Operating System

Dr. Hamada I. AbdulWakel

2023-2024

Lecture 1

Course Details

Course Code	CS322
Course Name	Operating System
Coordinate Unit	CS depart. FCI
Term	Semester II (Winter)
Level	CS and IS Undergraduate - Level 3

Communications and Course Materials

In this course, our materials will based on the following book and the weekly lectures.

Main Book

Operating System - from **Minia Portal**

Grading Policy

Attendance & Take-home sheets	10
Practical assignments	10
Lab Exam	10
Midterm exam	10
Final exam	60

Policies

- Show mutual respect and listen.
- Everyone is punctual tried their best to attend the lecture on time. Exception 15 minutes.
- Please turn off mobiles.
- Expect interaction. You will be asked at least one question in lecturers between now and the end of term.
- Avoid distractions. Sneak in quietly and take the nearest available seat.
- Discussions will be in my office during office hours.
- People sitting on the back row!!!!.

Course Objectives

- know basic components of an operating system.
- Comprehend how an operating system virtualises CPU and memory.
- Discuss various scheduling and swapping policies.
- Explain the details of some units such as memory management, I/O, and file systems.
- Illustrate the above concepts such as UNIX/LINUX and Windows by using two case studies.
- Get to know how an operating system protects the computer system.

Book Review

The Book Chapters

Ch1 Introduction.

Ch2 Operating-System Structures.

Ch3 Processes.

Ch4 Threads.

Ch5 Process Synchronization.

Ch6 CPU Scheduling.

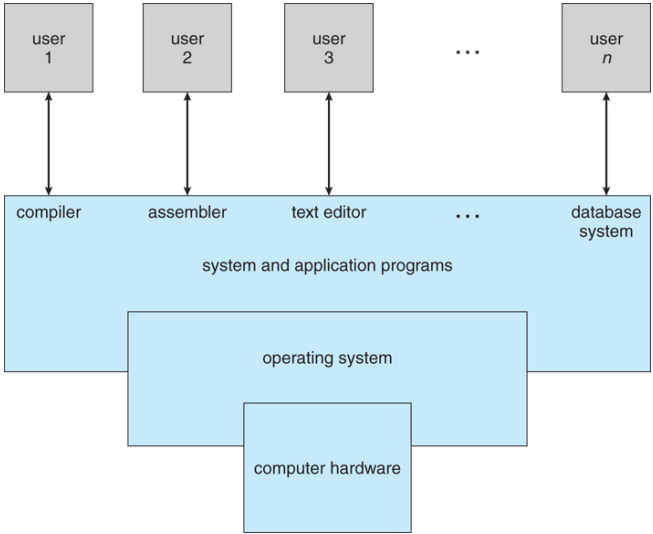
Ch7 Deadlocks.

Ch8 Main Memory.

Ch9 Virtual Memory.

Ch1: Introduction

Computer System Structure



Computer System Structure

- Computer system can be divided into four components:
  1. **Hardware:** provides basic computing resources such as CPU, memory, and I/O devices.
  2. **Operating system:** controls and coordinates use of hardware among various applications and users
  3. **Application programs:** define the ways in which the system resources are used to solve the computing problems of the users.
  4. **Users:** people, machines, other computers.

Operating System

- A program that manages the computer hardware. It also provides a basis for application programs and acts as an intermediary between the computer user and the computer hardware.
- OS goals:
  1. Execute user programs and make solving user problems easier.
  2. Make the computer system convenient to use.
  3. Use the computer hardware in an efficient manner.
  4. Manages secondary memory and I/O devices.

OS Different Views

- **Users view:**
  - The goal is to maximize the work (or play) that the user is performing.
  - In this case/ the operating system is designed mostly for ease of use and good performance.
  - Don’t care about resource utilization.
- **Shared computer view:**
  - In this case OS is designed to maximize resource utilization to assure that all available CPU time, memory, and I/O are used efficiently, and that no individual user takes more than his fair share.

OS Different Views ...

- **System view:**
  - A computer system has many resources that may be required to solve a problem: CPU time, memory space, file-storage space, I/O devices, and so on.
  - Operating system as a resource allocator.
  - OS must decide how to allocate them to specific programs and users so that it can operate the computer system efficiently and fairly.

Computer Startup

- The bootstrap program is loaded at power-up or reboot:
  - Typically stored in ROM, generally known as firmware.
  - Initializes all aspects of system.
  - Loads operating system kernel and starts execution.
  - "The one program running at all times on the computer" is the **kernel**.
  - Along with the kernel, there are two other types of programs:
    1. System programs (e.g., MS. Paint), which are associated with the OS system but are not necessarily part of the kernel.
    2. Application programs (e.g., PDF reader), which include all programs not associated with the operation of the system.
  - Mobile OSs often include not only a core kernel but also **middleware**.
    - A set of software frameworks that provide additional services to application developers.

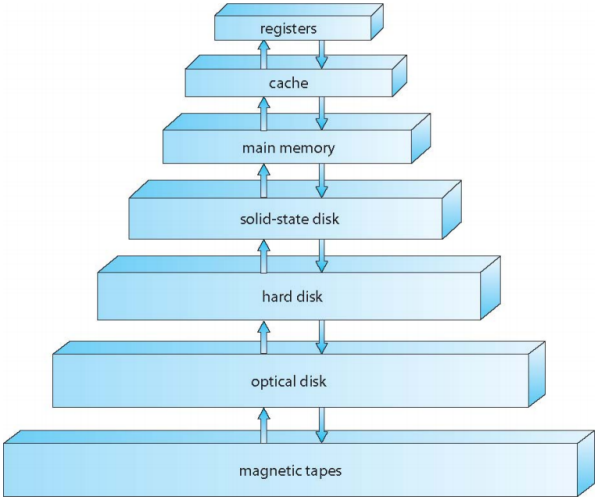
Computer-System Operation

- I/O devices and the CPU can execute concurrently.
- Device controller informs CPU that it has finished its operation by causing an interrupt.
- Interrupt transfers control to the interrupt service routine generally, through the interrupt vector, which contains the addresses of all the service routines.
- A trap or exception is a software-generated interrupt caused either by an error or a user request.
- Software may trigger an interrupt executing a special operation called a system call (also called a monitor call).

Storage Structure

- Storage systems organized in hierarchy.
  - Speed.
  - Cost.
  - Volatility.
- Main memory: storage media that the CPU can access directly.
  - Random access.
  - Typically volatile.
- Secondary storage: extension of main memory that provides large nonvolatile storage capacity.
  - Hard disks: rigid metal or glass platters covered with magnetic recording material.
    - Disk surface is logically divided into tracks, which are subdivided into sectors.
    - The disk controller determines the logical interaction between the device and the computer.

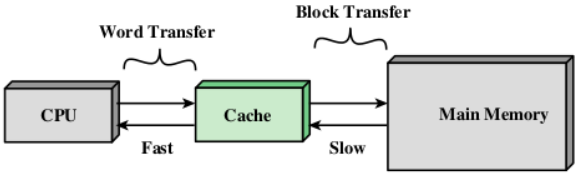
Storage Hierarchy



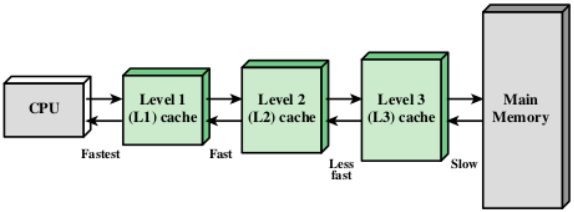
Caching

- **Caching:** copying information into faster storage system; main memory can be viewed as a cache for secondary storage.
- Important principle, performed at many levels in a computer (in hardware, operating system, software).
- Information in use copied from slower to faster storage temporarily.
- Faster storage (cache) checked first to determine if information is there.
  - If it is, information used directly from the cache (fast).
  - If not, data copied to cache and used there.
- Cache smaller than storage being cached.
  - Cache management important design problem.
  - Cache size and replacement policy.

Cache and Main Memory

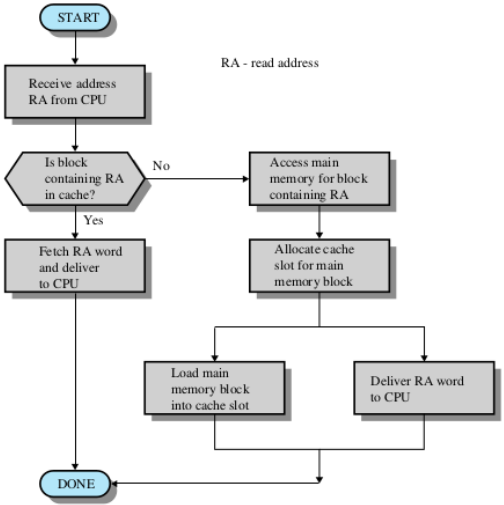


(a) Single cache



(b) Three-level cache organization

Cache Read Operation



Cache and Block Size

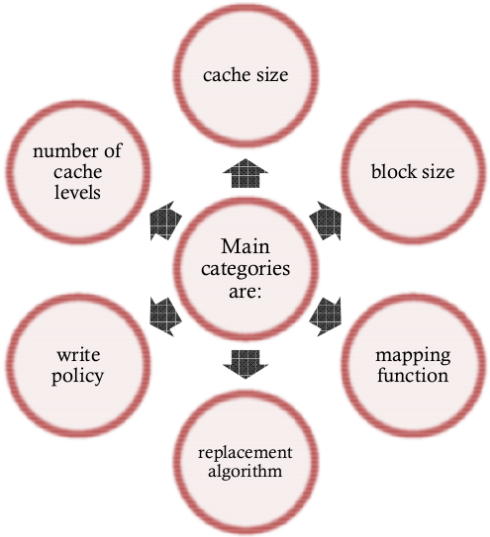
Cache Size

small caches have significant impact on performance

Block Size

the unit of data exchanged between cache and main memory

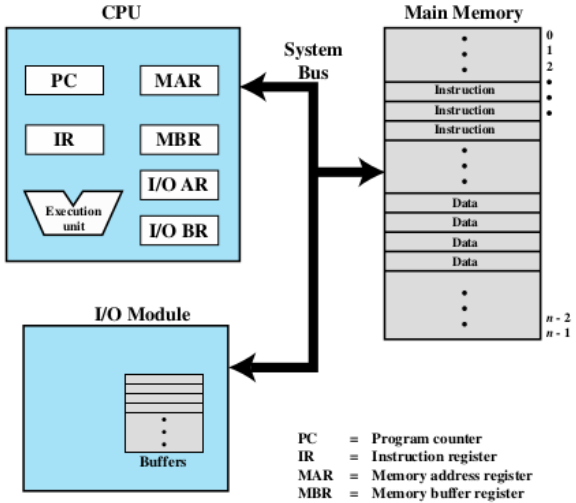
Cache Design



I/O Modules and System Bus

- I/O modules: Moves data between the computer and external environment such as:
  - Storage.
  - Communication equipment.
  - Terminals.
- System bus:
  - Provides for communication among processors, main memory, and I/O modules.
  - A pathway made up of electronic cables that carry the data back and forth from the computer's central processing unit (CPU) to other areas of the computer.

Computer Components: Top-Level View

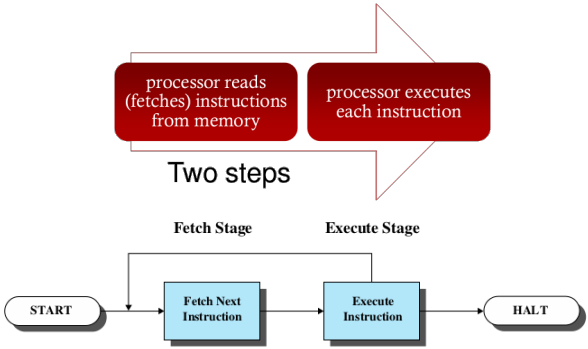


Process Management

- A process is a program in execution. It is a unit of work within the system. Program is a **passive entity**, process is an **active entity**.
- Process needs resources to accomplish its task.
  - CPU, memory, I/O, files.
  - Initialization data.
- Process termination requires reclaim of any reusable resources.
- Single-threaded process has one **program counter (PC)** specifying location of next instruction to execute.
  - Process executes instructions sequentially, one at a time, until completion.
- Typically system has many processes, some user, some operating system running concurrently on one or more CPUs.

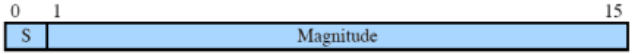
Instruction Execution/Cycle

- A program consists of a set of instructions stored in memory. PC is incremented after each fetch.



A Hypothetical Machine

(a) Instruction format



(b) Integer format

Program counter (PC) = Address of instruction
Instruction register (IR) = Instruction being executed
Accumulator (AC) = Temporary storage

(c) Internal CPU registers

0001 = Load AC from memory
0010 = Store AC to memory
0101 = Add to AC from memory

(d) Partial list of opcodes

Example of Program Execution