Programming in C

Dr. Neill Campbell
Neill.Campbell@bristol.ac.uk

University of Bristol

June 16, 2021

Table of Contents

Preamble

Hello World!

Grammar

Flow Control

Functions

About the Course

These course notes were originally based on:

C By Dissection (3rd edition)
Al Kelley and Ira Pohl

because I liked arrays being taught late(r). I've since changed my mind a little & have re-jigged the notes quite heavily for this year.

Resources

- Free : `https://en.wikibooks.org/wiki/C_Programming`

Resources

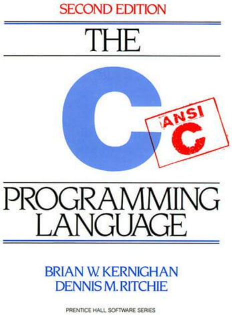
- Free : `https://en.wikibooks.org/wiki/C_Programming`
- A list of more : `https://www.linuxlinks.com/excellent-free-books-learn-c/`

Resources

- Free : `https://en.wikibooks.org/wiki/C_Programming`
- A list of more : `https://www.linuxlinks.com/excellent-free-books-learn-c/`
- Whatever you use, make sure it's **ANSI C** or **C99** that's being taught, not something else e.g. C11 or C++.

Resources

- Free : `https://en.wikibooks.org/wiki/C_Programming`
- A list of more : `https://www.linuxlinks.com/excellent-free-books-learn-c/`
- Whatever you use, make sure it's **ANSI C** or **C99** that's being taught, not something else e.g. C11 or C++.
- If you fall in love with C and know you're going to use it for the rest of your life, the reference 'bible' is K&R 2nd edition. It's not a textbook for those new to programming, though.



Computer Science Ethos

- Talk to your friends, ask for help, work together.

Computer Science Ethos

- Talk to your friends, ask for help, work together.
- Never pass off another persons work as your own.

Computer Science Ethos

- Talk to your friends, ask for help, work together.
- Never pass off another persons work as your own.
- Do not pass work to others - either on paper or electronically - even after the submission deadline.

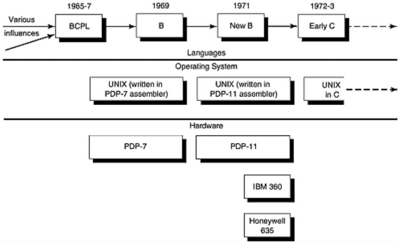
Computer Science Ethos

- Talk to your friends, ask for help, work together.
- Never pass off another persons work as your own.
- Do not pass work to others - either on paper or electronically - even after the submission deadline.
- If someone takes your code and submits it, we need to investigate where it originated - all students involved will be part of this.

Computer Science Ethos

- Talk to your friends, ask for help, work together.
- Never pass off another persons work as your own.
- Do not pass work to others - either on paper or electronically - even after the submission deadline.
- If someone takes your code and submits it, we need to investigate where it originated - all students involved will be part of this.
- Don't place your code on publicly accessible sites e.g. github - other students may have extensions etc.

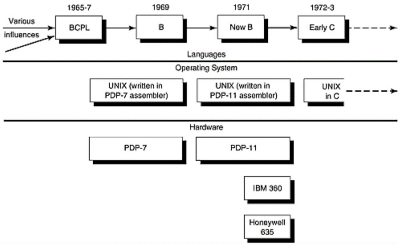
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards

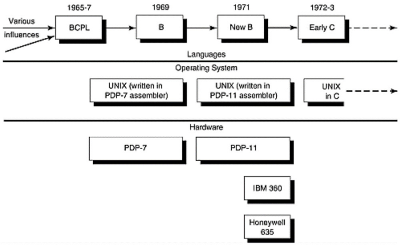
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970

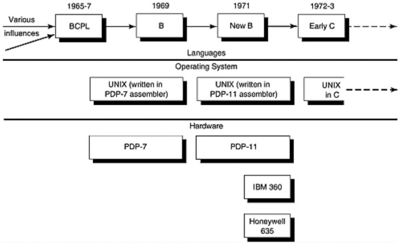
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.

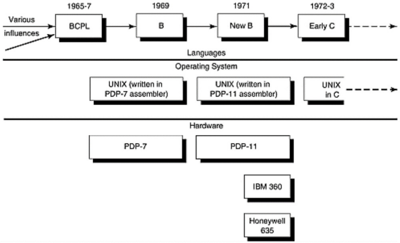
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.
- C - Dennis Ritchie 1972 designed for (& implemented on) a UNIX system.

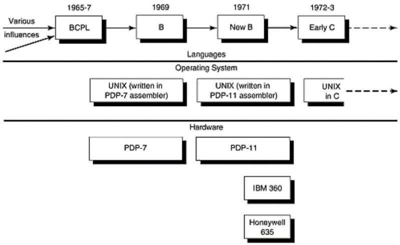
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.
- C - Dennis Ritchie 1972 designed for (& implemented on) a UNIX system.
- K&R C (Kernighan and Ritchie) 1978

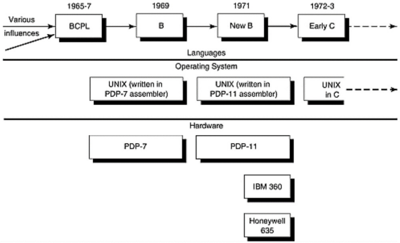
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.
- C - Dennis Ritchie 1972 designed for (& implemented on) a UNIX system.
- K&R C (Kernighan and Ritchie) 1978
- ANSI C

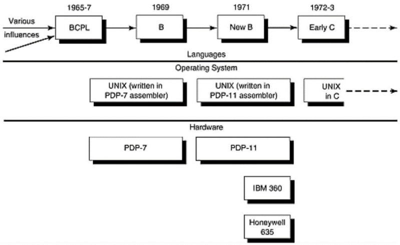
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.
- C - Dennis Ritchie 1972 designed for (& implemented on) a UNIX system.
- K&R C (Kernighan and Ritchie) 1978
- ANSI C
- C99 (COMSM1201)

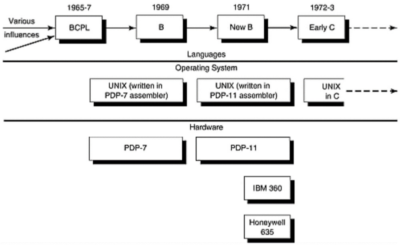
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.
- C - Dennis Ritchie 1972 designed for (& implemented on) a UNIX system.
- K&R C (Kernighan and Ritchie) 1978
- ANSI C
- C99 (COMSM1201)
- C++ - Object Oriented Programming (OOP)

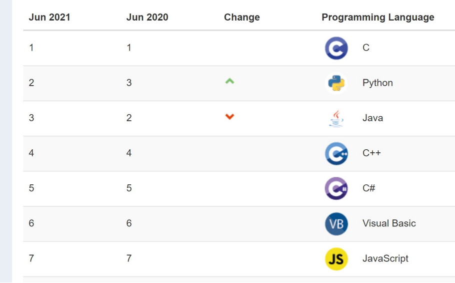
History of C



From **Deep C Secrets** by *Peter Van Der Linden*

- BCPL - Martin Richards
- B - Ken Thomson 1970
- Both of above are *typeless*.
- C - Dennis Ritchie 1972 designed for (& implemented on) a UNIX system.
- K&R C (Kernighan and Ritchie) 1978
- ANSI C
- C99 (COMSM1201)
- C++ - Object Oriented Programming (OOP)
- Java (Subset of C++, WWW enabled).

Why C ?

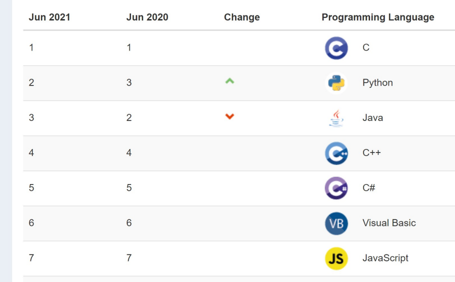


Jun 2021	Jun 2020	Change	Programming Language
1	1		C
2	3	▲	Python
3	2	▼	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript

`https://www.tiobe.com/tiobe-index/`

- One of the most commonly used programming languages according to `tiobe.com`

Why C ?

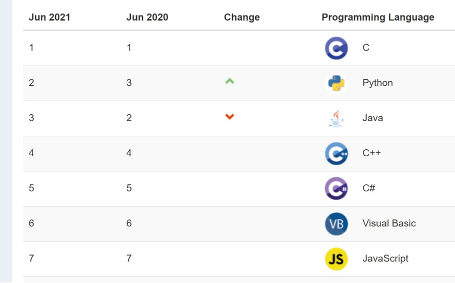


Jun 2021	Jun 2020	Change	Programming Language
1	1		C
2	3	▲	Python
3	2	▼	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript

`https://www.tiobe.com/tiobe-index/`

- One of the most commonly used programming languages according to `tiobe.com`
- Low-level (c.f. Java)

Why C ?

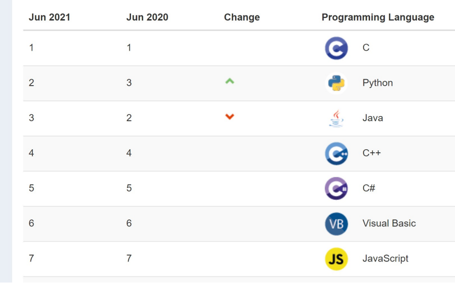


Jun 2021	Jun 2020	Change	Programming Language
1	1		C
2	3	▲	Python
3	2	▼	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript

`https://www.tiobe.com/tiobe-index/`

- One of the most commonly used programming languages according to `tiobe.com`
- Low-level (c.f. Java)
- Doesn't hide nitty-gritty

Why C ?

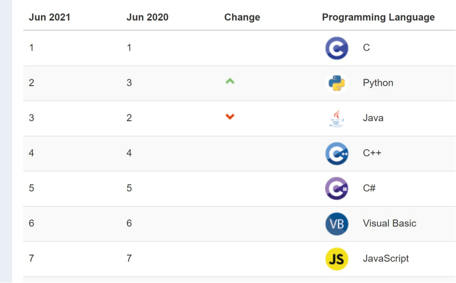


Jun 2021	Jun 2020	Change	Programming Language
1	1		C
2	3	▲	Python
3	2	▼	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript

`https://www.tiobe.com/tiobe-index/`

- One of the most commonly used programming languages according to `tiobe.com`
- Low-level (c.f. Java)
- Doesn't hide nitty-gritty
- Fast ?

Why C ?



Jun 2021	Jun 2020	Change	Programming Language
1	1		C
2	3	▲	Python
3	2	▼	Java
4	4		C++
5	5		C#
6	6		Visual Basic
7	7		JavaScript

`https://www.tiobe.com/tiobe-index/`

- One of the most commonly used programming languages according to `tiobe.com`
- Low-level (c.f. Java)
- Doesn't hide nitty-gritty
- Fast ?
- Large parts common to Java

Programming and Software Engineering

- Was traditionally Lectured 2(or 3) hours a week for weeks 1-12

Programming and Software Engineering

- Was traditionally Lectured 2(or 3) hours a week for weeks 1-12
- With COVID-19 I'll post the equivalent online, broken into manageable chunks

Programming and Software Engineering

- Was traditionally Lectured 2(or 3) hours a week for weeks 1-12
- With COVID-19 I'll post the equivalent online, broken into manageable chunks
- Programming (C), data structures, algorithms - searching, sorting, string processing, trees etc.

Assessment

- Weekly (unmarked) exercises that, if completed, should ensure you are able to pass the unit.

Assessment

- Weekly (unmarked) exercises that, if completed, should ensure you are able to pass the unit.
- Approximately three/four assignments and one lab test.

Assessment

- Weekly (unmarked) exercises that, if completed, should ensure you are able to pass the unit.
- Approximately three/four assignments and one lab test.
- One major project due in early TB2 (35%).

Assessment

- Weekly (unmarked) exercises that, if completed, should ensure you are able to pass the unit.
- Approximately three/four assignments and one lab test.
- One major project due in early TB2 (35%).
- Hard to gauge timings, so don't make any plans in advance - I'll change it if we're going too fast.

Table of Contents

Preamble

Hello World!

Grammar

Flow Control

Functions

Table of Contents

Preamble

Hello World!

Grammar

Flow Control

Functions

Table of Contents

Preamble

Hello World!

Grammar

Flow Control

Functions

Table of Contents

Preamble

Hello World!

Grammar

Flow Control

Functions