CSE 4502: Operating Systems Lab

Introduction to the Course

Department of Computer Science and Engineering (CSE)
Islamic University of Technology (IUT)

06 September 2024

Introduction to CSE 4502

Course Teachers

S. M. Sabit Bananee
Lecturer,
Department of CSE,
Islamic University of Technology (IUT).

Dr. Muhammad Mahbub Alam
Professor,
Department of CSE,
Islamic University of Technology (IUT).

Syed Rifat Raiyan
Lecturer,
Department of CSE,
Islamic University of Technology (IUT).

Md. Monowarul Amin
Student, Batch 2020
Program: SWE
Department of CSE,
Islamic University of Technology (IUT).

Course Objectives

Building a simple Operating Systems (OS) kernel implementing the basic OS concepts like:

- Process and Thread Management,
- Physical Memory management,
- Virtual Memory management,
- Multiprogramming,
- File systems, etc.

Course Organization

- There will be Five (5) experiments and one (1) project.
- To complete a single experiment, students will have two (2) lab weeks.
    - In the first lab week of an experiment, a brief description of the experiment will be delivered, and students will present their solution to the previous experiment. They must bring a prepared ppt to present.
    - The second lab week of the experiment will consist of the query, discussion, and question and answer session.
- A student must submit their solution to the created Google Classroom assignment on the day before coming for the second week of lab.
- Students must have object-oriented concepts as a prerequisite.

Resources and Tools

- The CS422/522: Design and Implementation of Operating Systems course at Yale University will be used as the primary resource.
- The ‘Qemu’ emulator with a multi-library-enabled GCC compiler will be used to emulate the kernel.
- As a debugger, ‘gdb’ (GNU debugger) will be used.
- Any code editor, such as Visual Studio Code, may be used for writing the code.
- Google Classroom will be the primary platform for accessing information and resources.

Assessment Methods & Grading Policy

A student will be evaluated based on

- his/ her presence during the lab sessions,
- his/her solutions to the experiment,
- question answering skills during the viva session.
- the submitted projects.

Attendance	10 %
Continuous Lab Evaluation	40 %
Projects	50 %

Experiments

Experiments	Name	Lab Weeks
Experiment 1	Bootloader & Physical Memory Management	Week 1 & Week 2
Experiment 2	Container & Virtual Memory Management	Week 3 & Week 4
Experiment 3	Process Management & Trap Handling	Week 5 & Week 6
Experiment 4	Multicore and Preemption	Week 7 & Week 8
Experiment 5	File Systems	Week 9 & Week 10
Experiment 6	Advanced OS Projects	Rest of the Semester.

Final Projects

- As a part of the course, students must submit a team project.
- A team will consist of two members.
- Each team must select a final project from the provided project list. The description of projects can be found here.
- One team member must submit their team information with three selected projects in the first week after the mid-examination.
- At best, a single project can be selected by four (4) teams.
- If the number of teams for a specific project is exceeded, then the teams will be selected based on the team members' CGPA.
- The team having a member with a higher CGPA will get the lowest priority in the selection process.
- Each team must submit their solution and project report before the end of the semester.

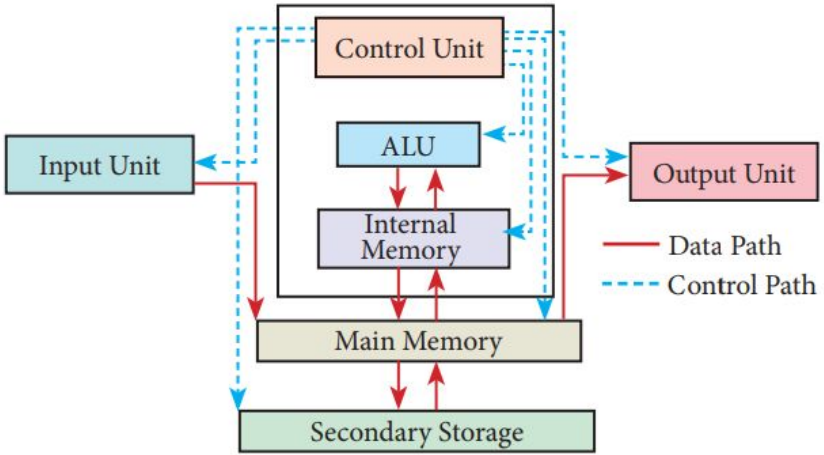
Project List

Projects	Title
Project 1	Video Mode
Project 2	UNIX Shell
Project 3	Debugger
Project 4	Scheduler
Project 5	Audio Support
Project 6	In-kernel Cryptographic Framework
Project 7	Advanced Synchronization - File Sharing
Project 8	Advanced Synchronization - Wait for Multiple Objects
Project 9	Signal
Project 10	Advanced Memory Allocation
Project 11	Advanced Synchronization: User-level Mutex
Project 12	Pub/Sub IPC
Project 13	Networking
Project 14	Resource Limitation and Bandwidth Control

Lab 01: Bootloader & Physical Memory Management

Software setup
Part 1: PC Bootstrap
Part 2: Bootloader
Part 3: Physical Memory Management

Introduction to Computer Systems



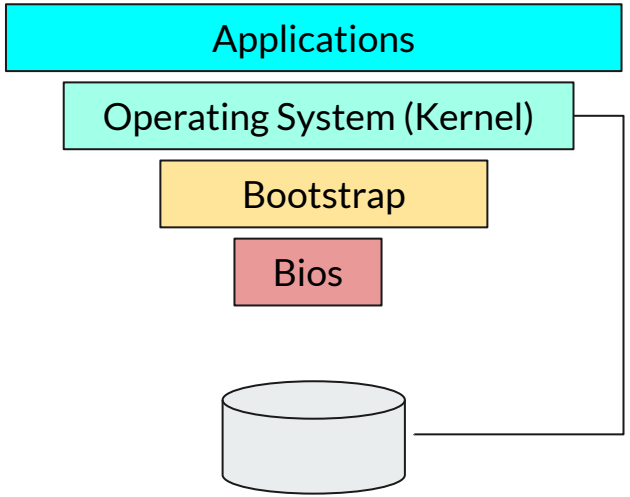


Figure 1: Components of a Computer.

System Boot Sequence

main job of bootstrap : fetches the OS and brings it to the main memory from the secondary memory

- Your turn on power
- CPU jumps to the beginning of BIOS ROM
- CPU executes POST (Power-On Self-Test) and initializes hardware
- CPU executes BIOS routine to load MBR (Master Boot Reader)
- CPU jumps to the routine in MBR
- MBR contains a routine (“ Boot Strap Loader”) to check the partition.
    - Find out which logical drive is the system boot drive
    - Load the boot block (“BootStrap”) of the boot drive and CPU jumps to it.
    - The Boot block contains a routine to start OS (start loading OS system files & drives)

Physical Memory Management

Thank you.