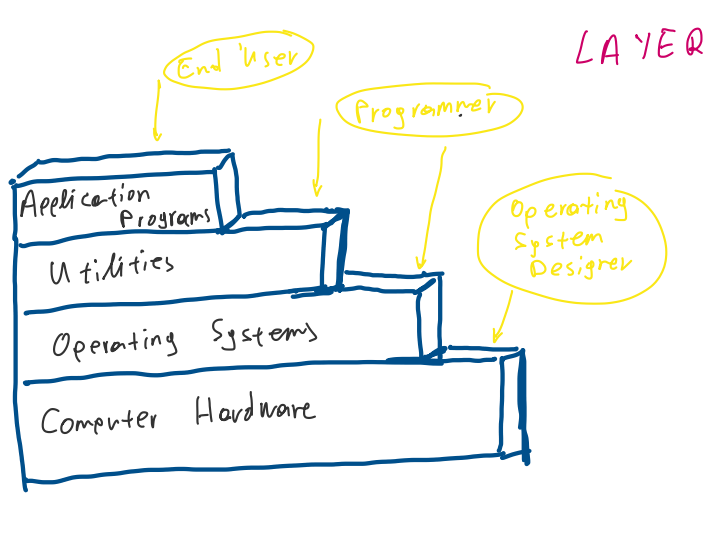
Friday, 18 October 2024 02:17

OPERATING SYSTEMS

- Program that controls the execution of application programs
- An interface between applications n' hardware

LAYERS AND VIEWS


End User
Programmer
Operating System Designer
Application Programs
Utilities
Operating Systems
Computer Hardware

SERVICES PROVIDED BY THE OS

- Program execution
- Access I/O devices
- Controlled access to files
- System Access

• ERROR DETECTION N' RESPONSE

① Internal and external hardware errors
② Software errors
③ Operating system cannot grant request of application

ACCOUNTING

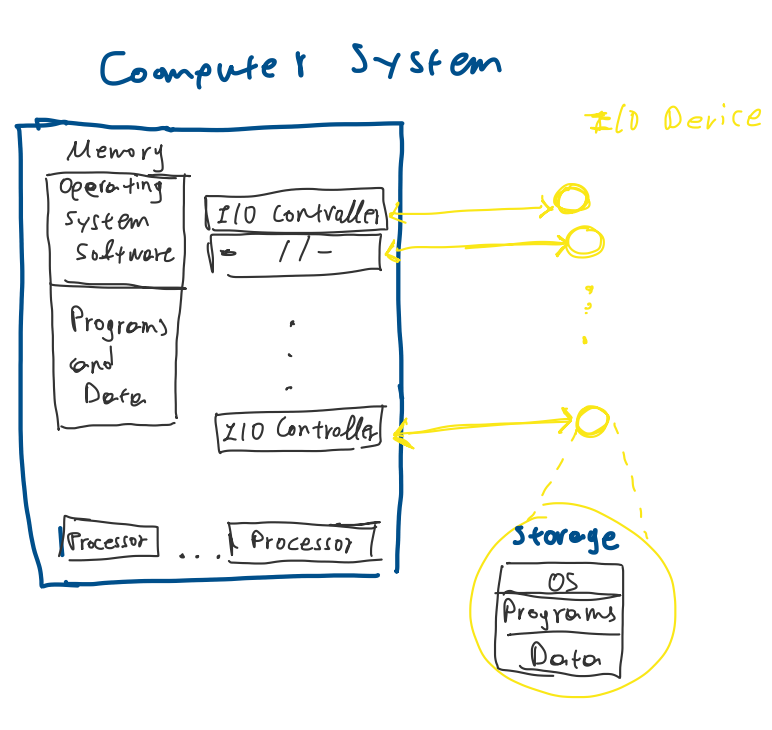
① Collect Usage Statistics
② Monitoring performance
③ Used to anticipate future enchancements
④ Used for billing purposes

- Responsible for managing resources

- Functions same way as ordinary computer software
  ↳ It is a program that is executed
- Operating system relinquishes control of the processor

OS as Resource Manager

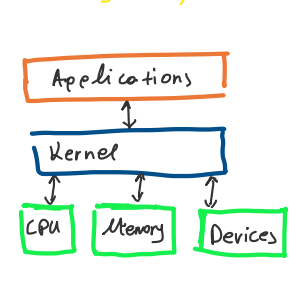
Computer System


I/O Device
Memory
Operating System Software
Programs and Data
I/O Controller
I/O Controller
I/O Controller
Processor
Processor
Storage
OS
Programs
Data

KERNEL

→ (is a computer program at the core of a computer's operating system n generally has complete control over everything in the system)

- Portion of operating system that is in main memory
- Contains most frequently used functions


Applications
Kernel
CPU
Memory
Devices

EVOLUTION OF OPERATING SYSTEMS

① Hardware upgrades plus new types of hardware:
- Device drivers

② New services
- These days: rare

③ Fixes
- Mostly: security fixes! (patching =))

SOME HISTORY

→ Serial processing
- No operating system
- Machines run from a console with display lights, toggle switches, input device, and printer
- Schedule time
- Setup included loading the compiler, source program, saving compiled program, and loading and linking

→ Monitor
- software that controls the sequence of events
- Batch jobs together - job control language
- Program returns control to monitor when finished

→ Hardware:
- Memory protection: some memory areas are accessible only to the monitor
- Privileged mode instructions: only accessible to the monitor
- Interrupts (early machines did not have this)

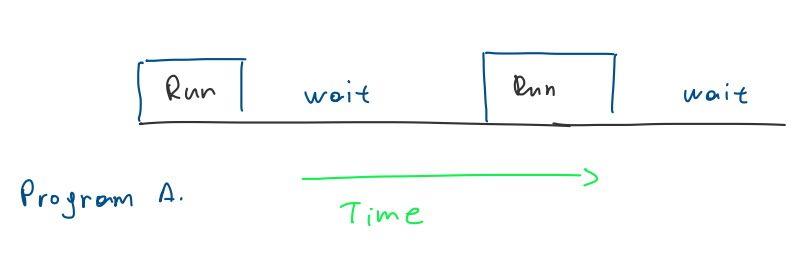
EXAMPLE

Read 1 record from file	15 µs
Execute 100 instructions	1 µs
Write 1 record to file	15 µs
	31 µs

Percent of CPU Utilization = $\frac{1}{31} = 0{,}032 = 3{,}2\%$

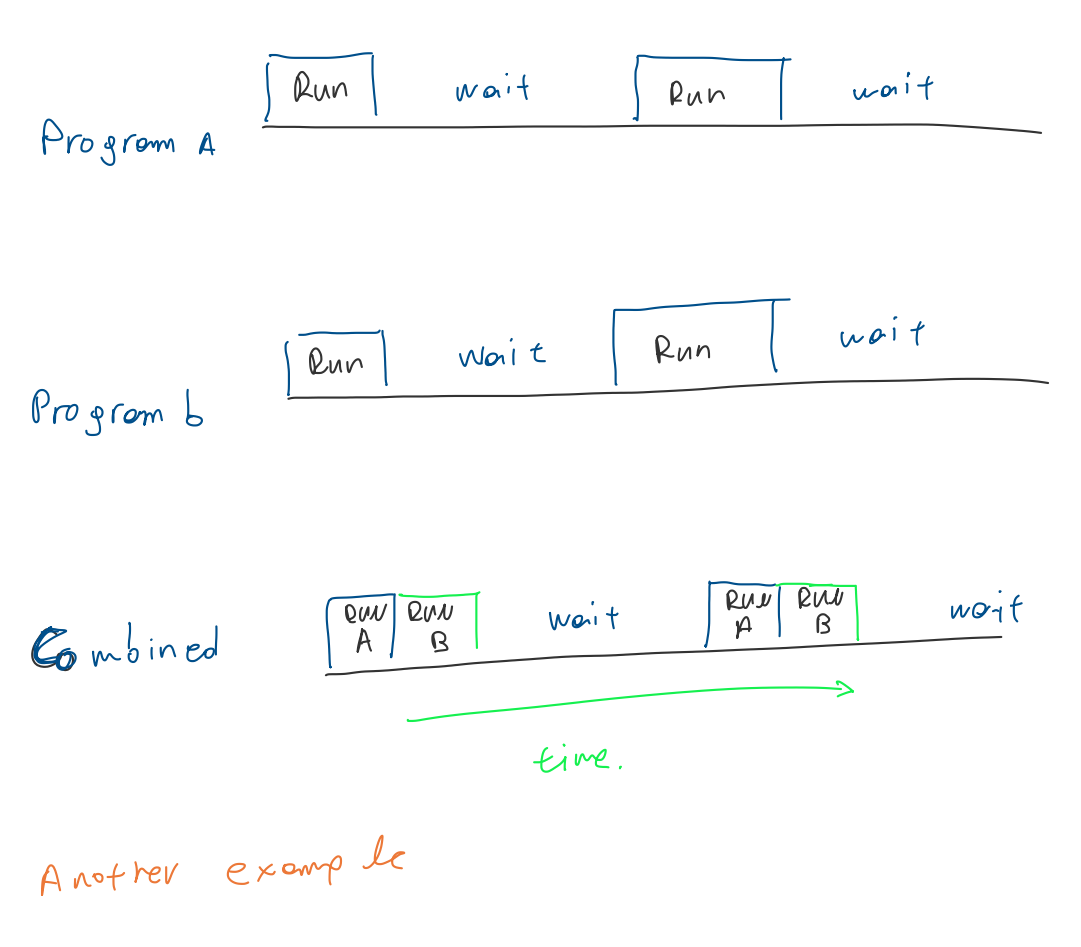
• UNIPROGRAMMING

- Processor must wait for I/O instruction to complete before processing

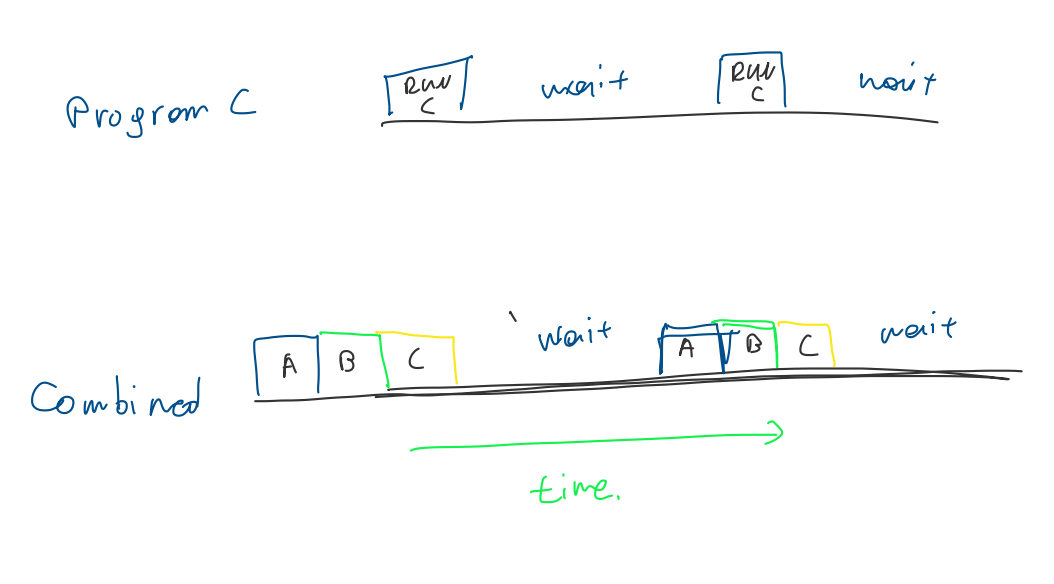

Run
wait
Run
wait
Program A.
Time

• MULTIPROGRAMMING

- When 1 job needs to wait for I/O, the processor can switch to the other job

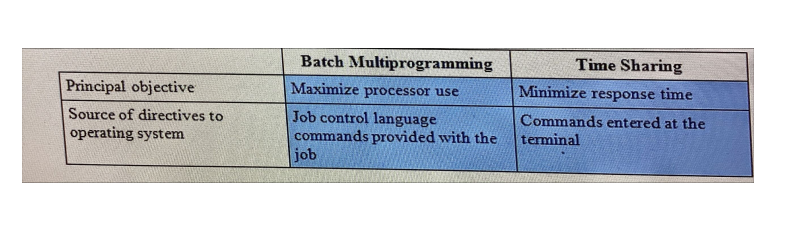

Program A
Run
wait
Run
wait
Program b
Run
wait
Run
wait
Combined
Run A
Run B
wait
Run A
Run B
wait
time.

Another example


Program C
Run C
wait
Run C
wait
Combined
A
B
C
wait
A
B
C
wait
time.

• TIME SHARING SYSTEMS

- Multiprogramming is used to handle multiple interactive jobs
- Processor's time is shared among multiple users
- Multiple users simultaneously access the system through terminals



	Batch Multiprogramming	Time Sharing
Principal objective	Maximize processor use	Minimize response time
Source of directives to operating system	Job control language commands provided with the job	Commands entered at the terminal

• MAJOR ACHIEVEMENTS

① Processor
② Memory managment
③ Information protection and security
④ Scheduling and resource managment
⑤ System Structure

PROCESSOR

- A program in execution
- An instance of a program running on a computer
- The entity that can be assigned to and executed on a processor
- A unit of activity characterized by:
  ① A single sequential thread of execution
  ② A current state
  ③ An associated set of system resources

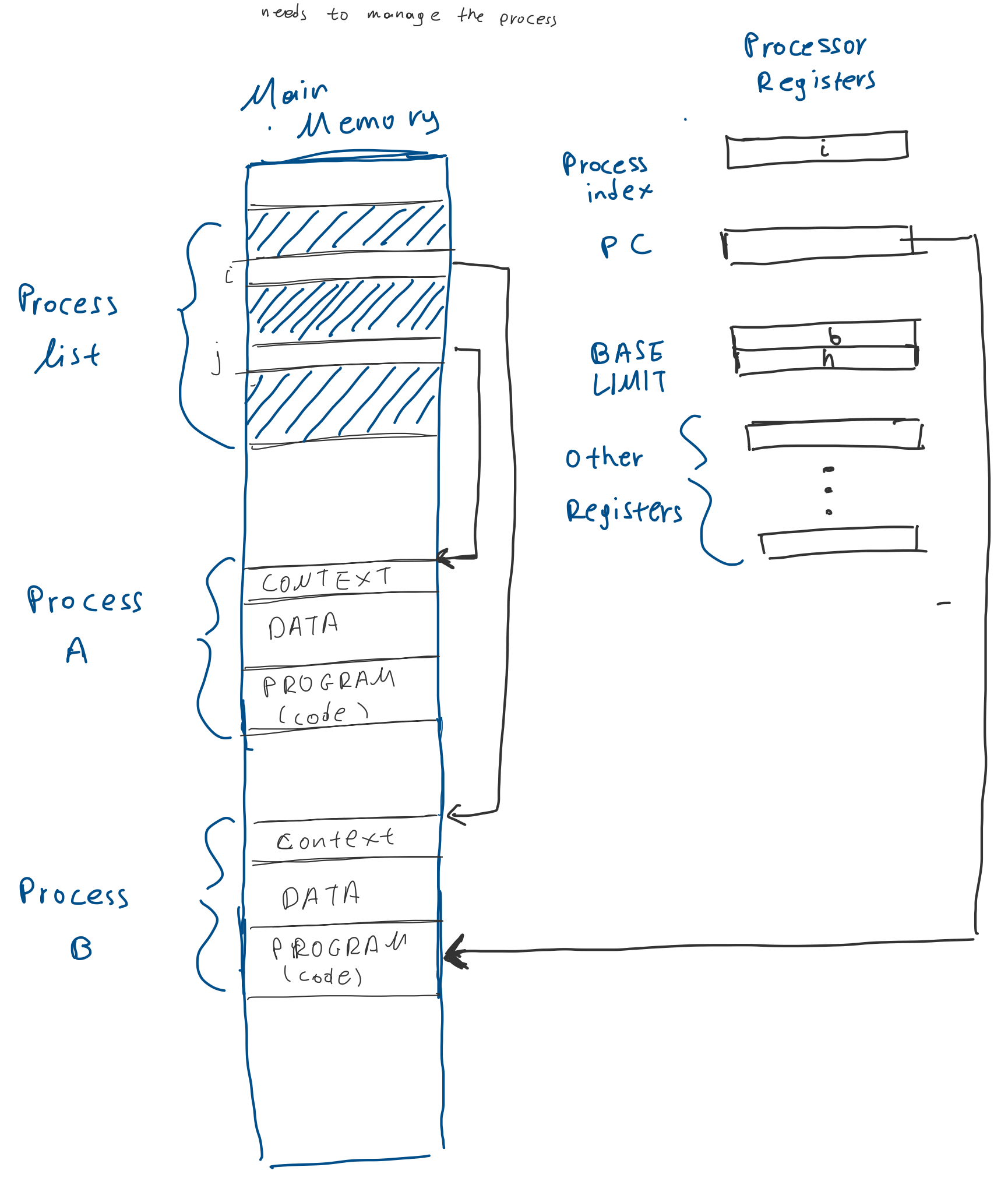
DIFFICULTIES WITH DESIGNING SYSTEM SOFTWARE

① Improper synchronization
② Failed mutual exclusion
③ Nondeterminate program operation
④ Deadlocks

PROCESS

Consist of 3 components

① An executable program
② Associated data needed by the program
③ Execution context of the program
- All information the operating system needs to manage the process


Main Memory
Processor Registers
Process index
i
PC
BASE
LIMIT
b
h
Other Registers
Process list
i
j
Process A
CONTEXT
DATA
PROGRAM (code)
Process B
Context
DATA
PROGRAM (code)

• MEMORY MANAGEMENT

- Process isolation → (set of different software and hardware tech. designed to protect ∀ process from other processes on the Operating System)
- Automatic allocation and Management
- Protection and access control

• VIRTUAL MEMORY

(Frees up RAM by swapping data that has NOT been used recently over to a storage device (f.e a SSD))

✓ Allows programmers to address memory from a logical point of view
- Another layer of indirection

✓ Allow the illusion of operating with a larger memory space than what is available in reality
- By storing some of the information on the file system

• PAGING.

① One way to implement virtual memory
② Allows the process to be comprised of a number of fixed-sized blocks, called pages
③ Virtual address is a page number and an "offset within the page

↗ Imagine we are trying to locate a house, ∀ house in the city has each own unique number, the number of houses will soon grow too big. Thus we introduce the concept of streets so that the numbers will be more managable "streets" = pages "houses" = offset

④ ∀ page may be located anywhere in main memory

⑤ Real address or physical address is the main memory address

INFORMATION PROTECTION AND SECURITY

✓ Availability:
- Concerned with protecting the system against interruption

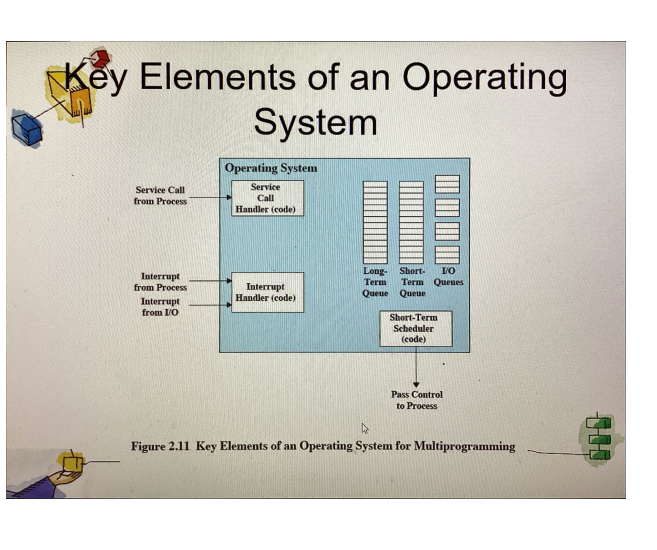
✓ Confidentiality
- Assuring that users cannot read data for which access is unauthorized

✓ Data integrity
- Protection of data from unauthorized modification

✓ Authentication
- Concerned with the proper verification of the identity of users and the validity of messages or data.

• SCHEDULING AND RESOURCE MANAGEMENT

- Fairness
  - Give equal and fair access to resources
- Differential responsiveness
  - Discriminate among different classes of jobs
- Efficiency
  - Maximize throughput, minimize response time, and accommodate as many uses as possible


Key Elements of an Operating System
Operating System
Service Call from Process
Service Call Handler (code)
Interrupt from Process
Interrupt from I/O
Interrupt Handler (code)
Long-Term Queue
Short-Term Queue
I/O Queues
Short-Term Scheduler (code)
Pass Control to Process
Figure 2.11 Key Elements of an Operating System for Multiprogramming

• MODERN OPERATING SYSTEMS

① Microkernel architecture
- Assigns only few essential functions to the kernel
  ① Address Spaces
  ② Interpross communication (IPC)
  ③ Basic scheduling

② Multithreading
- Process is divided into threads that can run concurrently
  - thread
    - Dispatchable unit of work
    - Executes sequentially and is interruptable

Process is a collection of one or more threads

③ Symmetric multiprocessor (SMP)
- There are multiple processors
- These processors share same main memory and I/O facilities
- All processors can perform the same functions

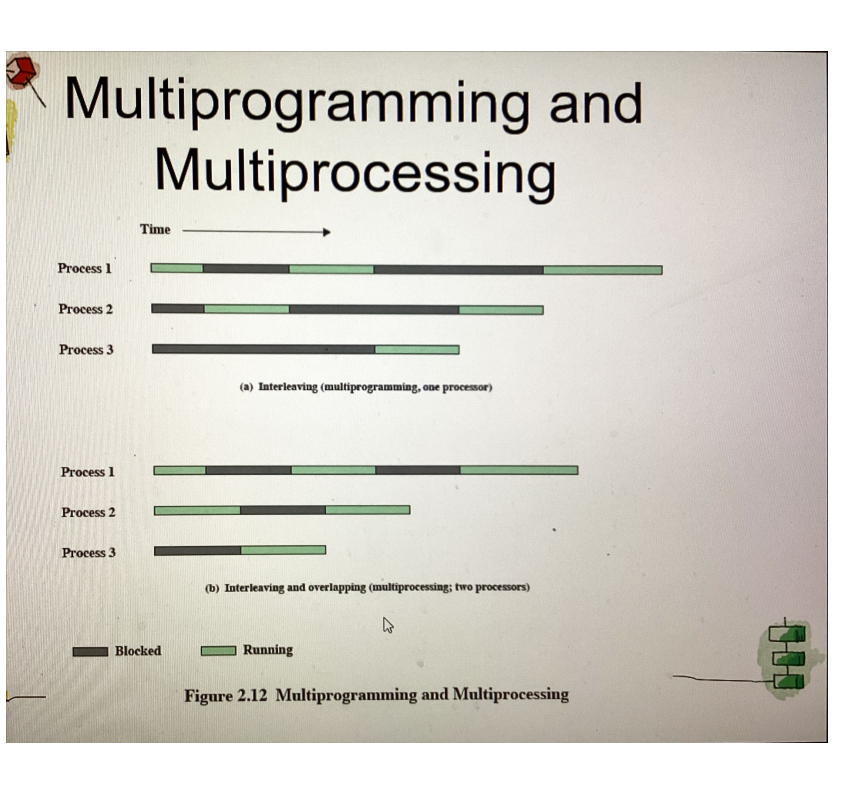

Multiprogramming and Multiprocessing

Figure 2.12 Multiprogramming and Multiprocessing

• MODERN OPERATING SYSTEMS

- Distributed operating systems
  - Provides the illusion of a single main memory space and single secondary memory space
- Object-oriented design
  - Used for adding modular extensions to a small kernel
  - Enables programmers to customize an operating system without disrupting system integrity