

- History: C and UNIX, C++, F77/F90
procedural vs. object oriented
- 1972/78 Kernighan & Ritchie, K&R → ANSI C
- source code → assembly code → binary

- general structure of a C program

- Control structures

for, while, do, break
 { } , " ; statement terminator

- reserved key words, case sensitive

Variable declarations
 char, int, float, double, void

static, unsigned
 ⇕
 const.

- Dynamic memory allocation
 malloc, realloc, sizeof

- operators

+ , - , / , *
 +4 , -- , / = , *=

- if tests

== , != , > , < , <=

- logic operators → bracketing → operator precedence!

- switch / case

(a=b)

- vectors, pointers &, *, dereferencing

- vector arithmetic

- pointers to pointers

- pointers to functions

- structures

- Functions

- parameters always passed by value (to pointer)

- declarations, header files, cproto

- command line arguments

- File I/O and formatting

- binary I/O

- preprocessors

- editors

- compiling, building makefiles

- debugging, lint

- profiling

- standard libraries

stdio

math.

stdlib

- Fortran and C conventions

- lack of features

- no array bound checking
- no garbage collection (memory leaks)
- no variable bound checking

→ address by careful programming / compiler options

- interfacing with external libraries