Introduction to C

CSC100 Introduction to programming in C/C++ Spring 2023

Marcus Birkenkrahe

January 27, 2023

Contents

1 What will you learn?

- What is C?
- What is its origin?
- What is its importance?
- What's the difference to C++?
- Why are we not just learning C++?
- What are C's strengths and weaknesses?
- Why are you learning C from me?
- What's next in the course?

Source:

- Textbook King (2008) ch.1[1]
- See also slides (GDrive)

[1] All sources are referenced at the end of the script, followed by the footnotes, which do unfortunately not render as links on GitHub. The book by King (2008) does not cover a few recent updates to the ANSI standard for C, like C11, and the current standard C17. The next major C standard revision (C23) is expected for 2023. Gustedt (2019) is a good (but quite difficult) book on "modern C".

2 What is C?

- C is a programming language created in the early 1970s.
- It grew out of the development of the UNIX operating system
- In turn, UNIX grew out of a space travel game (Brock, 2019).

Figure 1: Thompson & Ritchie & DEC PDP-11, 1970. (Brock, 2019)

3 How popular is C?

- C consistently ranks among the top 3 programming languages.
- TIOBE Language of the year 2008, 2017, 2019

- Popularity contest: cp. TIOBE Index[2]

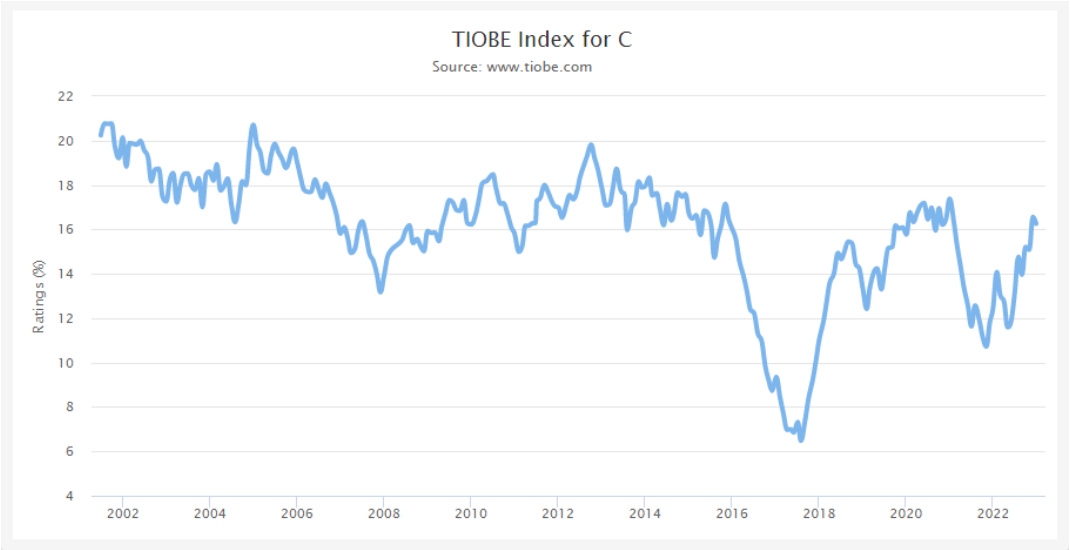


Figure 2: TIOBE Index for C, 2002-2023 (Source: TIOBE)

Jan 2023	Jan 2022	Change	Programming Language	Ratings	Change
1	1		Python	16.36%	+2.78%
2	2		C	16.26%	+3.82%
3	4	▲	C++	12.91%	+4.62%
4	3	▼	Java	12.21%	+1.55%
5	5		C#	5.73%	+0.05%
6	6		Visual Basic	4.64%	-0.10%
7	7		JavaScript	2.87%	+0.78%
8	9	▲	SQL	2.50%	+0.70%
9	8	▼	Assembly language	1.60%	-0.25%
10	11	▲	PHP	1.39%	-0.00%

Figure 3: TIOBE Index ranking 1-10 (tiobe.com)

_+

- Of the top 10 languages, 7 are direct descendants of C - only Visual Basic, SQL and Assembly language are not C-type languages

4 How important is C?

Some well-known programs written in C:

- The Linux kernel (and therefore, Android)
- UNIX operating system (core of MacOS)
- Windows 1.0 to Windows XP
- Doom (early video game)
- Wolfenstein 3D
- Git version control system
- C compilers (Clang, GCC/MinGW)
- Any software that crosses platforms easily (portable)

See also: "Why C programming is awesome" (Hawkes, 2016).

5 What is a programming language?

> "A programming language is a *formal language* comprising of a set of *strings* that produce various kinds of *machine code* output. Programming languages are one kind of computer language, and are used in computer programming to implement algorithms." (Source: Wikipedia)

- **Formal** language?
- Set of **strings**?
- **Machine** code?

[2]Since 2000, C has consistently ranked among the top two languages in the TIOBE index (based on searches).

Figure 4: IEEE index (Source: Cass, 2019)

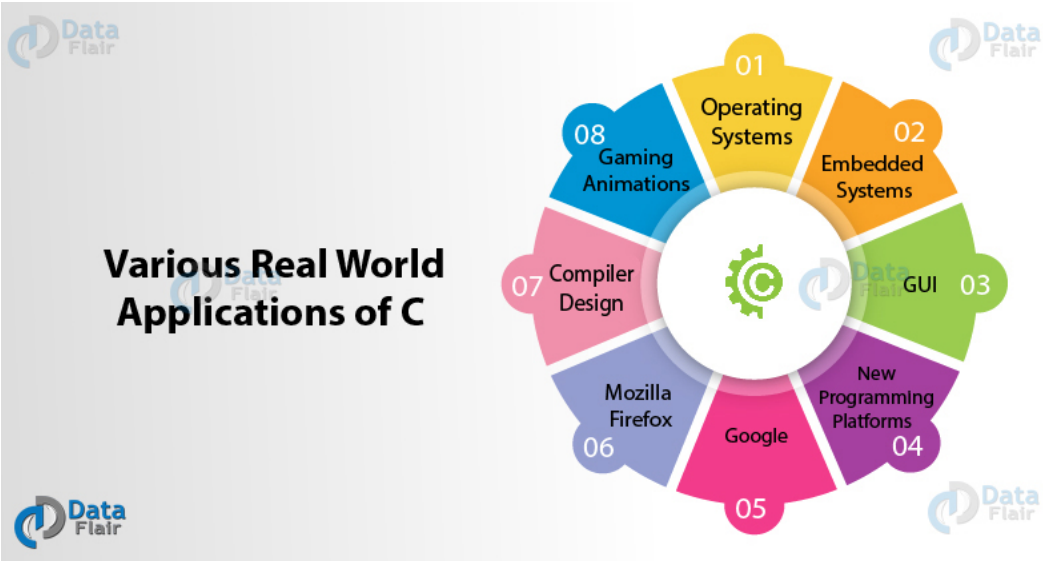


Figure 5: Real world applications of C (Source: DataFlair)

- **Algorithm**?

Fortunately, computer (and data) science isn't really a science at all! (it's more of a craft like mining, knitting, or pottering.)

6 Where does C come from?

- By-product of the UNIX operating system 1969 [3]
- Developed on DEC PDP-7 (computer with 8K words of main memory)
- Written originally in assembly language
- UNIX rewritten in C by 1973 for DEC PDP-11
- Standardization of C, 1973-2018

[3] The motivation to create Unix, according to Wikipedia, was to port Thompson's space travel video game to the PDP-7 mainframe computer. So in a way we owe modern computing to gaming.

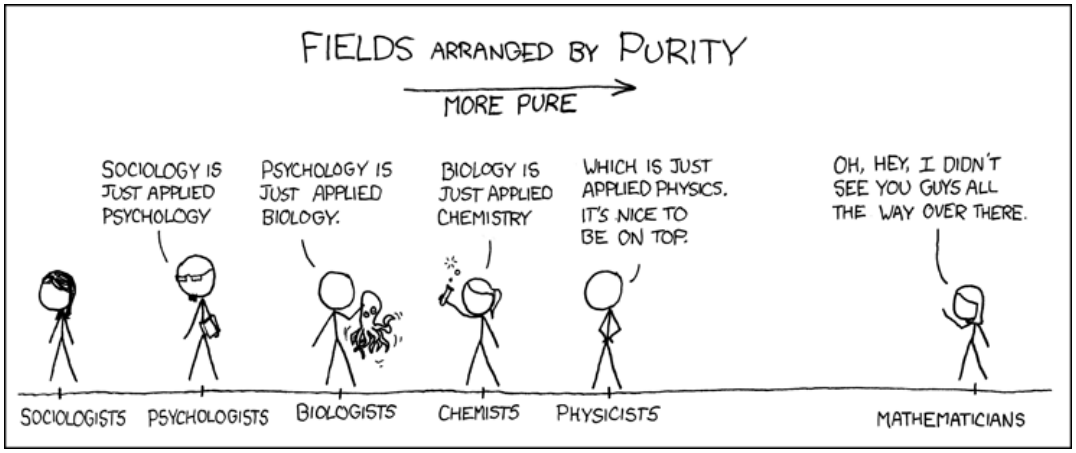


Figure 6: Scientific fields arranged by "purity" (xkcd)

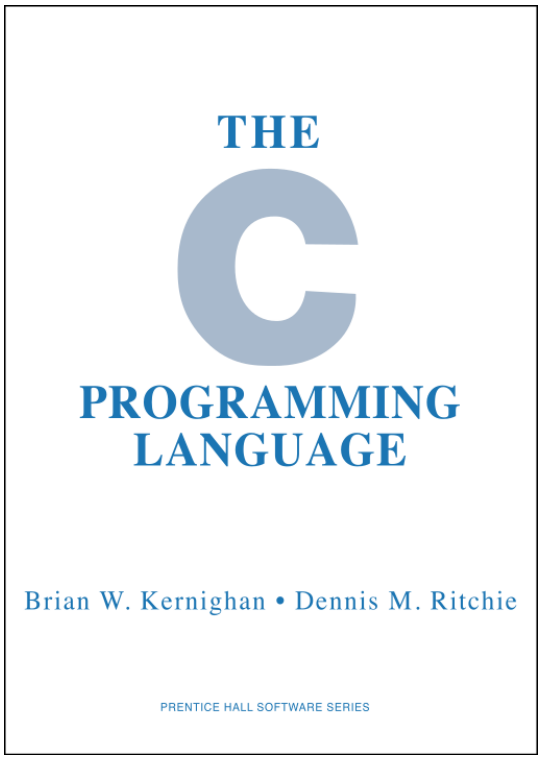


Figure 7: "K&R" (Kernighan/Ritchie, 1978)

Challenge: what does "8K words of main memory" mean? [4]

See also: C Programming Language | Brian Kernighan & Lex Fridman

- Text processing problems were inherited from Unix
- Examples should be realistic, useful and representative
- If you're the first in anything, everybody else has to follow

7 Standardization

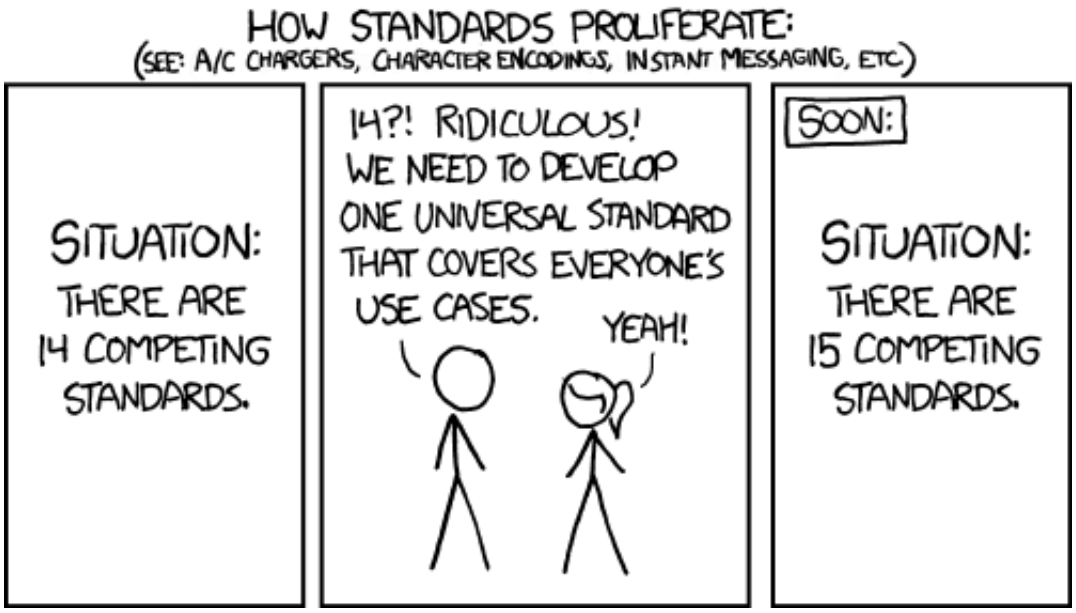


Figure 8: How standards proliferate (Source: xkcd)

Details: see ANSI (American National Standards Institute)
Python 3 was not "backwards compatible" with Python 2.7

[4] How many bits can be stored in memory of 8K words depends on the bit length of a word (or byte). One byte holds $8 = 2^3$ bits (binary digits, or memory locations capable of storing 2 states). 8K byte correspond to $8 * 2^{10} = 8 * 1{,}024 = 8{,}192$ bits. By comparison, the main memory of my laptop is $16\text{GB} = 16 * 2^{30} = 3.2\text{E}{+}31$ bits. It follows from these memory restrictions that UNIX (and C) had to be designed to be very small, or space effective.

8 How computers work

Well, at least this is one way of looking at it.

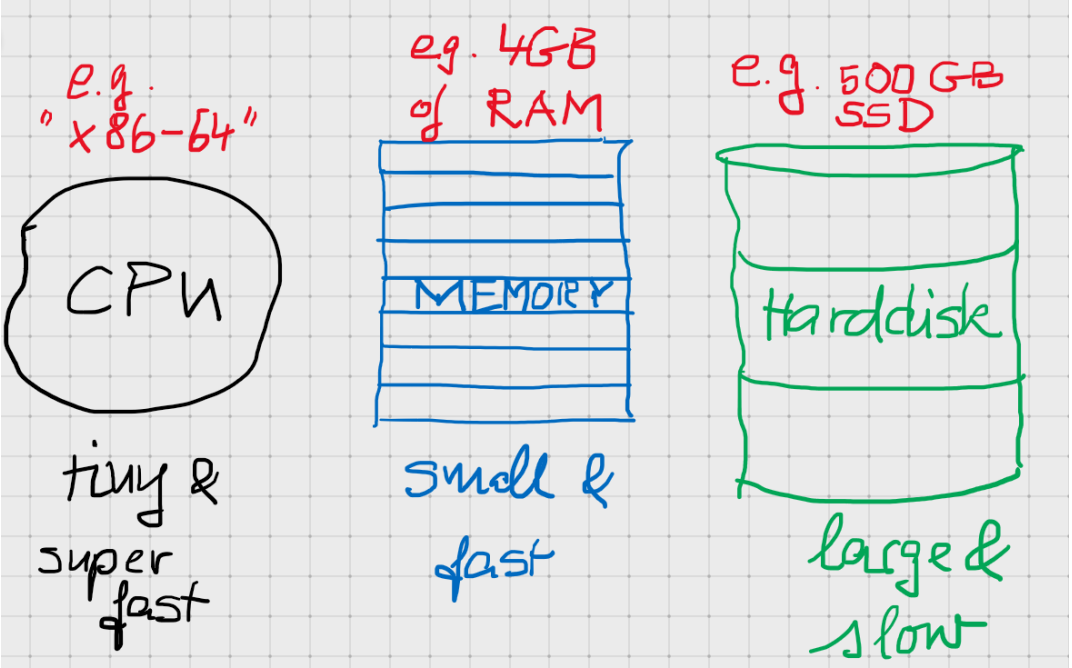


Figure 9: Computer architecture (simplified)

The "hard disk" can also be a Solid State Drive (SSD) or some other form of Non-Volatile Memory (NVM) - i.e. it doesn't disappear when the power goes out.

9 How programs are processed

9.1 Simplified process

1. **WRITE** source code in an editor (NVM = harddisk)
2. **COMPILE** source code to machine code (RAM = memory)
3. **RUN** program (CPU = Central Processing Unit)
4. **DISPLAY** results (RAM = Memory)
5. **SAVE** result (NVM = harddisk)

9.2 Complete process

Specifically for C and our compiler GCC, this process looks technically like this:

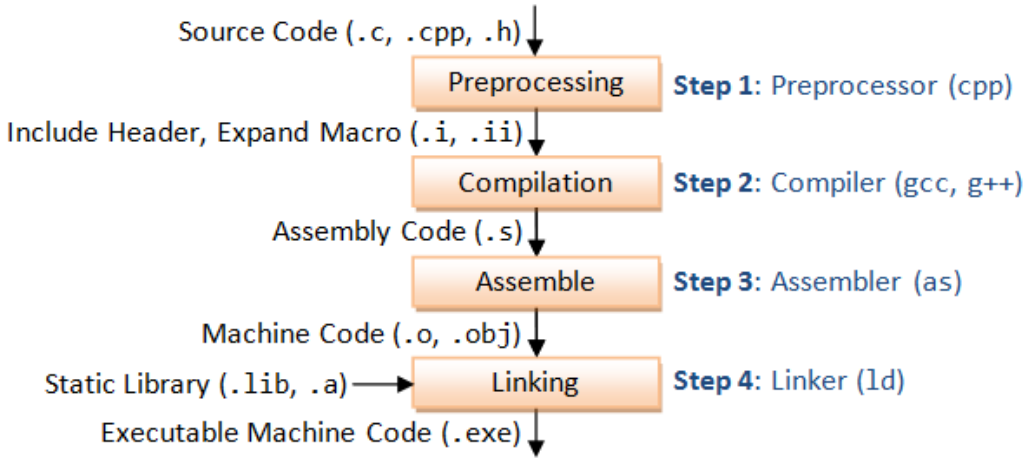


Figure 10: GCC compilation process (Source: Hock-Chuan, 2018).

10 Looks matter

She's a beauty.

Challenge: was C the first programming language? [5]

11 The Latin of programming languages

The following languages are C-based:

- C++ (OOP extension of C, games)
- Java (OOP, VM-portable, and no pointers)

[5] Answer: no. By 1966, there were already ca. 700 programming languages (Chatley et al, 2019), today there are almost 9,000. C descends from ALGOL60, other important languages are Lisp (functional language), SIMULA (first OOP language), and PROLOG (logic language).

Figure 11: PDP-11/70

- C# (Microsoft, games)
- Perl (scripting, text mining)

  "C is to programming languages what Latin is to Western natural languages." (Anonymous)

- C is lightning fast and terribly tiny

  "C is fast because it's the speed of light, and relativity?"[6] (Stackoverflow)

12 Benchmarking

Language	Time, s	Memory, MiB	Energy, J
C++/g++	$11.561_{\pm 0.257}$	$2.53_{\pm 00.89}$ + $1.37_{\pm 00.86}$	$198.02_{\pm 10.79}$
C/gcc	$12.638_{\pm 0.191}$	$0.63_{\pm 00.00}$ + $1.00_{\pm 00.04}$	$271.04_{\pm 04.12}$
Rust	$13.468_{\pm 0.603}$	$1.99_{\pm 00.07}$ + $0.31_{\pm 00.03}$	$260.96_{\pm 26.40}$
C/clang	$14.006_{\pm 0.558}$	$0.69_{\pm 00.00}$ + $0.95_{\pm 00.03}$	$277.62_{\pm 28.40}$

Figure 12: Image source: Kostya benchmark, GitHub, Nov 2021

- Excerpts for parsing and printing a *Mandelbrot* set
- *Benchmarks* depend on algorithm implementation
- C always wins the size battle (*memory allocation*)
- Some very *specialized* languages are even faster

[6] This is a joke based on someone mixing up c (speed of light constant) and C (the programming language).

Python/pypy	$65.668_{\pm 2.354}$	$63.37_{\pm 00.05}$ + $47.76_{\pm 00.05}$	$1333.51_{\pm 81.90}$
Julia	$75.405_{\pm 1.755}$	$200.84_{\pm 00.22}$ + $0.61_{\pm 00.00}$	$1569.12_{\pm 122.00}$
Ruby/truffleruby (--jvm)	$120.743_{\pm 5.839}$	$581.66_{\pm 04.97}$ + $539.81_{\pm 13.64}$	$2466.87_{\pm 108.51}$
Ruby/truffleruby	$131.313_{\pm 4.944}$	$445.97_{\pm 01.32}$ + $574.91_{\pm 13.29}$	$2693.24_{\pm 211.06}$
Haskell	$220.958_{\pm 4.270}$	$3.81_{\pm 00.02}$ + $26.13_{\pm 00.00}$	$4785.77_{\pm 295.26}$

Figure 13: Image source: Kostya benchmark, GitHub, Nov 2021

13 Strengths and weaknesses of C

STRENGTH	WEAKNESS
Efficiency	Permissiveness (Error-prone)
Portability	Terseness and Understanding
Power	Large program maintenance
Flexibility	
Standard library	
Integration with UNIX	

- Efficiency: do a lot with little effort (small programs)
- Portability: it works everywhere, on anything
- "Power": you can do brain surgery with a pencil
- Flexbility: you can do the same thing in many different ways
- "Standard library": pre-defined functions/tasks; "stdio.h", a standard library for "I/O" (Input/output)
- Integration with UNIX (because UNIX is the motherlobe)

14 What is the difference between C and C++?

C++ is a superset of C.

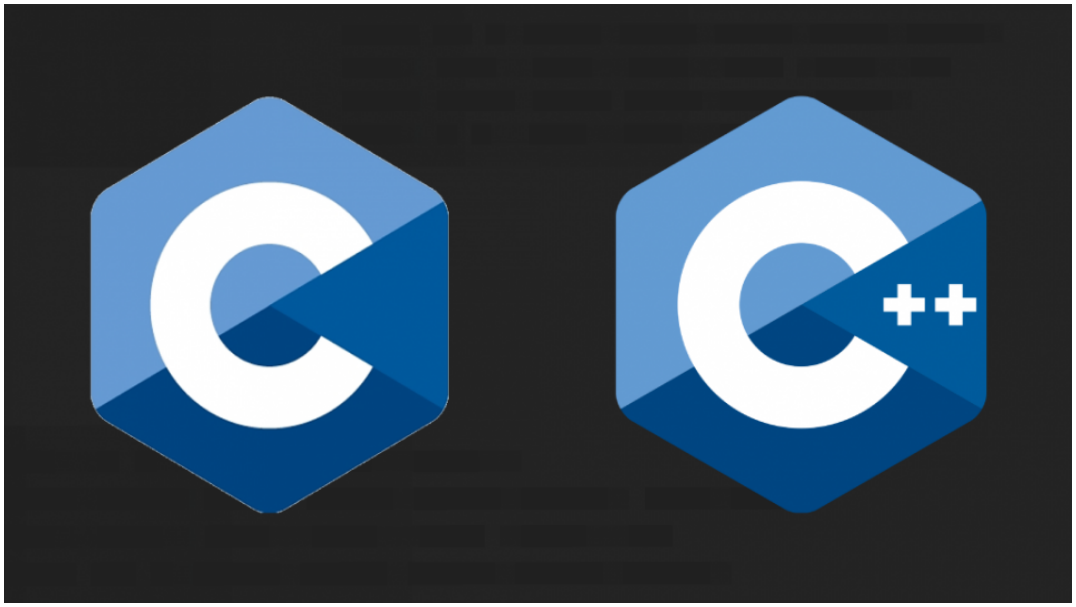


Figure 14: C/C++ logos

WHAT	C	C++
TIME	Thompson/Ritchie 1970s	Stroustrup 1980s
TYPE	Imperative procedural	Object-oriented
GOOD	System programming	Games and graphics
USED	Internet of Things	Flight Software

Source: Lemonaki, 2021.

15 Why are we not just learning C++?

- Object-orientation is a difficult paradigm (C++)
- System programming is pure power (C)
- C is simpler, smaller, and faster

- Bjarne Stroustrup (2011): "C is obsolete"[7]

[7]However, he is biased, since he is the creator of C++. The title of the video is misleading: Stroustrup believes that every C program should rather be a proper C++ program. However, he also concedes that C++ is still too complex for many ("We have to clean it up").

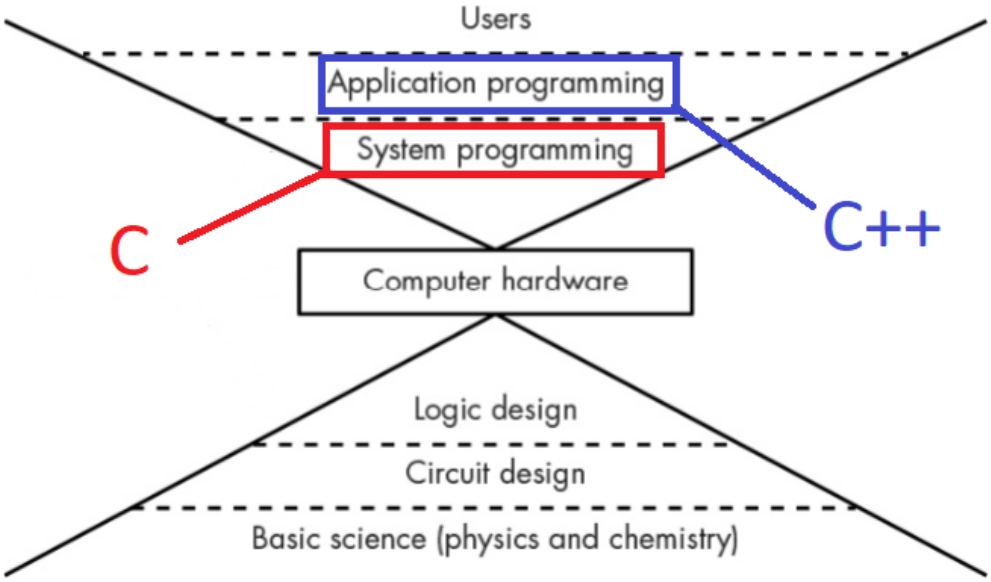


Figure 15: Computer Landscape. (Modified from: Steinhart, 2019)

- Linus Torvalds (2007): "C++ is a horrible language"[8].

  Also, there's this:

  > "Languages are tools. Memorizing them no more makes you a computer scientist than studying hammers makes you a carpenter." -Neilsen

  - It's easy to pick up additional languages
  - Data structures and algorithms are key to understanding
  - First language could be anything[9]

[8] Torvalds (who wrote the Linux kernel in C) argues here in favor of writing his hugely successful version control program `git` in C instead of C++. He highlights some of the strengths of C: efficient, system-level, portable code.

[9] My first real programming language was FORTRAN (specialized on scientific computing), then C++. Recently, I picked up R (for data science). In between I've sampled (not mastered) many others, including: Python, Lisp, PROLOG, C, PHP, SQL, SQLite etc.

16 Why am I teaching C/C++?

It's personal.

> I used C++ during my PhD studies at DESY, Germany, to write a library of multigrid functions (numerical method for lattice gauge theory simulations in theoretical particle physics).[10]

17 What will happen to C/C++ in the next 20 years?

Whatever happens, good new for learning C.

I increasingly see propaganda for replacing C++ by Rust (Kirsh, 2021), another relatively new language with OOP support and better security properties.

18 Summary

1. The C programming language was created 50 years ago
2. C is small, simple, very fast, and close to the computer
3. Linux (and Android) are largely written in C
4. The object-oriented programming (OOP) language C++ contains C
5. System programming is a powerful skill set

19 Glossary

[10] I changed my name from 'Speh' to 'Birkenkrahe' when I got married.

The C++ Virtual Library

Recent Changes -- Mail additions to this list to <lilje@desy.de>
You can also *search for a keyword on this server.*

The HTML formatted version on the draft C++ standard.

Getting Start(l)ed
: Documents and sources on C++ and OOP. The draft C++ standard is now generally available. Please read the informal announcement.

Editing
: Customizable environment for Emacs editors

Learning C++
: Virtual courses and tutorials.

Newsgroups
: Internet groups for discussions and questions on C++

Free Packages
: Freely available C++ packages from various application areas.

Conferences
: List of OOP and Computing conferences

FreeHEP
: software information and reviews from freeHEP [FAQ]

OOLP
: Discussion on Object-Oriented Literate Programming

Tools & Products
: The C++ Products List and Description. See also: the Darmstadt archive

General OO
: Object-Oriented programming resources.

To report errors, use this form to contact Lutz, please.

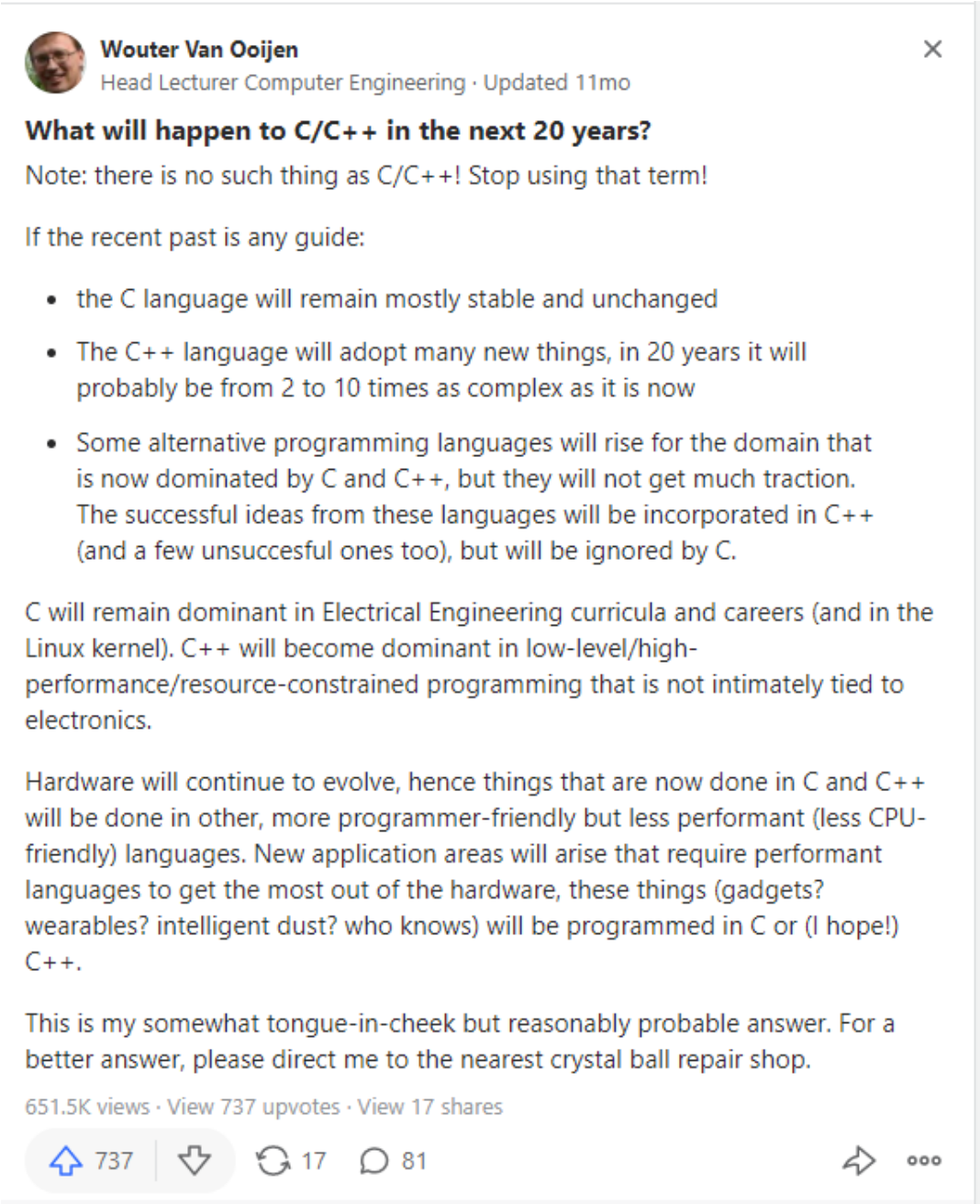
Last updated October 25th, 1995
Lutz Lilje

<lilje@desy.de>

Big thanks to Marcus for starting up all this!
Info about Marcus Speh

<marcus@x4u.desy.de>

Figure 16: The C++ Virtual Library, 1993-1995 (DESY)



Wouter Van Ooijen
Head Lecturer Computer Engineering · Updated 11mo

What will happen to C/C++ in the next 20 years?

Note: there is no such thing as C/C++! Stop using that term!

If the recent past is any guide:

- the C language will remain mostly stable and unchanged
- The C++ language will adopt many new things, in 20 years it will probably be from 2 to 10 times as complex as it is now
- Some alternative programming languages will rise for the domain that is now dominated by C and C++, but they will not get much traction. The successful ideas from these languages will be incorporated in C++ (and a few unsuccesful ones too), but will be ignored by C.

C will remain dominant in Electrical Engineering curricula and careers (and in the Linux kernel). C++ will become dominant in low-level/high-performance/resource-constrained programming that is not intimately tied to electronics.

Hardware will continue to evolve, hence things that are now done in C and C++ will be done in other, more programmer-friendly but less performant (less CPU-friendly) languages. New application areas will arise that require performant languages to get the most out of the hardware, these things (gadgets? wearables? intelligent dust? who knows) will be programmed in C or (I hope!) C++.

This is my somewhat tongue-in-cheek but reasonably probable answer. For a better answer, please direct me to the nearest crystal ball repair shop.

651.5K views · View 737 upvotes · View 17 shares

737 17 81

Figure 17: One expert's opinion (Source: Quora)

CONCEPT/TOPIC	DEFINITION
DEC PDP-11	1970s mainframe computer
UNIX	Operating system (ca. 1969)
ANSI	American National Standard Institute
String	A data type representing text
Assembler	Machine code (hard to write/read)
Algorithm	Fixed process or set of rules
Linux	Operating system (ca. 1991)
C	Imperative, procedural programming language
compiler	Software to translate source into machine code
C++	Object-oriented (OO) superset of C
Clang	C/C++ compiler
gcc	GNU compiler bundle (incl. C/C++)
Java,C#	OO programming language
Perl	Scripting language
Git	Software version control system
GitHub	Developer's platform (owned by Microsoft)
Library	Bundle of useful functions and routines
Portability	Ability of software to run on different hardwares
Efficiency	Software speed of execution and memory requirements
Permissiveness	Degree to which a language tolerates ambiguities
Object-orientation	Ability to define abstractions
System programming	Programming close to the machine
Application programming	Programming close to the user

20 What's next?

- Getting started: Infrastructure (Lab)
- MinGW (compiler) + Emacs (editor) + GitHub (collaboration)
- First program: "hello world" (Lecture + Lab)

21 References

- Big Think (Jun 13, 2011). Bjarne Stroustrup: Why the Programming Language C Is Obsolete | Big Think [video]. URL:youtu.be/KlPC3O1DVcg.
- Brock (October 17, 2019). The Earliest Unix Code: An Anniversary Source Code Release [Blog]. URL: computerhistory.org.
- Cass (6 Sept 2019). The Top Programming Languages 2019 > Python remains the big kahuna, but specialist languages hold their own. IEEE Spectrum. URL: spectrum.ieee.org.
- Chatley R., Donaldson A., Mycroft A. (2019) The Next 7000 Programming Languages. In: Steffen B., Woeginger G. (eds) Computing and Software Science. Lecture Notes in Computer Science, vol 10000. Springer, Cham. `https://doi.org/10.1007/978-3-319-91908-9_15`
- Data Flair (n.d.). Applications of C Programming That Will Make You Fall In Love With C [Tutorial]. URL: data-flair.training.
- DESY (Oct 25, 1995). The C++ Virtual Library. URL: desy.de
- Gustedt (2019). Modern C. Manning.

- Hock-Chuan (2018). GCC and Make: Compiling, Linking and Building C/C++ Applications [website]. URL: ntu.edu.sg.
- Kernighan/Ritchie (1978). The C Programming Language. Prentice Hall. Online: wikipedia.org.
- King (2008). C Programming - A Modern Approach. Norton. Online:knking.com.
- Kirsh (September 13, 2021). Rust vs C++ and Is It Good for Enterprise? [blog]. URL: www.incredibuild.com.
- Lemonaki, Dionysia (November 4, 2021). C vs. C++ - What's The Difference [blog]. URL: freecodecamp.org.
- Neilsen (Aug 14, 2020). Quora. URL: qr.ae/pGzZ9z.
- Steinhart (2019). The Secret Life of Programs. NoStarch Press. URL:nostarch.com.
- TIOBE (Jan 2022). TIOBE Index for January 2022 [website]. URL:tiobe.com.
- Torvalds (6 Sep 2007). Linus Torvalds on C++ [blog]. URL:harmful.cat-v.org.
- xkcd(n.d.) Purity [cartoon]. URL: xkcd.com/.