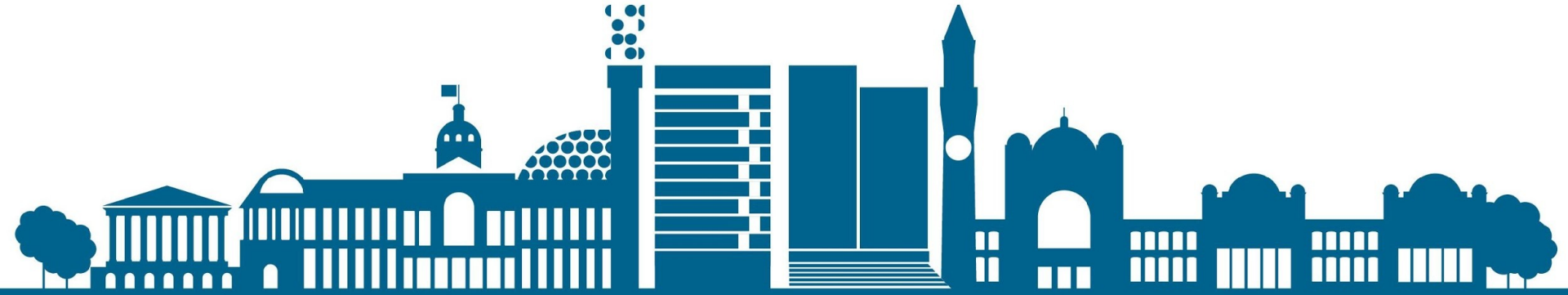
UNIVERSITY OF BIRMINGHAM

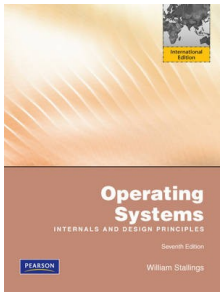
Computer Systems
Introduction to Operating Systems

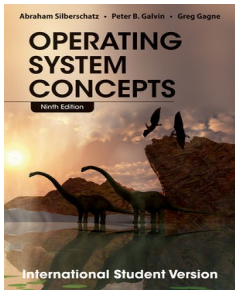


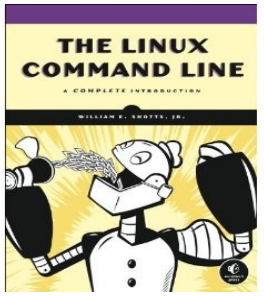
Lecture Objective

To develop a basic understanding of what is an **operating system** and why do we need it ?

Reading Material







- A lot of this course will be based upon material from the following two books:

Proper

- **Operating Systems: Internals and Design Principles** (7th edition) by William Stallings & Goutam Kumar

Easy

- **Operating System Concepts (9th edition)** by Silberschatz, Galvin & Gagne

- **The Linux Command Line:** A Complete Introduction (1st edition) by William Shotts (2012) (Electronic Book only)

- It is **recommended** that you read these to supplement what you will learn in the lectures

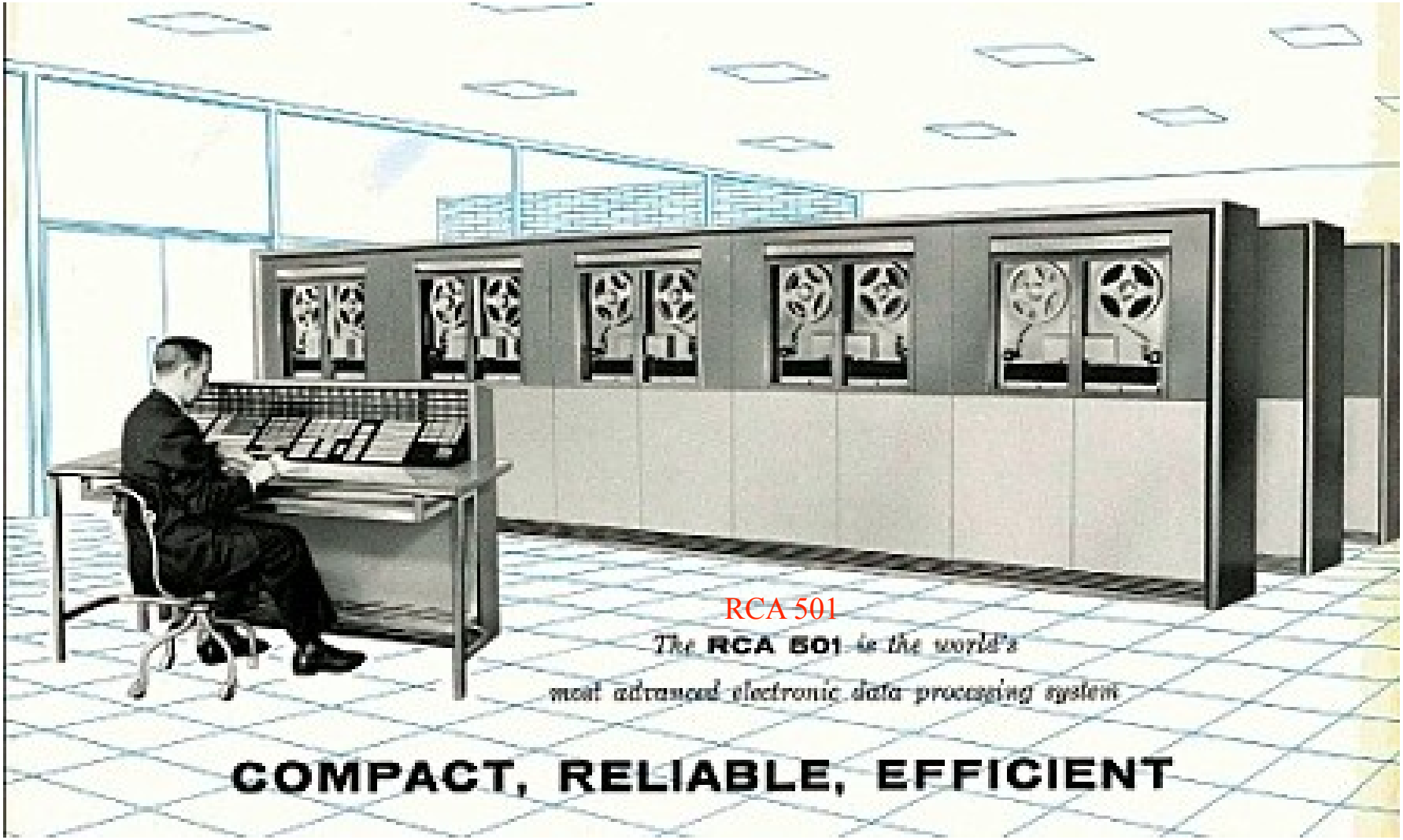
- There are also lots of great research papers available online - any material covered from these will be linked to from Canvas.

Why learn Operating Systems & Networks?

- A modern education in Computer Science would be incomplete without studying these topics
- Knowledge of the above strengthens the practical appreciation of other parts of your courses:
  - Software Workshop
  - Implementation of Data Structures & Algorithms
- Needless to say, it is very interesting!

A brief history of Computer Science at Birmingham

- Computing at University of Birmingham can be traced back to the KDF9 computing system
  - 16K words of memory
  - Magnetic tape decks for storage
  - Paper tape input and output


RCA 501
The RCA 501 is the world's
most advanced electronic data processing system
COMPACT, RELIABLE, EFFICIENT

A room plan for KDF9 deployment

1968: The KDF9 in the Aston Webb

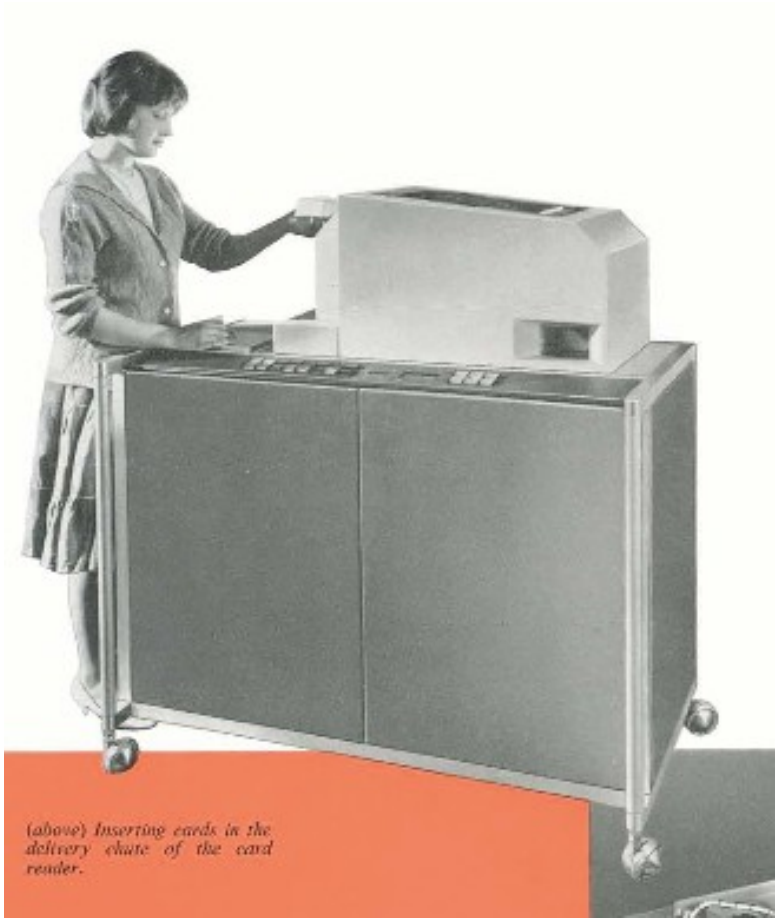
1968: An operator works on the KDF9

The Role of Human Operator

- Each program had to be keyed in by hand to control the system function required.
- There were few programs to read or write from a keyboard or a disk
- Distinct lack of monitors (screens) or keyboards
- The operators (the ladies in the pictures) controlled all
  - Loaded magnetic tapes
  - Fed in cards/paper tape
  - Maintained the system

The Role of Human Operator

- Operators worked in shifts
  - Fed in cards/paper tape
  - Approximately 40 hours per week, per shift.
  - In 1967, only a single operator.
  - By 1970, four shifts operated on the system
- Laborious
- Expensive
- Tedious
- Error-prone


(above) Inserting cards in the delivery chute of the card reader.

The Role of Human Operator

1976: The ICL 1906A (pictured) replaced the KDF9

- As time progressed, the University installed faster and more powerful computers.
- However, the limiting factor was the role of the operators
- An automatic system was required to free the operators from the mundane tasks
- Hence, the arrival of the **Operating System**

What is an Operating Systems?

Operating systems are those **programs** that **interface** the machine with the applications programs. The main function of these systems is to **dynamically allocate** the **shared** system **resources** to the executing programs. As such, research in this area is clearly concerned with the **management** and **scheduling** of **memory**, **processes**, and **other devices**. But the interface with adjacent levels continues to shift with time. Functions that were originally part of the operating system have migrated to the hardware. On the other side, programmed functions extraneous to the problems being solved by the application programs are included in the operating system.

—WHAT CAN BE AUTOMATED?: THE COMPUTER SCIENCE AND ENGINEERING RESEARCH STUDY,
MIT Press, 1980

What is an Operating System?

- An Operating System is a program or collection of programs that makes it easier for us to use a computer.
- An Operating System provides simpler abstraction of the underlying hardware.
- An Operating System is resource manager.

Examples:

- DOS, OS/2, Windows XP, Windows 2000
- Ubuntu, FreeBSD, Fedora, Solaris, Mac OS
- iOS, Android, Symbian OS, Lynx OS

What is an Operating System?

- It is a layer of system software that acts as an intermediary between a computer user and the computer's hardware.
- From a user's perspective:
  - It is designed for ease of use - it hides the hardware complexity:
    - Executing user programs
    - Convenient interface
- Resource Utilization does not appear to be important!

What is an Operating System?

- From the perspective of the system:
  - The OS is intimately involved with the hardware
  - It can be viewed as a Resource Manager (Allocator)
    - CPU time, memory space, file-storage space, I/O devices etc.
    - Requests for resources may be numerous, and conflicting.
- It can also be seen as a control system between the I/O devices and the user's programs.
- Managing the execution to prevent errors and improper use.

Objectives of an Operating Systems

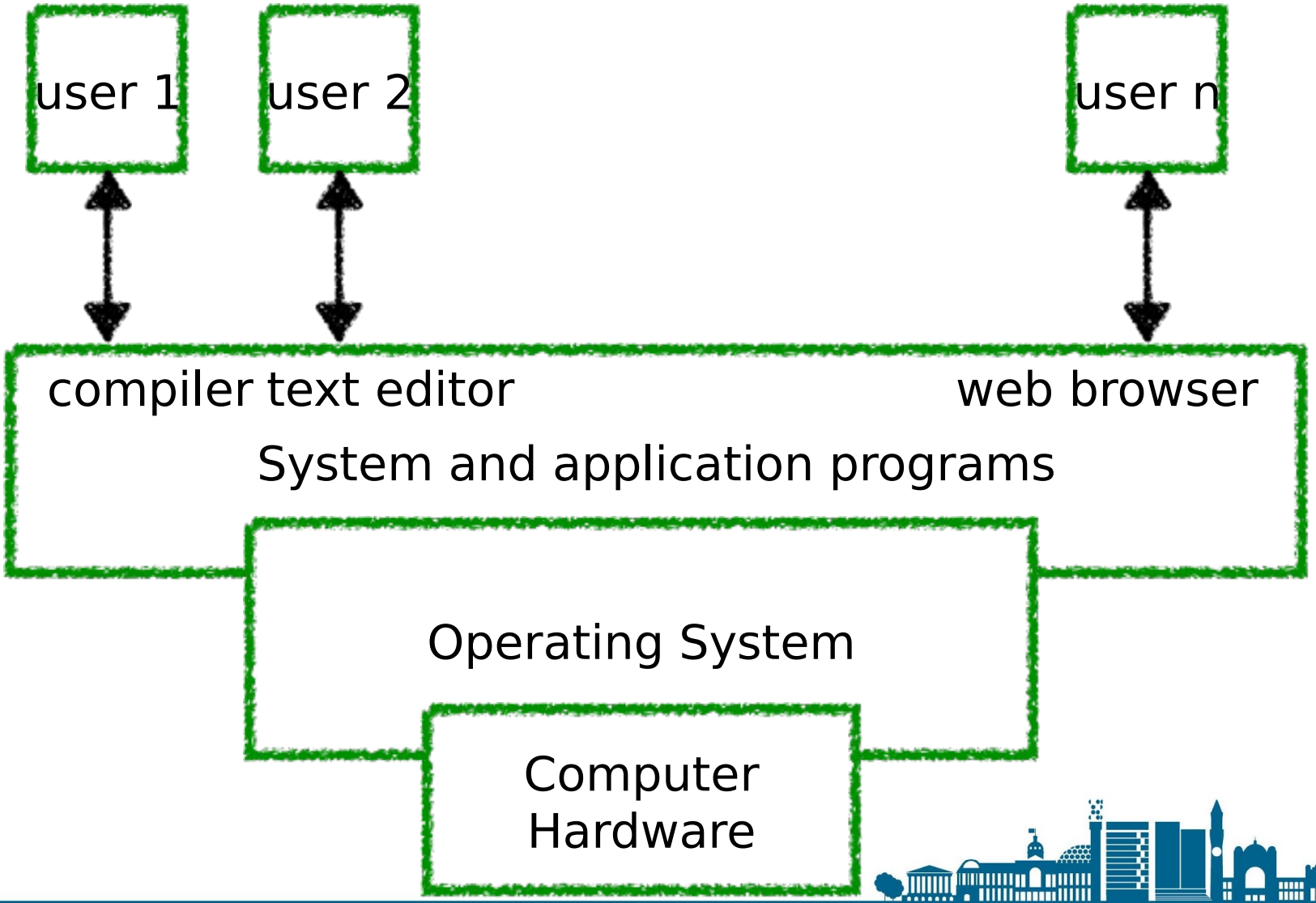
- A program that controls the execution of application programs
- An interface between applications and hardware

Main Objectives of an OS:
Convenience
Efficiency
Ability to Evolve To upgrade

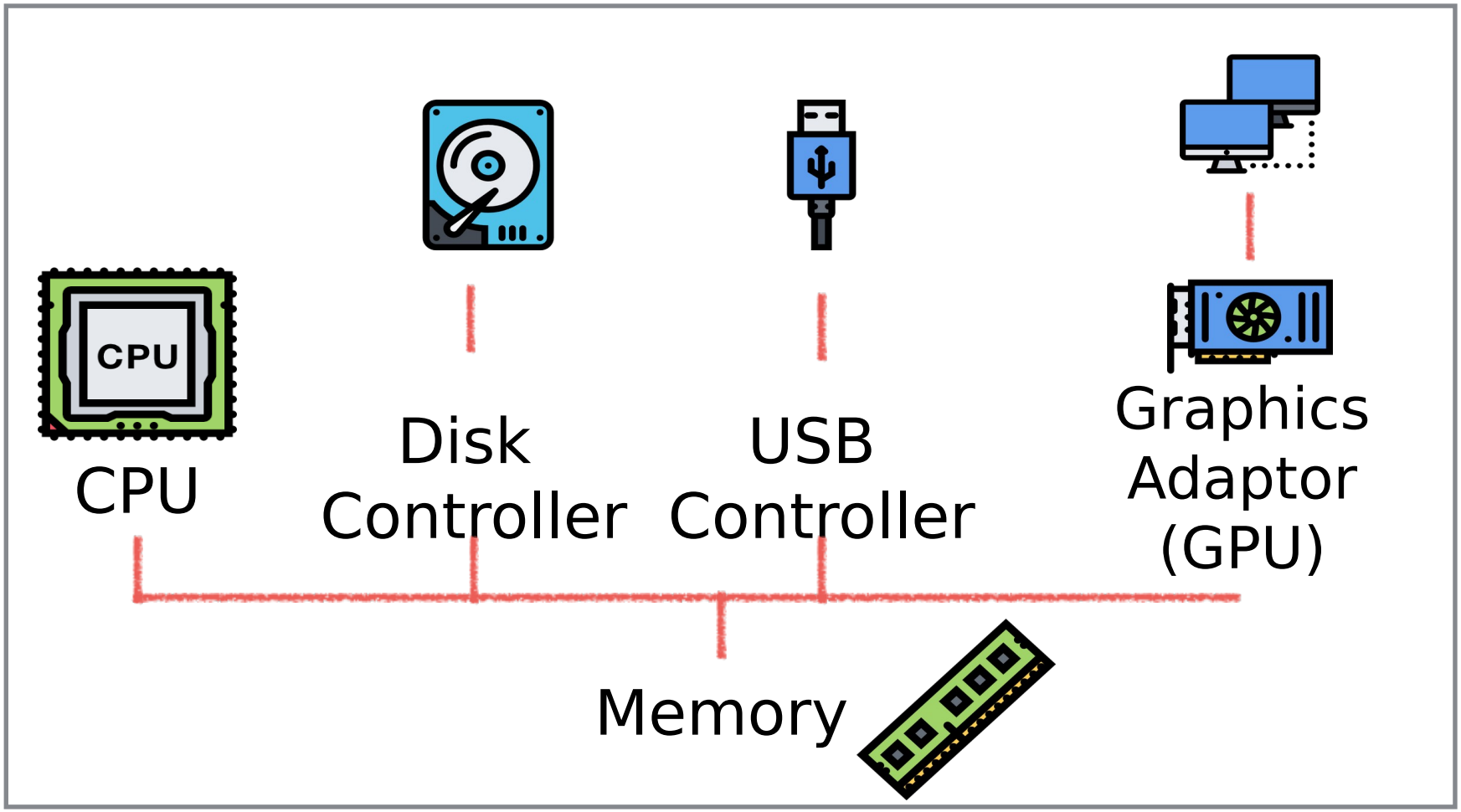
The Components of a Computer System

- **Hardware:** basic computing resources
- **Operating system:** controls and coordinates use of hardware amongst applications and users
- **Application programs:** define the ways in which the system resources are used to solve computing problems: compilers, text editors, email clients, webs browsers, games
- **Users:** people, machines and other computers

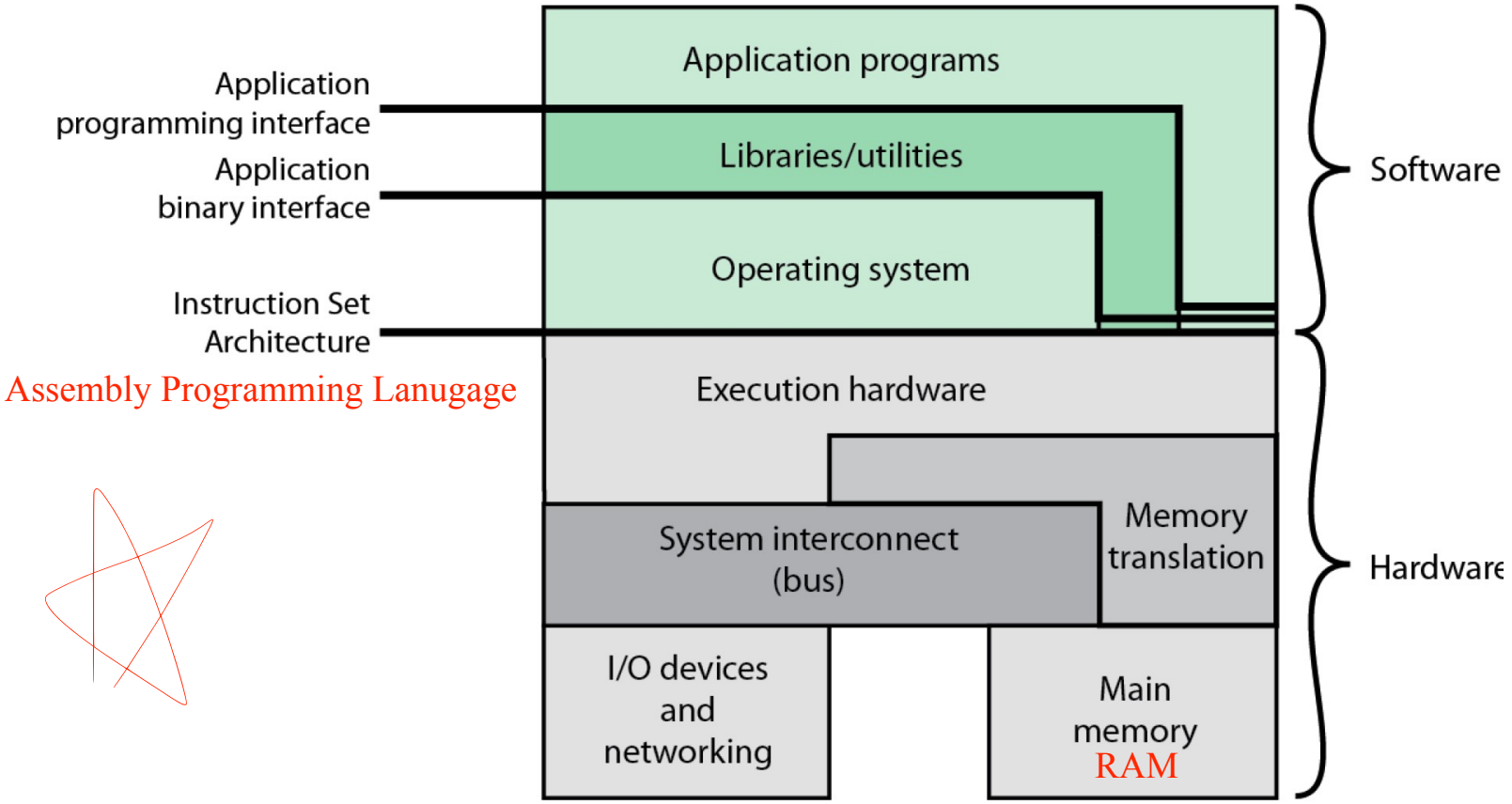
The Components of a Computer System



A Modern Computer System



Computer Hardware & Software Infrastructure



Operating System Services

- Program development
- Program execution
- Access I/O devices
- Controlled access to files
- System access
- Error detection and response

  Terminate or kill your programs when encounter errors
- Accounting

  Consumption

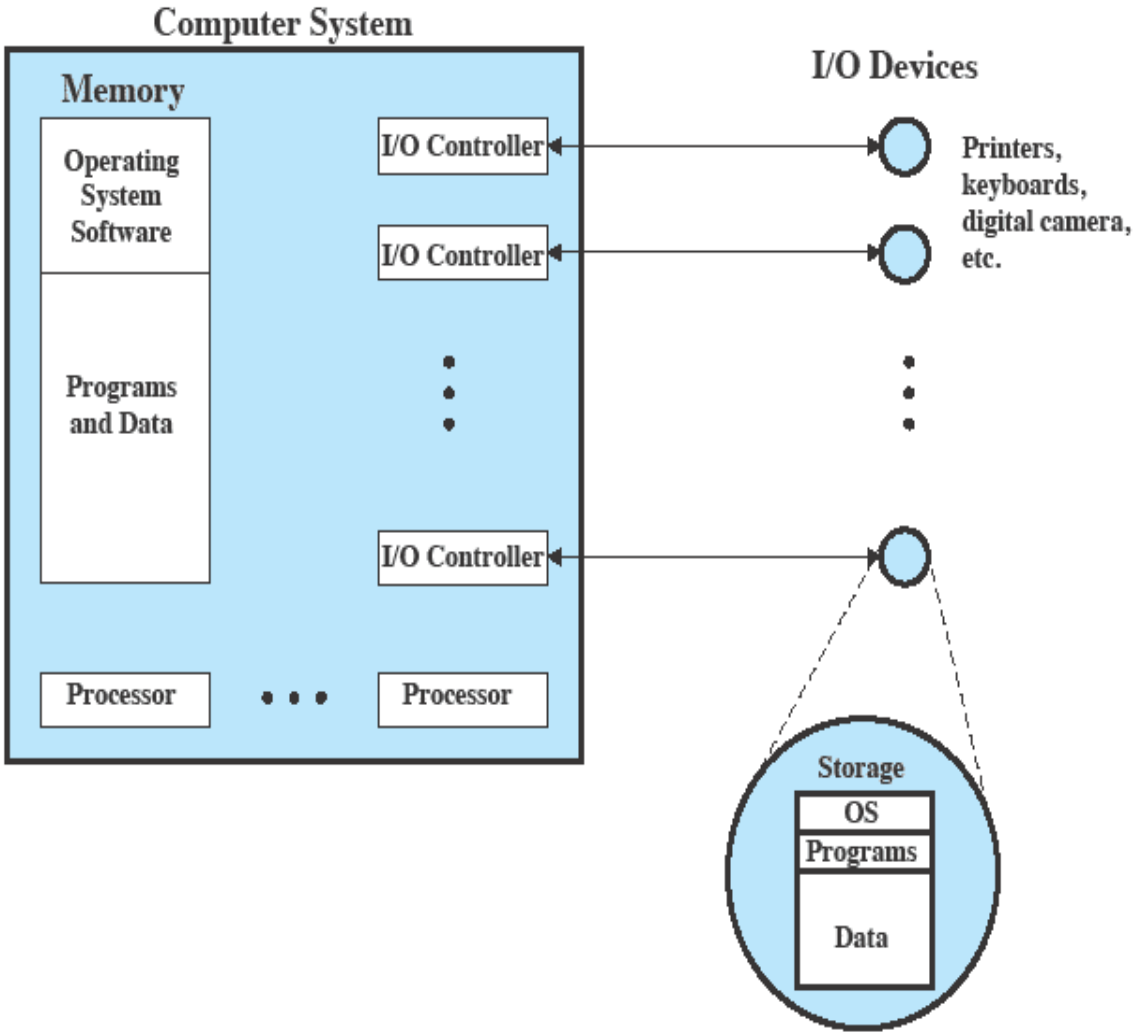
Role of an Operating System

- A computer is a set of resources for the movement, storage, and processing of data and for the control of these functions.
- The OS is responsible for managing these resources

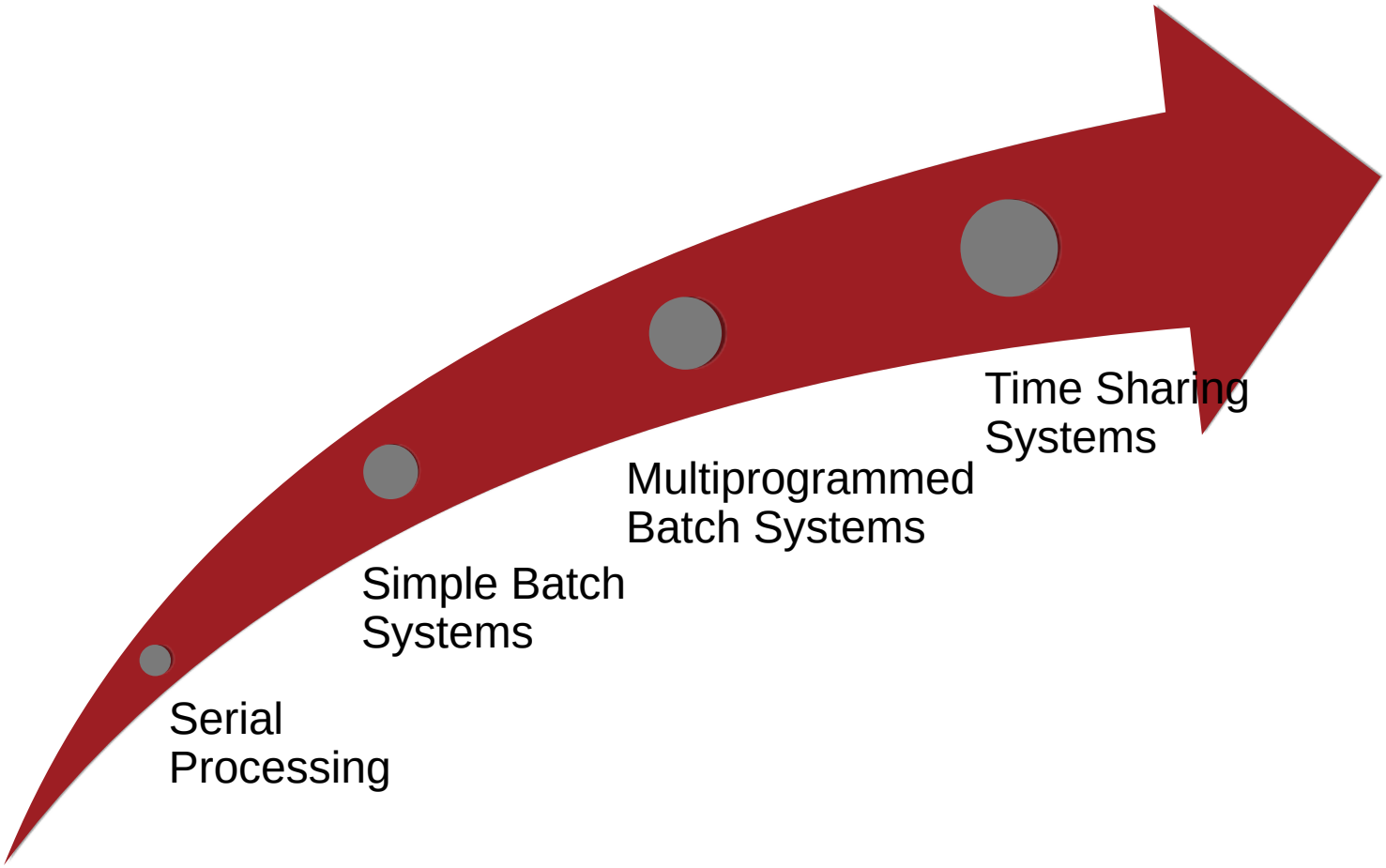
Normally, we think of a control mechanism as something external to that which is controlled.

Example: Heating System and Thermostat

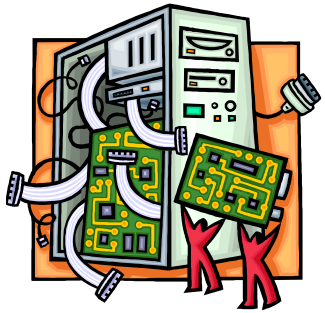
Operating System as a Resource Manager



Evolution Operating Systems



Serial Processing



EARLIEST COMPUTERS:

- No operating system
  - Programmers interacted directly with the computer hardware
- Computers ran from a console with display lights, toggle switches, some form of input device, and a printer
- Users have access to the computer in “series”

PROBLEMS:

Scheduling

- Most installations used a hardcopy sign-up sheet to reserve computer time
- Time allocations could run short or long, resulting in wasted computer time

Setup time

- A considerable amount of time was spent just on setting up the program to run

Simple Batch Systems

Early computers were very expensive

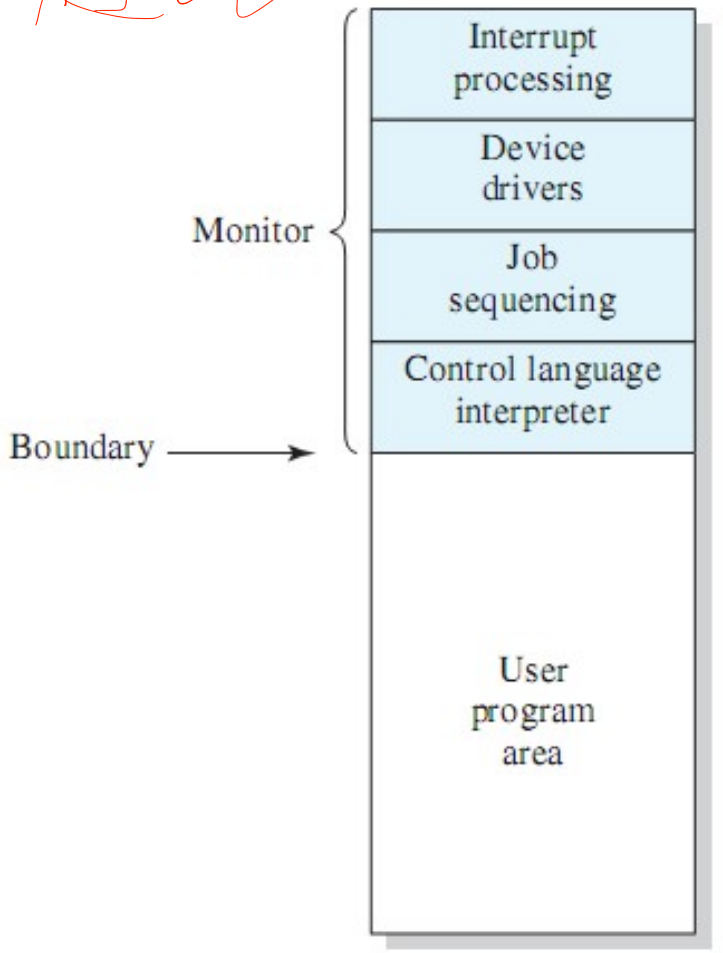
- Important to maximize processor utilization

Monitor

- User no longer has direct access to processor
- Job is submitted to computer operator who batches them together and places them on an input device
- Program branches back to the monitor when finished

Role of a Monitor

- Monitor controls the sequence of events
- Resident Monitor is software that always resides in memory
- Monitor reads in job and gives control
- Job returns control to monitor



Role of a Monitor

- Processor executes instruction from the memory containing the monitor
- Executes the instructions in the user program until it encounters an ending or error condition
- "control is passed to a job" means processor is fetching and executing instructions in a user program
- "control is returned to the monitor" means that the processor is fetching and executing instructions from the monitor program

Simple Batch Systems – Monitor Overhead

Processor time **alternates** between execution of **user programs** and execution of **the monitor**

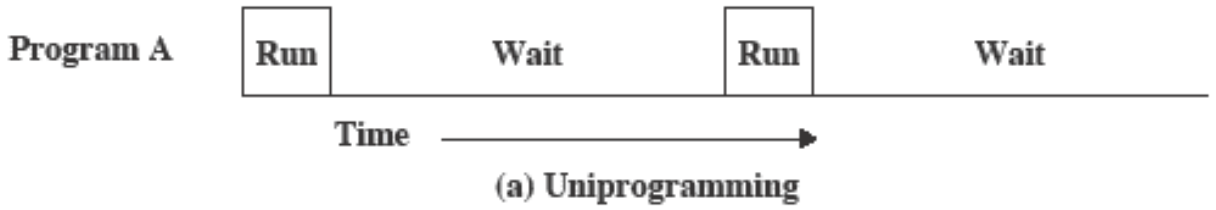
Sacrifices:

- Some main memory is now given over to the monitor
- Some processor time is consumed by the monitor

Despite overhead, the simple batch system improves utilization of the computer

Uniprogramming

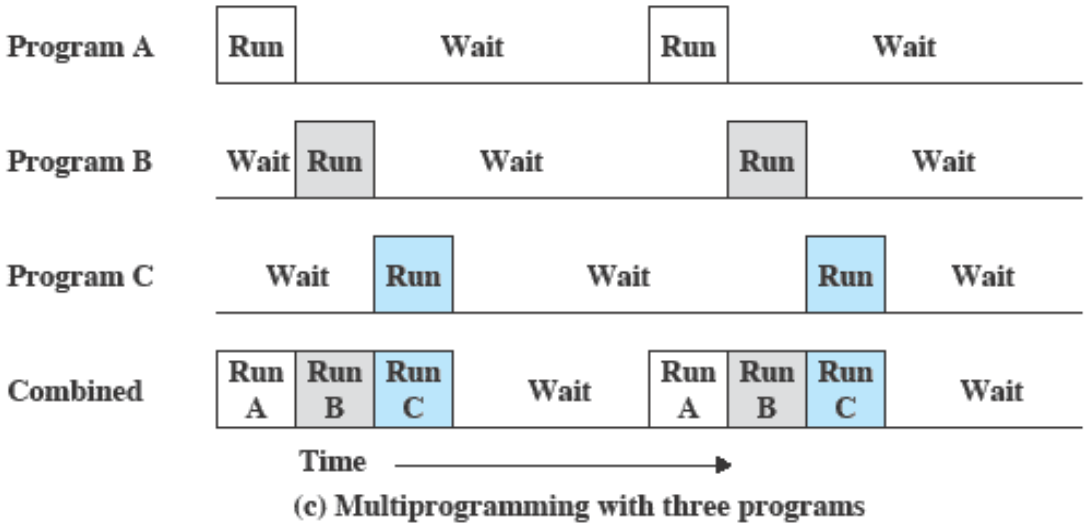
The processor spends a certain amount of time executing, until it reaches an I/O instruction; it must then wait until that I/O instruction concludes before proceeding.



(a) Uniprogramming

Multi-programming

There must be enough memory to hold the OS (resident monitor) and one user program
When one job needs to wait for I/O, the processor can switch to the other job, which is likely not waiting for I/O



(c) Multiprogramming with three programs

Time Sharing Systems

- Can be used to handle multiple interactive jobs
- Processor time is shared among multiple users
- Multiple users simultaneously access the system through terminals, with the OS interleaving the execution of each user program in a short burst or quantum of computation

	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

Summary

- The Role of Human Operator
- How monitor replaced the human operator and evolved into an Operating System
- Operating System as a Resource manager
- Evolution of Operating Systems from Serial to Time Sharing Systems

References / Links

- Chapter # 1: **Operating System Concepts** (9th edition) by Silberschatz, Galvin & Gagne

Read this