

last update: 26th January 2024

Session: 2023-24

COMP281 Lecture 1

Principles of C and Memory Management

Phil Jimmieson

Year 1 Modules

Introduction to Programming

Data Structures and Algorithms

Object-oriented Programming

Skill Set

- To solve problems using computational thinking;
- To write and apply pseudo code algorithms;
- To use appropriate data structures & algorithms;
- To debug programs and test software.

COMP281

Principles of C and
Memory Management

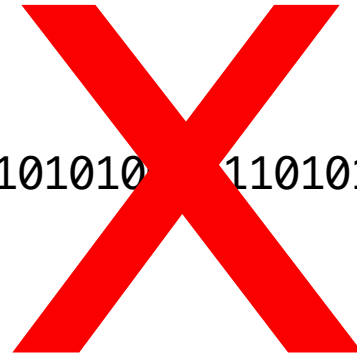
Today

- **COMP 281 Principles of C and Memory Management?**
what this module is about
- **General module information.**

What is C?

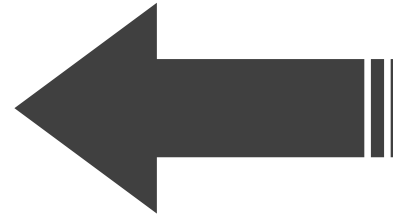


1010011010100111101010100011010011010101101010100011010011010101111110...





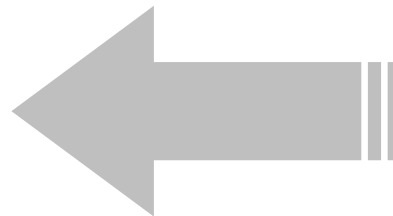
101001101010011110101010001...



Programming language

Abstracted away

101001101010011110101010001...



Programming language

The “Hello, World!” program

```
#include <stdio.h>
int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```

code

compile

output

Executable
Program

Edit -> compile -> run

- Make this into a file called `hello.c` using a text editor, e.g., vim
- Compile into a program and run:

```
% gcc hello.c
```

```
% ./a.out
```

```
Hello, world!
```

```
%
```



On Windows, the default executable program name is
a.exe
(Windows program names must end with .exe)

C programs are built up of functions

```
#include <stdio.h>

int main(void)
{
    printf("Hello, World!\n");
    return 0;
}
```

- Functions
 - Take in arguments
 - Compute something
 - Return a result
- The `main()` function
 - is where program execution starts

Why do we study C?

C History



<u>Year</u>		<u>Designed by</u>
1960	ALGOL	International Group
1967	BCPL	Martin Richards
1970	B	Ken Thompson
1972	Traditional C	Dennis Ritchie
1978	K&R C	Brian Kernighan & Dennis Ritchie
1989/1990	C89/C90	ANSI&ISO Committee
1999	C99	ISO Committee
2011	C11	ISO Committee
2018	C17	ISO Committee

C powers the World – Operating Systems and Databases



Windows / Windows Phone



macOS / iOS



Android



Linux



Oracle Database



MS SQL Server



MySQL



PostgreSQL

C powers the World – Embedded Systems



Plane



Car



Microwave



Dishwasher



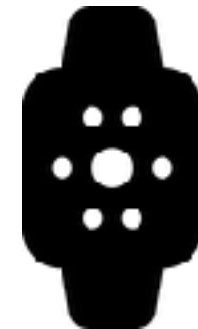
Drone



Vacuum Robot



Coffee Machine



Smart Watch

C Advantages – Portability and Efficiency

- C is almost a ***portable assembly language***. It is as close to the machine as possible while it is almost universally available for existing ***processor architectures***.
- Compilers, libraries, and interpreters of other programming languages are often ***implemented in C***. Interpreted languages like *Python*, *Ruby*, and *PHP* have their primary implementations written in C.

C Advantages – Portability and Efficiency

- C is almost a ***portable assembly language***. It is as close to the machine as possible while it is almost universally available for existing ***processor architectures***.
- Compilers, libraries, and interpreters of other programming languages are often ***implemented in C***. Interpreted languages like *Python*, *Ruby*, and *PHP* have their primary implementations written in C.

C Advantages – System Resource Management

- Arbitrary memory address access and **pointer arithmetic**.
- **Deterministic usage** of resources that fit for resource-limited systems.
- C has a very **small runtime**. The memory footprint for its code is smaller than for most other languages, e.g., C++.

C Advantages – Lingua Franca

- Many implementations of **new algorithms** in books are first (or only) made available in C by their authors.
- C is an old and widespread language – it's easy to find **all kinds of algorithms** written in C and lots and lots of examples.

Anything bad about C?

Disadvantages

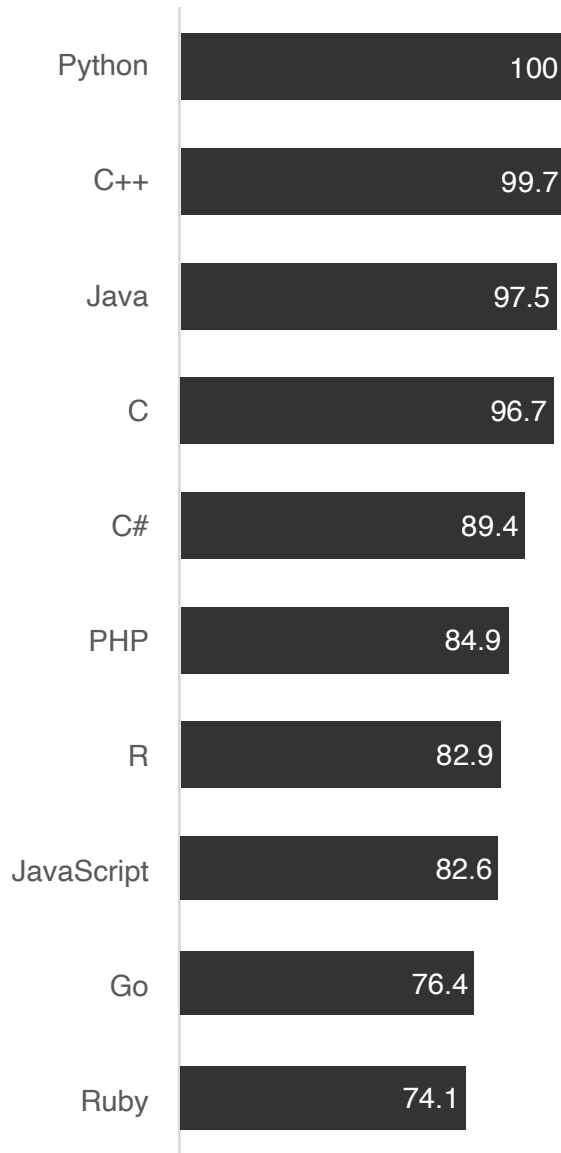
- No concept of Object Oriented Programming (OOP).
- No concept of Namespace.
- No Constructor or Destructor.
- Difficult to debug.
- Compilers cannot handle exceptions (run-time errors).
- No strict data type checking, e.g. an integer value can be passed for floating datatype).

Programming Language	2024	2023	2022	2021	2020	2019	2014	2009	2004
Python	1	1	1	3	3	4	7	5	10
C	2	2	2	1	2	2	1	2	2
C++	3	3	4	4	4	3	4	3	3
Java	4	4	3	2	1	1	2	1	1
C#	5	5	5	5	5	6	5	7	8
JavaScript	6	7	7	7	7	8	8	8	7
PHP	7	10	11	8	8	7	6	4	5
Visual Basic	8	6	6	6	6	5	11	-	-
SQL	9	8	9	12	10	9	-	-	6
Scratch	10	20	-	-	-	-	-	-	-
Go	11	12	-	-	-	-	-	-	-
Fortran	12	27	-	-	-	-	-	-	-
Swift	16	11	-	-	-	-	-	-	-

Programming Language	2024	2023	2022	2021	2020	2019	2014	2009	2004
Python	1	1	1	3	3	4	7	5	10
C	2	2	2	1	2	2	1	2	2
C++	3	3	4	4	4	3	4	3	3
Java	4	4	3	2	1	1	2	1	1
C#	5	5	5	5	5	6	5	7	8
JavaScript	6	7	7	7	7	8	8	8	7
PHP	7	10	11	8	8	7	6	4	5
Visual Basic	8	6	6	6	6	5	11	-	-
SQL	9	8	9	12	10	9	-	-	6
Scratch	10	20	-	-	-	-	-	-	-
Go	11	12	-	-	-	-	-	-	-
Fortran	12	27	-	-	-	-	-	-	-
Swift	16	11	-	-	-	-	-	-	-

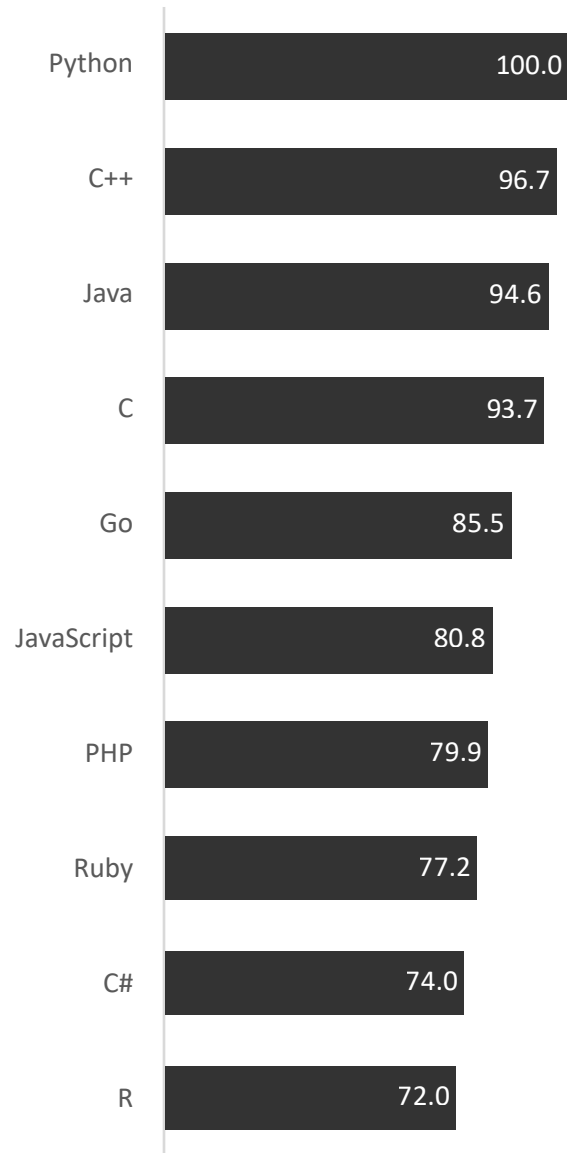
IEEE Spectrum

overall ranking



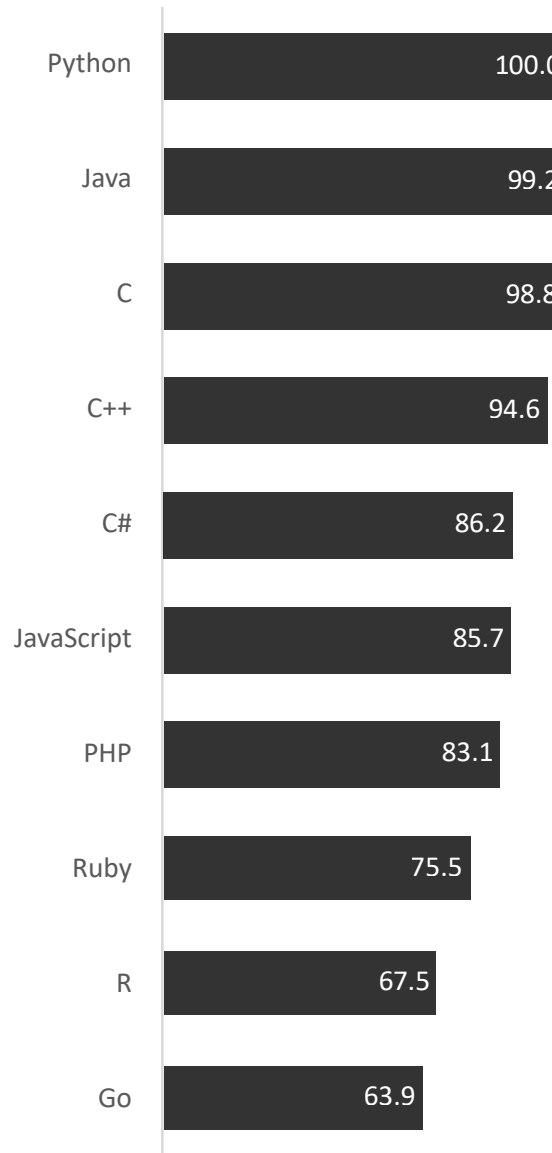
Trending

growing rapidly



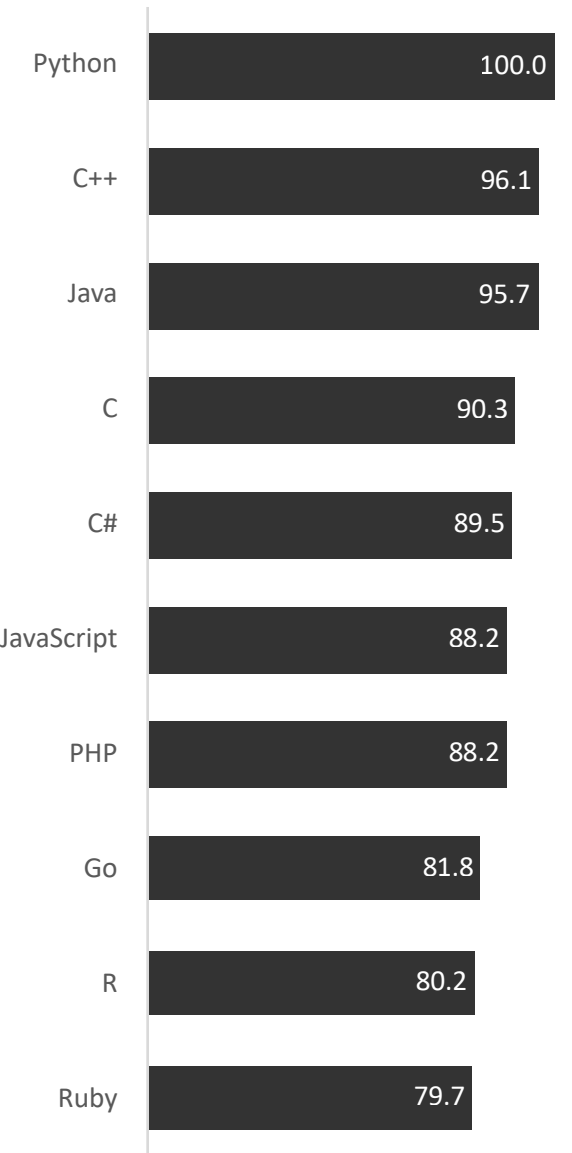
Jobs

in demand by employers



Open

popular on open source hubs



What Are the Greenest Programming Languages?

When we choose programming languages, we usually decide based on their syntax, performances, or even learning curve. Yet, a few years ago a group of Portuguese researchers investigated the energy consumption of **27 most popular programming languages, measuring execution time, energy consumption, and peak memory use.**

Here is the abstract of the paper:

“This paper presents a study of the runtime, memory usage, and energy consumption of twenty-seven well-known software languages. We monitor the performance of such languages using ten different programming problems, expressed in each of the languages. Our results show interesting findings, such as slower/faster languages consuming less/more energy, and how memory usage influences energy consumption. We show how to use our results to provide software engineers support to decide which language to use when energy efficiency is a concern.”

The results are the following:

C is the most efficient programming language, while Python and Perl are the least environmental friendly programming languages.

Authors: Rui Pereira, Marco Couto, Francisco Ribeiro, Rui Rua, Jacome Cunha, Jaoa Paulo Fernandes, Joao Saraiva (Universidade fo Minho, Portugal)

What Are the Greenest Programming Languages

When we choose programming languages, we consider their syntax, performances, or even learning curve. Yet, a few years ago a group of Portuguese researchers published a paper titled **Measuring the energy consumption of 27 most popular programming languages**, measuring execution time, energy consumption, and peak memory use.

Here is the abstract of the paper:

“This paper presents a study of the runtime, memory consumption, and energy consumption of twenty-seven well-known programming languages. We monitor the performance of these languages expressed in each of the languages. Our results show that some languages, such as slower/faster languages consuming less/more energy, and how memory usage influences energy efficiency is a concern.”

The results are the following:

C is the most efficient programming language, followed by Rust, C++, Ada, Java, Pascal, Chapel, Lisp, Ocaml, Fortran, Swift, Haskell, C#, Go, Dart, F#, JavaScript, Racket, TypeScript, Hack, PHP, Erlang, Lua, Jruby, Ruby, Python, and Perl.

Authors: Rui Pereira, Marco Couto, Francisco Lourenço (Universidade do Minho, Portugal)

<https://greenlab.di.uminho.pt/papers/2017/10/sleFinal.pdf>

	Energy
(c) C	1.00
(c) Rust	1.03
(c) C++	1.34
(c) Ada	1.70
(v) Java	1.98
(c) Pascal	2.14
(c) Chapel	2.18
(v) Lisp	2.27
(c) Ocaml	2.40
(c) Fortran	2.52
(c) Swift	2.79
(c) Haskell	3.10
(v) C#	3.14
(c) Go	3.23
(i) Dart	3.83
(v) F#	4.13
(i) JavaScript	4.45
(v) Racket	7.91
(i) TypeScript	21.50
(i) Hack	24.02
(i) PHP	29.30
(v) Erlang	42.23
(i) Lua	45.98
(i) Jruby	46.54
(i) Ruby	69.91
(i) Python	75.88
(i) Perl	79.58

their syntax, performances, or even learning curve. Yet, a few years ago a group of Portuguese researchers published a paper titled **Measuring the energy consumption of 27 most popular programming languages**, measuring execution time, energy consumption, and peak memory use.

“This paper presents a study of the runtime, memory consumption, and energy consumption of twenty-seven well-known programming languages. We monitor the performance of these languages expressed in each of the languages. Our results show that some languages, such as slower/faster languages consuming less/more energy, and how memory usage influences energy efficiency is a concern.”

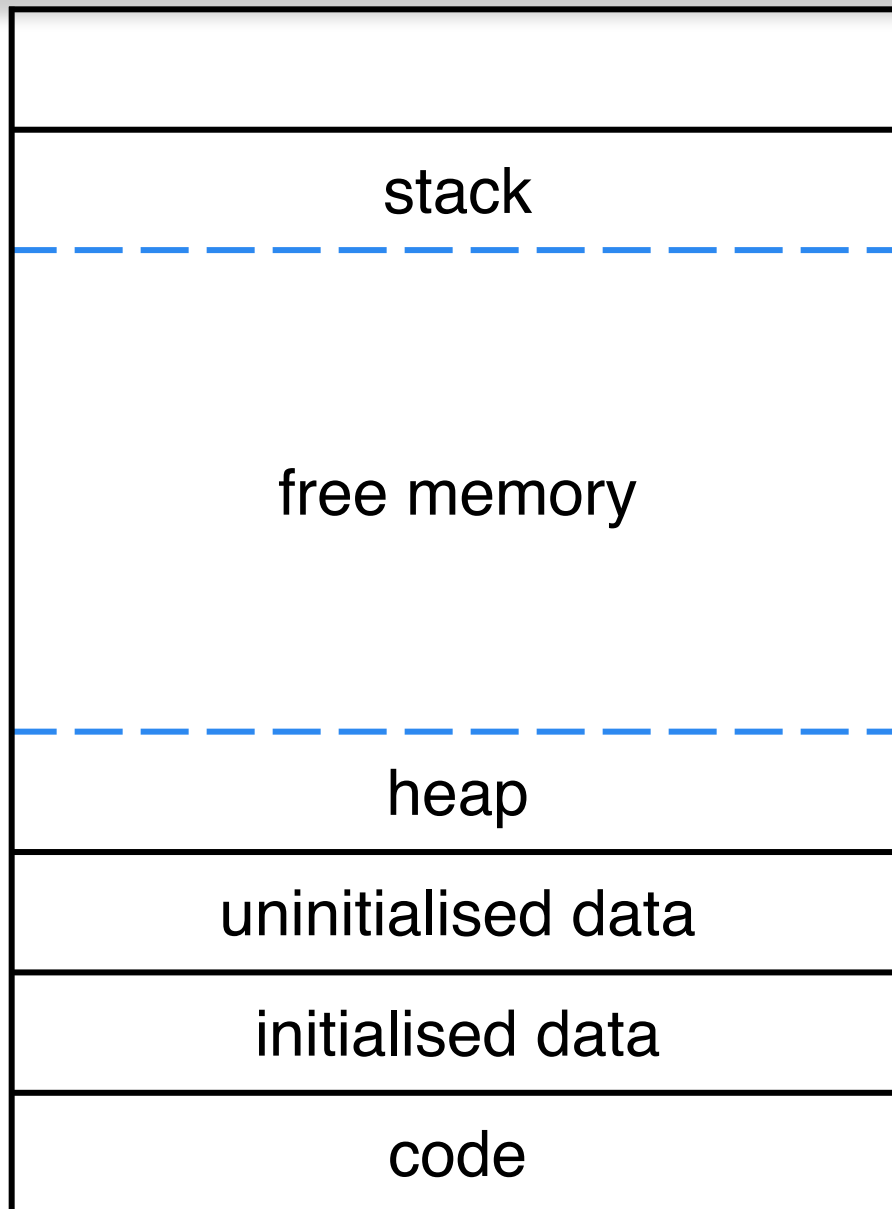
C is the most efficient programming language, followed by Rust, C++, Ada, Java, Pascal, Chapel, Lisp, Ocaml, Fortran, Swift, Haskell, C#, Go, Dart, F#, JavaScript, Racket, TypeScript, Hack, PHP, Erlang, Lua, Jruby, Ruby, Python, and Perl.

Authors: Rui Pereira, Marco Couto, Francisco Lourenço (Universidade do Minho, Portugal)

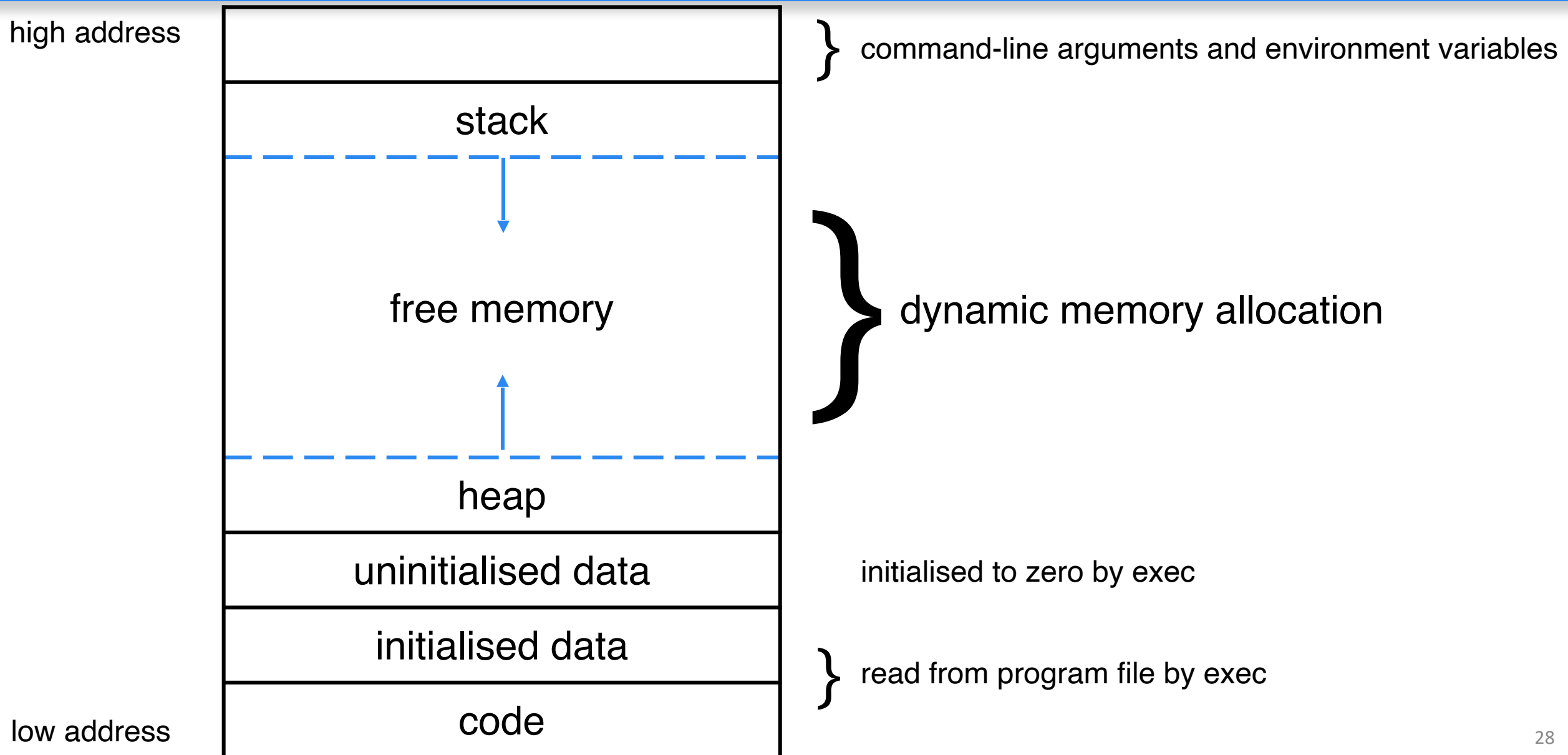
<https://greenlab.di.uminho.pt/papers/2017/10/sleFinal.pdf>

What's Memory Management, and Why do we study it?

Memory Organisation of a typical program



Memory Organisation of a typical program



COMP281

Principles of C
+
Memory Management



- You already have experience with programming, so hopefully you can pick up C syntax quickly and start having *fun* with coding.
- You will further practice programming & debugging skills.

General Module Information

Contact



Phil Jimmieson



phil@liverpool.ac.uk



0151 795 4236



<https://cgi.csc.liv.ac.uk/~phil>



Room 1.20 Ashton Building

Module Delivery

- Two lectures per week
 - Start in Week 1
 - Monday 15:00-16:00, Thursday 15:00-16:00
 - videos available as soon as possible afterwards, accessed from the Canvas site
- Two-hour lab session per week
 - Start in **Week 2**
 - You have been assigned to a group in one of four slots:
 - Tuesday 9:00 - 11:00 Group 3 & 11:00 - 13:00 Group 4
 - Friday 11:00 - 13:00 Group 1 & 15:00 - 17:00 Group 2

Learning Outcomes

At the end of the module, you will be able to

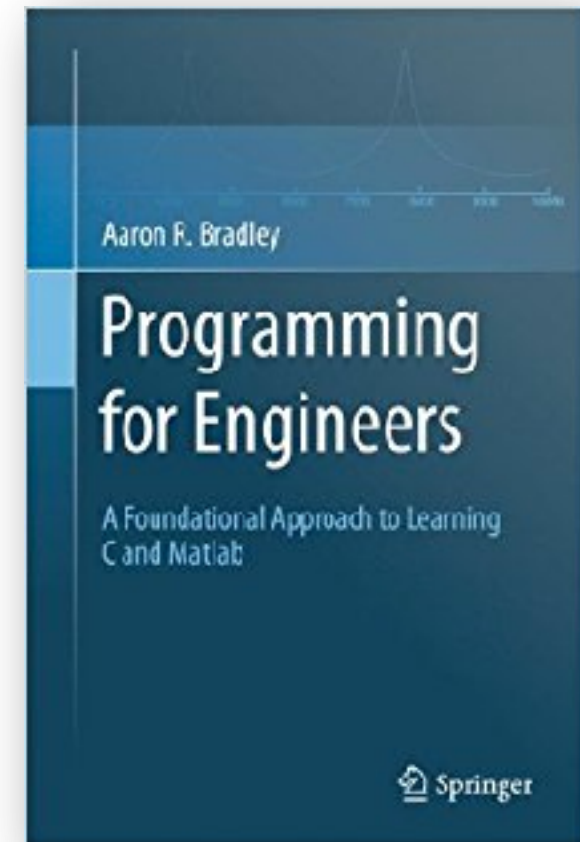
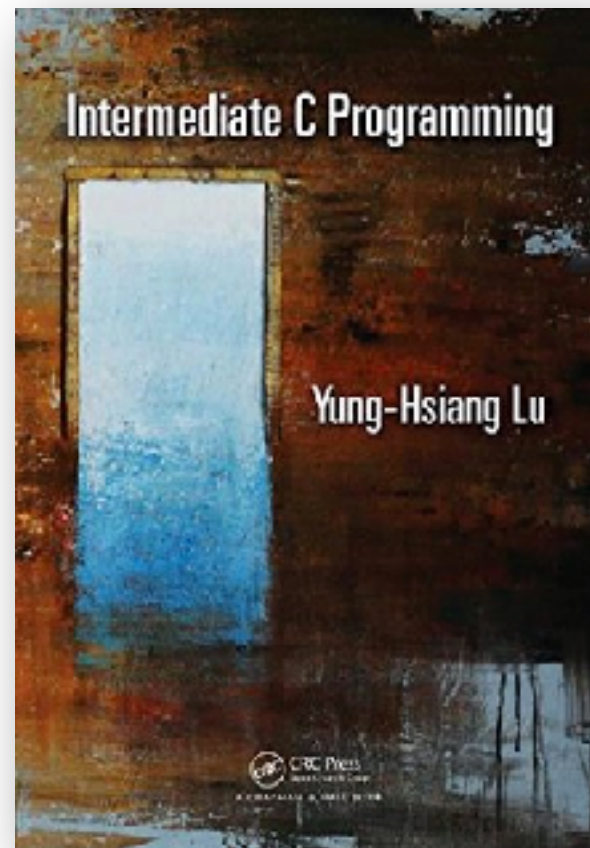
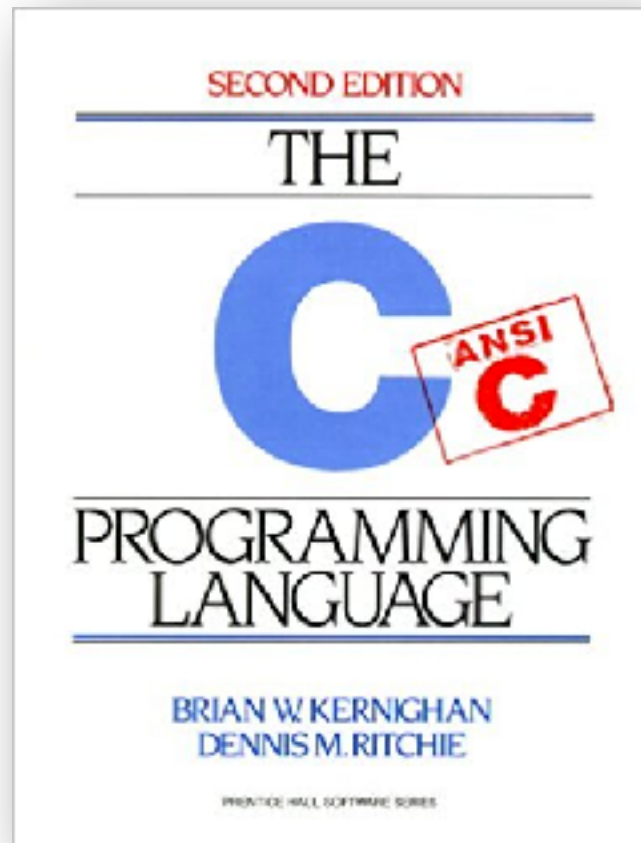
- Analyse and explain the use of memory resources within software applications, including memory usage on the **stack** during function calls and **heap**-based dynamic memory management;
- Use debugging tools to **inspect memory usage**, and to assist in the development of software;
- Develop applications using the C programming language, including use of **command-line** driven C development tools;
- Deal with underlying memory-based issues in using **dynamic data structures** through the implementation and management of at least one familiar data structure using the C programming language.

Assessment

- The module is 100% CA
 - No exam 🖐
- There will be 2 assessments worth 50% each
- Schedule (current plan)
 - Assignment 1: set at start of week 3, deadline Friday of week 4
 - Assignment 2: set at end of week 4, deadline Friday of week 6

Module Texts

- There is no required text for the module. Recommended texts:



Setup Programming Environment

- The programming environment Cygwin / Atom is setup on the lab machines in CS
- You should also set it up on your own (Windows) machine

- **Text editors**



Vim



Sublime Text



atom



Brackets



Visual Studio Code



BBEdit

- **The compiler**

- Windows / linux / Mac

<https://www.cygwin.com>

The COMP281 Webpage

- Canvas web site: COMP281-202324
 - All lecture notes will be posted as soon as possible prior to the lecture.
 - Video recordings of the lectures

What you should be doing

Teaching Schedule

“

- Lectures: two one-hr lectures per week for 5 weeks -> 10 hr in total
- Labs: one two-hr lab per week for 5 weeks -> 10 hr in total
- Self study: 55 hours
- In total: 75 hours

”

- Module Specification

What you should be doing

Of course

- Attend the lectures
- Attend labs

Self study

- Practice is the key to study coding
- Try to do a bit of coding every few days
- Submit assignments on time

How to get help

If you are stuck during the module

- Ask questions by email
- Attend the lab session and ask demonstrators
- Check any worked solutions on the website/lecture notes
- Read the text/slides
- Google/Stack Overflow/Co-Pilot/ ChatGPT (but beware of PLAGIARISM!!)
 - Your code will be checked for similarity

Summary

Today

- Principles of C and Memory Management?
what this module is about
- General module information.

After this lecture

- Setup your own C programming environment

Next lecture

- Compiling and Running C Programs
- C Language Basics