

HW1: malloctopus

Heap memory allocator library

- Ported CSAPP 32-bit implementation for 64-bit systems
- Experimented with different data structures & strategies
 - Implicit, explicit, segregated free lists

Further learning:

- google/tcmalloc: thread-caching malloc
- CSAPP 9.10: Garbage collection

HW2: greptile 🦎

Multi-threaded pattern matching utility

- Traversed UNIX file hierarchy with using `readdir()` API
- Worker thread pool, ring buffer paradigm, synchronization

Further learning:

- greptile part 3: performance evaluation
- ripgrep [design](#)

HW3: cowchat

Chat server built with advanced I/O techniques

- Multiplexing client connections using select()
- Interprocess communication: fork, pipes, signals
- Domain sockets for connection sharing
- Shared memory & named semaphores for synchronization

Further learning:

- APUE 14.5 Asynchronous I/O
- APUE 14.6 Scatter reads, gather writes
- Linux io_uring: https://unixism.net/loti/what_is_io_uring.html

HW4: ladebug

Assembly-level command line debugger

- Built on top of Linux ptrace() API
- x86-64 assembly fundamentals
- Software breakpoints using x86 int3 instruction
- Function address resolution using ELF symbol table

Further learning:

- Hardware breakpoints (watchpoints)
- Source-level debugging: DWARF debug data format

HW5: seald ☒

Linker for no-pie executables

- Parsed input relocatable ELF objects
- Performed symbol resolution and relocations across arbitrary number of object files

Further learning:

- Handle .rodata, .data, .bss (global/static variables, string literals, etc.)
- Linking with static and shared libraries

HW6: zookeeper [🐙 🦎 🐄 🐞 📧]

Linux container built on top of system facilities

- cgroups: Resource control
- Namespaces: Resource isolation
- Capabilities/seccomp: Kernel access control

Further learning:

- History of virtualization & virtual machines
- Experiment with Docker and Podman

What comes next

COMS 4118: Operating Systems I

- Linux kernel internals: syscalls, synchronization, memory, scheduling, filesystems

COMS 4115: Programming Languages & Translators

- Compiling source code to intermediate code; optimizations
- Garbage collection, language runtime systems

COMS 4113: Fundamentals of Distributed Systems

- IPC, synchronization, parallelization

What comes next

COMS 4186: Malware Analysis & Reverse Engineering

- Reverse engineering malware, more x86 assembly debugging
- Windows internals & executable format (PE)

CSEE 4824: Computer Architecture

- Hardware optimizations, cache considerations
- Storage hierarchy: cache, memory, disk

Thank you!

