

Advance C Programming

Course Objective

This course is intended for programmers having experience of writing solid code. The course discusses coding techniques used by best "C" programmers. Main objective of the course is to discuss many aspects of C--from how memory is laid out to the details of pointers and arrays, how linkers work and how executable are structured. This course prepares strong foundation before one starts with cryptic kernel programming.

Course Content

- Introduction to C standards
- Declaration & Definitions in depth
- Struct ,enums,typedefs, #define
- Bitwise Arithmetic (with tricky programs)
- Precedence & Associativity
- Introduction to Lint
- Arrays & Pointer Arithmetic
- Is Array same as pointers?
- Pointers for Multidimensional arrays, Dynamic arrays
- Function Pointers etc.
- Unscrambling dynamic allocation
- Implement malloc & free
- Libraries Linking & Loading
- Dynamic linking
- Runtime System
- Segments & Executables
- O.S and a.out
- Standard calling conventions
- Activation Records
- X86 Memory Architecture & C
- Debugging Memory Leaks &
- Segmentation Faults
- Type Casting

Reference:

The C Programming Language, Second Edition
Brian W. Kernighan and Dennis M. Ritchie.

Prerequisite:

Basic C programming