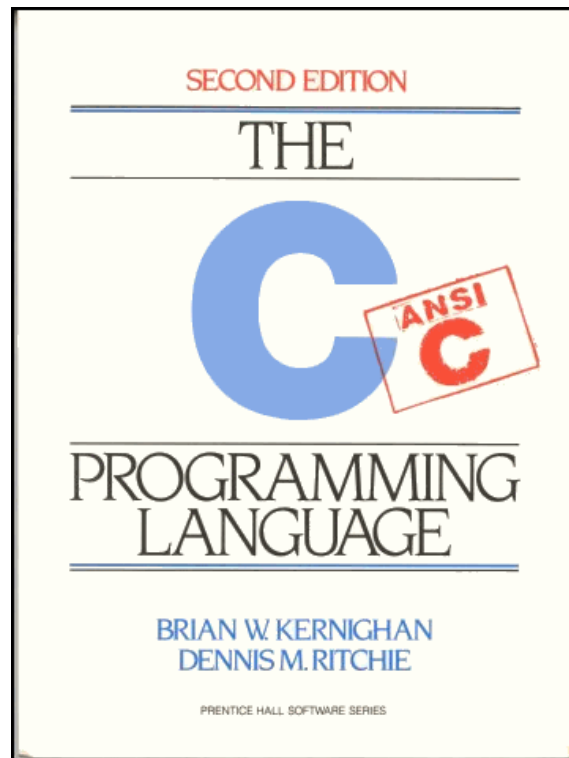


Zero Equals False

2017-11-29 Marty Jacobs

Learn C programming and the rest will come



When it comes to computers and tech, the general consensus is that “Newer is better.” A core i series processor is better than a Core 2 Duo, and iPhone 10 is better than an iPhone 6. So when the fact that one language has held its place in the list of most used programming languages for over three decades now is, in many ways, surprising. C is a concise language, and in its 45 years of existence, it has cemented its place as an essential tool in any programmer’s arsenal. But why?

What’s so good about it? Well.. **C is fast.** C was built to write code close to machine level, and that makes it one of, if not the fastest language you can use in many, many situations. Most of the critical parts of the OS you’re using right now were (far more likely than not) written in C, and when it comes to, say, a chip in a refrigerator, you simply cannot use modern languages like Java or Python. While that code is often also written in Assembly, C gives you a good level of abstraction and trades of just a little bit of speed. This is what makes C so useful in Electronics Engineering, and until a modern language like Rust can demonstrate the utility and robustness required in such applications it’s going to stay there.

Also, **C is flexible.** You can do more with C than you can do with newer languages. Most newer languages don’t even allow usage of pointers, a tool that’s integral to nearly every C program written. Pointers, while hard to understand, are extremely powerful and give you greater control over the memory of your program. The tradeoff here is that C expects you to know what you’re doing. If you don’t use your pointers or memory properly, you can run into memory leaks, segmentation faults,

and many other errors that can corrupt the memory of your program/computer. This is the reason why so many languages ditch pointers, and why you should be extra-careful when using these tools.

And that's good. Because it teaches you something really important: **plan your programs out**. With C code, there's no going into a program without fully knowing what it's going to do at every step of the way. And to be good at C, you need to master the fine details of what your program's doing, and even understand how much memory it's using and where. This way, you and only you are in control of what's happening, which is the most important thing for a programmer.

Finally, it's worth mentioning that learning C won't be a complete waste of time if you don't use it because it is **transferable knowledge** (so you might use it for something else!). C has been so popular that it has inspired a whole host of languages (dubbed "C-Style languages"), which use its syntactical conventions. Have you heard of C++, Java and C#? these are frequently used languages, all inspired by C. By learning C, you gain a basic understanding of the flow of these languages and how they run, though all of them bring some or the other changes which make them unique. So, if you're interested in programming, C is a great place to start.

Thanks for reading! Subscribe below for a free gift.

Subscribe

OLDER

Google's Kubernetes - Natively supported on the Docker platform

Search

ZERO EQUALS FALSE

