

# Home

Welcome to System Programming coursebook!

This coursebook is being built by students and faculty from the University of Illinois. It is based on a crowd-source authoring wikibook experiment by Lawrence Angrave from CS @ Illinois, but is the github link

(<https://github.com/illinois-cs241/coursebook>)

now its own .tex based project. It's source is located at <https://github.com/illinois-cs241/coursebook> and can find a pdf version of the book as well.

This book is an introduction to programming in C, and system programming (processes, threads, synchronization, networking and more!). We assume you've already had some programming experience, in an earlier computer science course. If you have any typos to report or content to request, feel free to file an issue at the link above. Happy Reading!

– Bhuvy

📄 One Big File

([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/main.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/main.pdf))



## 1. Introduction ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/introduction.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/introduction.pdf))

Authors

1. ([./Introduction#authors](#))



## 2. Background ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/background.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/background.pdf))

Systems Architecture

([./Background#systems-architecture](#))

1. ([./Background#systems-architecture](#))

Debugging and Environments

([./Background#debugging-and-environments](#))

2. ([./Background#debugging-and-environments](#))

Valgrind

3. ([./Background#valgrind](#))

GDB

4. ([./Background#gdb](#))

Homework 0

([./Background#homework-0](#))

5. ([./Background#homework-0](#))

UIUC Specific Guidelines

([./Background#uiuc-specific-guidelines](#))

6. ([./Background#uiuc-specific-guidelines](#))



## 3. C Programming Language ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/introc.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/introc.pdf))

History of C

([./Introc#history-of-c](#))

1. ([./Introc#history-of-c](#))

Features

2. ([./Introc#features](#))

Crash course intro to C

([./Introc#crash-course-intro-to-c](#))

3. ([./Introc#crash-course-intro-to-c](#))

Preprocessor

4. ([./Introc#preprocessor](#))

Language Facilities

([./Introc#language-facilities](#))

5. ([./Introc#language-facilities](#))

The C and Linux

([./Introc#the-c-and-linux](#))

6. ([./Introc#the-c-and-linux](#))

Common C Functions

([./Introc#common-c-functions](#))

7. ([./Introc#common-c-functions](#))

- C Memory Model
  - (./Intro#c-memory-model)
- 8. Pointers
  - (./Intro#pointers)
- Common Bugs
  - (./Intro#common-bugs)
- 10. Logic and Program flow mistakes
  - (./Intro#logic-and-program-flow-mistakes)
- 11. Topics
  - (./Intro#topics)
- 12. Questions/Exercises
  - (./Intro#questions/exercises)
- 13. Rapid Fire: Pointer Arithmetic
  - (./Intro#rapid-fire-pointer-arithmetic)
- 14.



#### 4. Processes ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/processes.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/processes.pdf))

- File Descriptors
  - (./Processes#file-descriptors)
- 1. Processes
  - (./Processes#processes)
- 2. Process Contents
  - (./Processes#process-contents)
- 3. Intro to Fork
  - (./Processes#intro-to-fork)
- 4. Waiting and Execing
  - (./Processes#waiting-and-execing)
- 5. exec
  - (./Processes#exec)
- 6. The fork-exec-wait Pattern
  - (./Processes#the-fork-exec-wait-pattern)
- 7. Further Reading
  - (./Processes#further-reading)
- 8. Questions/Exercises
  - (./Processes#questions/exercises)
- 9.



#### 5. Memory Allocators ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/malloc.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/malloc.pdf))

- Introduction
  - (./Malloc#introduction)
- 1. C Memory Allocation API
  - (./Malloc#c-memory-allocation-api)
- 2. Intro to Allocating
  - (./Malloc#intro-to-allocating)
- 3. Memory Allocator Tutorial
  - (./Malloc#memory-allocator-tutorial)
- 4.

Case study: Buddy Allocator, an example of a segregated list

(./Malloc#case-study-buddy-allocator-an-example-of-a-segregated-list)

5.

Further Reading

(./Malloc#further-reading)

6.

Topics

7. (./Malloc#topics)

Questions/Exercises

8. (./Malloc#questions/exercises)



## 6. Threads ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/threads.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/threads.pdf))

Processes vs threads

(./Threads#processes-vs-threads)

1.

Thread Internals

(./Threads#thread-internals)

2.

Simple Usage

(./Threads#simple-usage)

3.

Pthread Functions

(./Threads#pthread-functions)

4.

Race Conditions

(./Threads#race-conditions)

5.

Topics

6. (./Threads#topics)

Questions

7. (./Threads#questions)



## 7. Synchronization ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/synchronization.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/synchronization.pdf))

Mutex

1. (./Synchronization#mutex)

Condition Variables

(./Synchronization#condition-variables)

2.

Thread Safe Data Structures

(./Synchronization#thread-safe-data-structures)

3.

Software Solutions to the Critical Section

(./Synchronization#software-solutions-to-the-critical-section)

4.

Working Solutions

(./Synchronization#working-solutions)

5.

Implementing Counting Semaphore

(./Synchronization#implementing-counting-semaphore)

6.

Ring Buffer

(./Synchronization#ring-buffer)

7.

Extra: Process Synchronization

(./Synchronization#extra-process-synchronization)

8.

Extra: Higher Order Models of Synchronization

(./Synchronization#extra-higher-order-models-of-synchronization)

9.

- External Resources
  - (./Synchronization#external-resources)
- 10. Topics
  - (./Synchronization#topics)
- Questions
  - (./Synchronization#questions)



## 8. Deadlock ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/deadlock.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/deadlock.pdf))

- Resource Allocation Graphs
  - (./Deadlock#resource-allocation-graphs)
- 1. Coffman conditions
  - (./Deadlock#coffman-conditions)
- 2. Approaches to solving deadlock
  - (./Deadlock#approaches-to-solving-deadlock)
- 3. Dining Philosophers
  - (./Deadlock#dining-philosophers)
- 4. Viable Solutions
  - (./Deadlock#viable-solutions)
- 5. Topics
  - (./Deadlock#topics)
- 6. Questions
  - (./Deadlock#questions)



## 9. Virtual Memory and Interprocess Communication ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/ipc.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/ipc.pdf))

- Translating Addresses
  - (./Ipc#translating-addresses)
- 1. mmap
  - (./Ipc#mmap)
- 2. Pipes
  - (./Ipc#pipes)
- 3. Named Pipes
  - (./Ipc#named-pipes)
- 4. Files
  - (./Ipc#files)
- 5. So Many IPC Options
  - (./Ipc#so-many-ipc-options)
- 6. Topics
  - (./Ipc#topics)
- 7. Questions
  - (./Ipc#questions)



## 10. Scheduling ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/scheduling.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/scheduling.pdf))

- High Level Scheduler Overview
  - (./Scheduling#high-level-scheduler-overview)
- 1. Measurements
  - (./Scheduling#measurements)
- 2. Measures of Efficiency
  - (./Scheduling#measures-of-efficiency)
- 3. Scheduling Algorithms
  - (./Scheduling#scheduling-algorithms)
- 4. Scheduling Conceptually
  - (./Scheduling#scheduling-conceptually)
- 5. Topics
  - (./Scheduling#topics)

Questions

7. ([./Scheduling#questions](#))

## 11. Networking ([https://github.com/illinois-cs241/courses241/coursebook/blob/pdf\\_deploy/networking.pdf](https://github.com/illinois-cs241/courses241/coursebook/blob/pdf_deploy/networking.pdf))

The OSI Model

([./Networking#the-osi-](#)

model)

1. Layer 3: The Internet Protocol  
([./Networking#layer-3:-the-internet-protocol](#))
2. Layer 4: TCP and Client  
([./Networking#layer-4:-tcp-and-client](#))
3. Layer 4: TCP Server  
([./Networking#layer-4:-tcp-server](#))
4. Layer 4: UDP  
([./Networking#layer-4:-udp](#))
5. Layer 7: HTTP  
([./Networking#layer-7:-http](#))
6. Non Blocking IO  
([./Networking#non-blocking-io](#))
7. Remote Procedure Calls  
([./Networking#remote-procedure-calls](#))
8. Topics  
([./Networking#topics](#))
9. Questions  
([./Networking#questions](#))



## 12. Filesystems ([https://github.com/illinois-cs241/courses241/coursebook/blob/pdf\\_deploy/filesystems.pdf](https://github.com/illinois-cs241/courses241/coursebook/blob/pdf_deploy/filesystems.pdf))

What is a filesystem?

([./Filesystems#what-is-](#)

a-

filesystem?)

1. Storing data on disk  
([./Filesystems#storing-data-on-disk](#))
2. Permissions and bits  
([./Filesystems#permissions-and-bits](#))
3. Virtual filesystems and other filesystems  
([./Filesystems#virtual-filesystems-and-other-filesystems](#))
4. Memory Mapped IO  
([./Filesystems#memory-mapped-io](#))
5. Reliable Single Disk Filesystems  
([./Filesystems#reliable-single-disk-filesystems](#))
6. Simple Filesystem Model  
([./Filesystems#simple-filesystem-model](#))
- 7.

- Topics
- 8. ([./Filesystems#topics](#))
- Questions
- 9. ([./Filesystems#questions](#))



## 13. Signals ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/signals.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/signals.pdf))

- The Deep Dive of Signals
- ([./Signals#the-deep-dive-of-signals](#))
- 1. Sending Signals
- ([./Signals#sending-signals](#))
- 2. Handling Signals
- ([./Signals#handling-signals](#))
- 3. Signal Disposition
- ([./Signals#signal-disposition](#))
- 4. Disposition in Child Processes (No Threads)
- ([./Signals#disposition-in-child-processes-\(no-threads\)](#))
- 5. Signals in a multithreaded program
- ([./Signals#signals-in-a-multithreaded-program](#))
- 6. Topics
- 7. ([./Signals#topics](#))
- Questions
- 8. ([./Signals#questions](#))



## 14. Review ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/review.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/review.pdf))

- C
- 1. ([./Review#c](#))
- Threading
- 2. ([./Review#threading](#))
- Deadlock
- 3. ([./Review#deadlock](#))
- IPC
- 4. ([./Review#ipc](#))
- Filesystems
- 5. ([./Review#filesystems](#))
- Networking
- 6. ([./Review#networking](#))
- Signals
- 7. ([./Review#signals](#))



## 15. Honors topics ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/honors.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/honors.pdf))

- The Linux Kernel
- ([./Honors#the-linux-kernel](#))
- 1. Containerization
- ([./Honors#containerization](#))
- 2.



## 16. Appendix ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/appendix.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/appendix.pdf))

- Shell
- 1. ([./Appendix#shell](#))
- Stack Smashing
- ([./Appendix#stack-smashing](#))
- 2.

System Programming Jokes  
 (./Appendix#system-  
 programming-  
 jokes)

3.



## Post Mortems ([https://github.com/illinois-cs241/coursebook/blob/pdf\\_deploy/post\\_mortems.pdf](https://github.com/illinois-cs241/coursebook/blob/pdf_deploy/post_mortems.pdf))

17.

- Shell Shock  
 (./Post\_Mortems#shell-shock)
- 1. Heartbleed  
 (./Post\_Mortems#heartbleed)
- 2. Dirty Cow  
 (./Post\_Mortems#dirty-cow)
- 3. Meltdown  
 (./Post\_Mortems#meltdown)
- 4. Spectre  
 (./Post\_Mortems#spectre)
- 5. Mars Pathfinder  
 (./Post\_Mortems#mars-pathfinder)
- 6. Mars Again  
 (./Post\_Mortems#mars-again)
- 7. Year 2038  
 (./Post\_Mortems#year-2038)
- 8. Northeast Blackout of 2003  
 (./Post\_Mortems#northeast-blackout-of-2003)
- 9. Apple IOS Unicode Handling  
 (./Post\_Mortems#apple-ios-unicode-handling)
- 10. Apple SSL Verification  
 (./Post\_Mortems#apple-ssl-verification)
- 11. Sony Rootkit Installation  
 (./Post\_Mortems#sony-rootkit-installation)
- 12. Civilization and Ghandi  
 (./Post\_Mortems#civilization-and-ghandi)
- 13. The Woes of Shell Scripting  
 (./Post\_Mortems#the-woes-of-shell-scripting)
- 14. Appnexus Double Free  
 (./Post\_Mortems#appnexus-double-free)
- 15. ATT Cascading Failures - 1990  
 (./Post\_Mortems#att-cascading-failures-1990)
- 16.





