ANDIJAN STATE PEDAGOGICAL INSTITUTE

Lecture materials · 2026–2027 academic year

SUBJECT

Information Systems and Digital Technologies

Lecture 11 · Operating System Functions and File Management

Part 2 of 2 · 2 hours

Lecturer: **Ismailov Alisher**

E-mail: alisherismailov1991@gmail.com

Contact: **99-499-73-64**

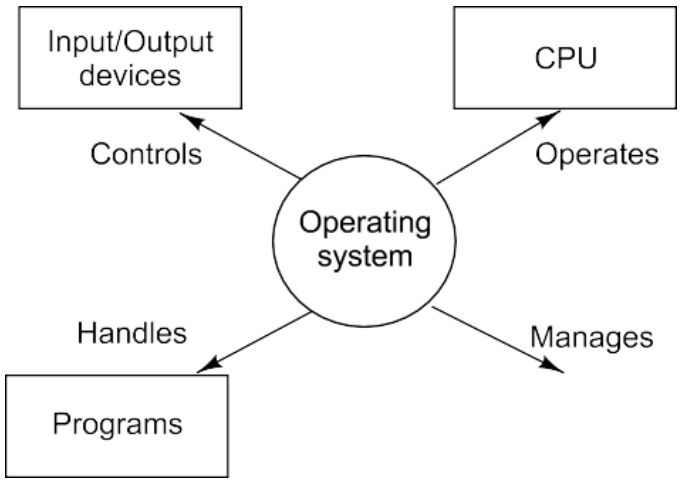
LECTURE 11 · 2 HOURS · PART 2/2

FUNDAMENTALS OF COMPUTERS · CHAPTER 11

Operating Systems

The master control program that makes the hardware useful

```
C:\> ver
process · memory · file · device · security
C:\> _
```



The roles of an operating system — textbook Fig. 11.1

By the end of this lesson, you can…

4. Compare the six types of operating system and the GUI and CLI interfaces.
5. Recognise key features of MS-DOS, UNIX and Windows.
6. List the services an OS gives to users and programs, and explain what a device driver does.
7. Explain dual-mode operation: why the CPU has a user mode and a kernel mode.
8. Compare open-source and proprietary operating systems, with examples.

Memory management: the right data in RAM

The OS gives main-memory space to processes and their data. This lets it:

- Improve performance
- Run many processes at once
- Share memory between processes

Execution time depends on having the data in main memory.

A GOOD MEMORY MANAGER ENSURES…

Correct relocation

Move data without disturbing running processes.

Protection

A process can only touch data it is allowed to.

Sharing

Processes can share information when needed.

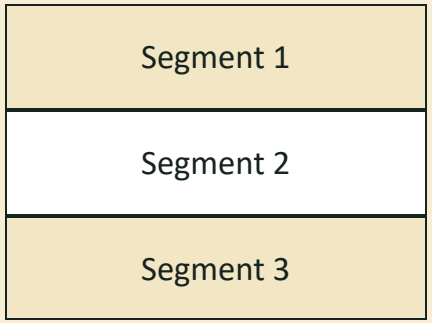
Using small free spaces

Defragmentation gathers small scattered gaps.

Three memory techniques

Segmentation

Memory split into blocks (segments). Each has a base address.



base + offset = actual address
`0100 H + 00AF H = 01AF H`

Paging

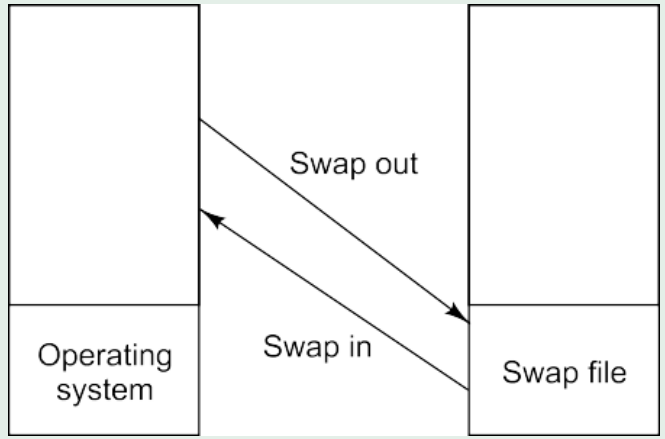
Memory split into equal-sized pages. A page table records where they are.



Virtual address = page number + offset
Shorter than the physical address.

Swapping

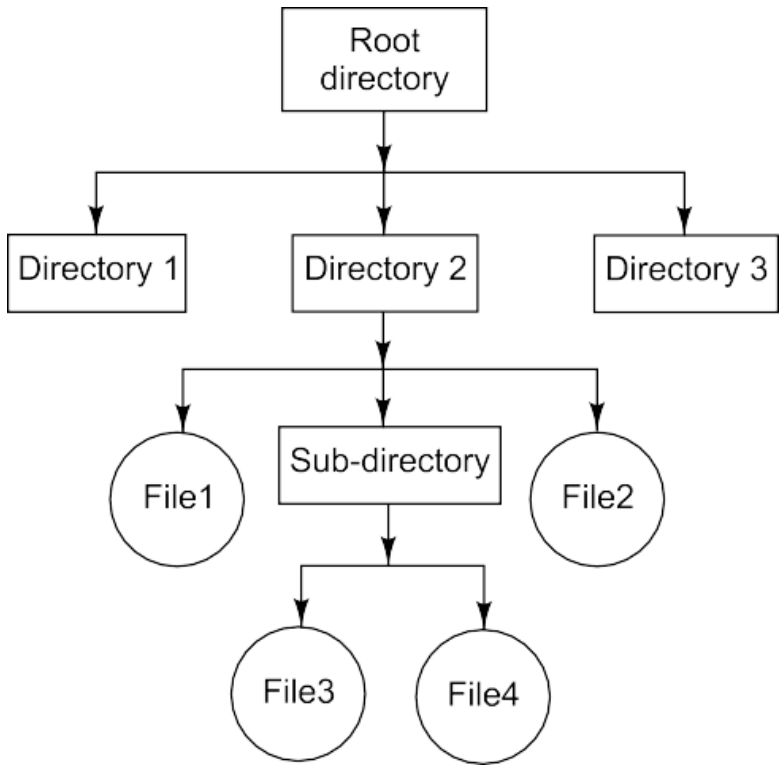
Move pages between RAM and a swap file on the hard disk.



Swap in: disk → RAM **Swap out:** RAM → disk

Swapping figure: textbook Fig. 11.11

File management: keeping files organised



General hierarchy of file storage — textbook Fig. 11.12

A file is a named collection of information. The OS helps you:

- Create files and put them in a location
- Find files quickly
- Share files between users
- Organise files in folders (directories)
- Change a file's content or name

File attributes

Name	**Type**
Location	**Size**
Date & time	**Read-only**
Hidden	**Archive**

Root directory → directories → subdirectories → files

What you can do with files

FILE OPERATIONS



Create



Save



Open



Modify



Close



Rename



Delete

ACCESS PERMISSIONS

R — **Read** — only read the content

W — **Write** — modify the content

X — **Execute** — run the file

FILE SYSTEMS

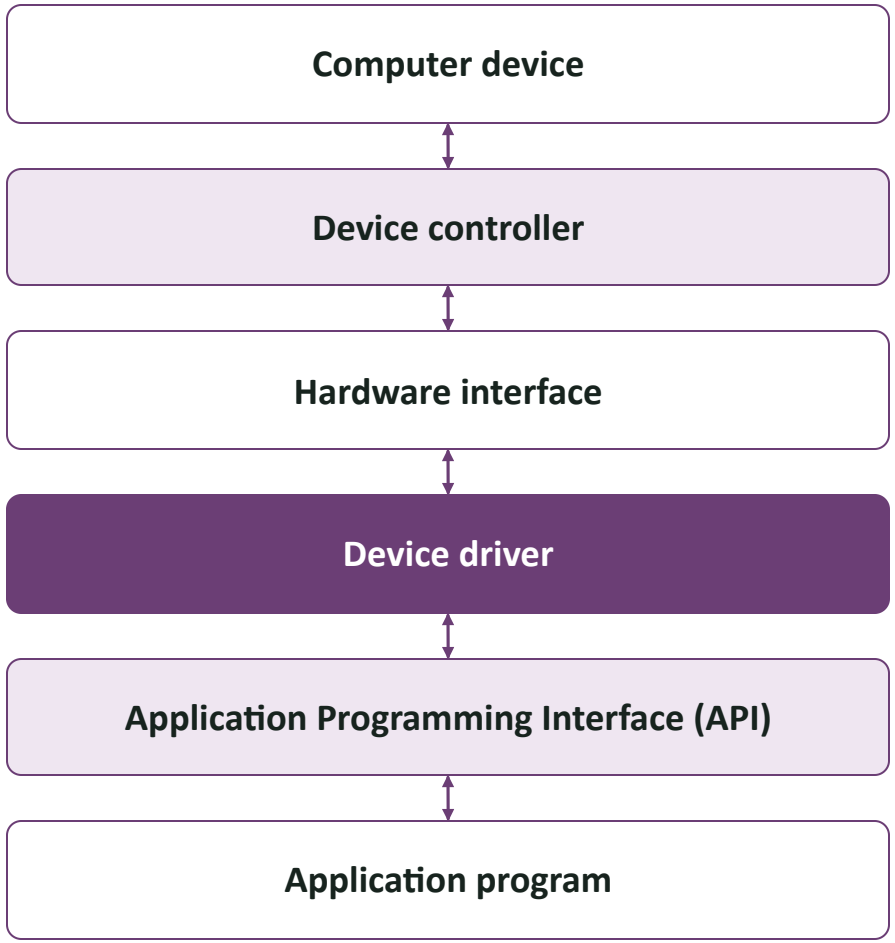
FAT

File Allocation Table. For smaller disks with simple folders. FAT12, FAT16, FAT32.

NTFS

New Technology File System. For large hard disks; fixes FAT's drawbacks.

Device management: drivers and controllers



Redrawn from textbook Fig. 11.13

Device controller

Hardware for each device, with buffer registers. Data passes only through controllers.

Device driver

Software that lets the OS talk to a controller.

API

Functions programs call to ask the kernel for device services.

Controller registers:
Command (what is asked) · **Status** (free or busy) · **Data** (what is moved)

Security: protecting inside and outside



Internal security

also called system protection

- Protects one process from another
- Only one process uses a resource at a time
- Least privilege: each process gets just enough rights — more privileges win a contested device

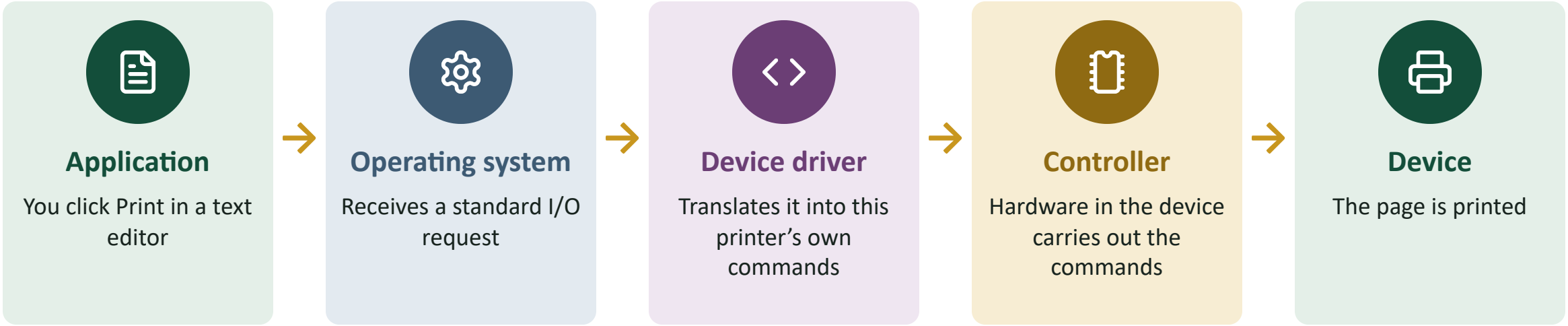


External security

also called system security

- Protects data, programs and resources from unauthorised access
- Most needed on a network or the Internet
- Main tool: the software firewall
- Also needed at physical, human and network levels

Device drivers: the OS's translators



Why drivers?

Thousands of different devices — the OS cannot know them all.

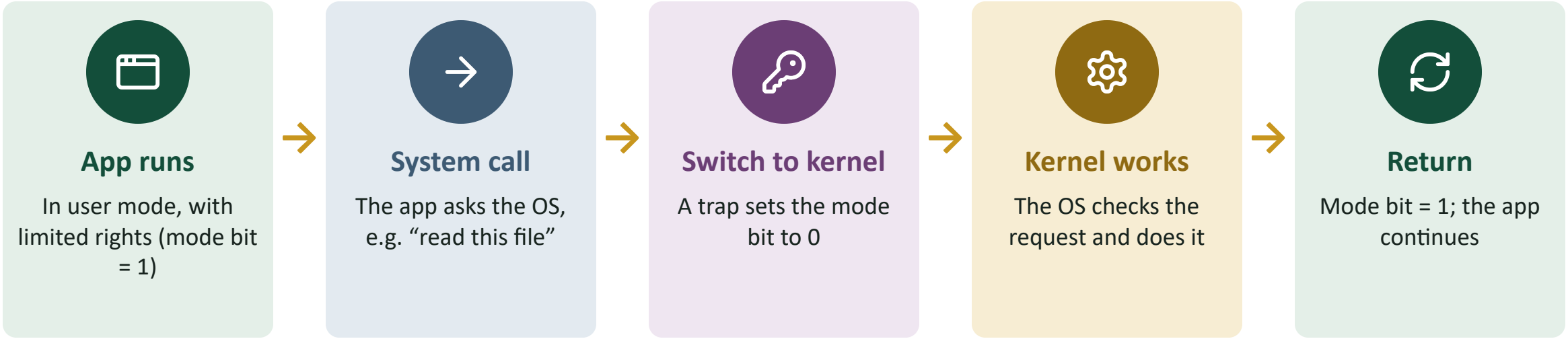
The maker writes a driver; the OS just loads it.

Getting drivers

Plug and play: the OS finds and installs many drivers itself.

Otherwise: the maker's website or OS updates.

Dual mode: user mode and kernel mode



Privileged instructions

Only allowed in kernel mode: direct I/O, timer control, memory-protection settings, halting the CPU.

Why two modes?

A buggy or malicious app cannot damage the OS or other programs.

If it tries, the OS stops that app.

User mode vs kernel mode

	User mode	Kernel mode
Mode bit	1	0
Who runs here	Applications: browser, games, office apps	The OS kernel and most device drivers
Access to hardware	Only through system calls	Direct, full access
Privileged instructions	Not allowed — trap to the OS	Allowed
If it crashes	Only that app closes	The whole system may stop

Remember: apps live in user mode and must ASK the kernel for anything risky.

CHECK FOR UNDERSTANDING

Memory, files, devices or security?

A A file is marked so that it can only be opened for reading.

B Pages not needed right now are moved to a swap file on the disk.

C The status register shows that the printer is busy.

D A firewall blocks an unknown program from reading your files.

E The actual address is found by adding the base address and offset.

F A large USB stick is formatted with a file system for small disks.

Bonus: name the exact term for each one.

Six types of operating system

Batch processing

Jobs grouped into batches and run one at a time, first come first served.

e.g. **Early 1950s systems**

Multi-user

Many users work on one powerful computer through terminals.

e.g. **UNIX, Linux, Windows 2000, VM-386**

Multitasking

One user runs many tasks at once by time slicing.

e.g. **UNIX, Linux, Windows 2000, XP**

Real-time

Critical tasks must finish on time: robots, spacecraft.

e.g. **MTOS, Lynx, RTX**

Multiprocessor

Several CPUs run processes at the same time, faster.

e.g. **Linux, UNIX, Windows 2000**

Embedded

Limited OS inside an electronic device.

e.g. **Palm OS, Windows CE**

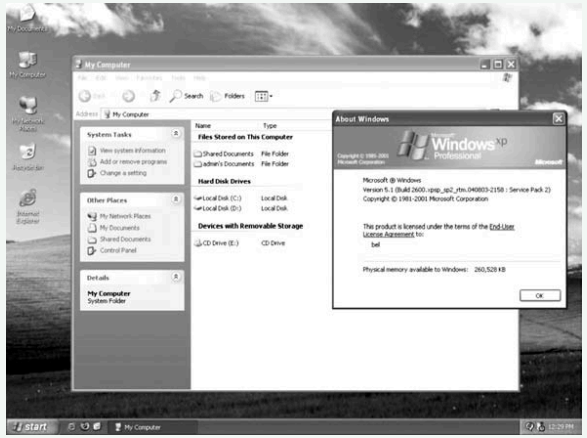
CHECK FOR UNDERSTANDING

Which type of OS?

- A spacecraft must fire its engine exactly on time.
- Forty terminals share one powerful central computer.
- Payroll jobs are grouped and run one by one overnight.
- A server keeps working when one of its four CPUs fails.
- A small handheld organiser runs Palm OS.
- You type an essay while music plays and a file downloads.

Two ways to talk to the OS: GUI and CLI

GUI — Graphical User Interface



Windows XP — textbook Fig. 11.14

- Point and click on icons
- Mouse, touch screen, light pen
- Easy for new users
- Confirms actions (e.g. before deleting)

User-friendly: no commands to remember.

CLI — Command Line Interface

```
C:\>copy c:\a1.txt d:\a1.txt
        1 file(s) copied.
C:\>_
```

Recreated from textbook Fig. 11.16 (MS-DOS)

- Type a command, press Enter
- The command line interpreter runs it
- Shows the prompt again with the output

Less friendly: you must remember many commands.

MS-DOS and Windows

MS-DOS · Microsoft, 1981

- Single-user, single-tasking; for Intel 8086 PCs
- Command line: the "DOS prompt"
- Runs only .COM, .BAT and .EXE files

`IO.SYS`	starts the computer
`MSDOS.SYS`	the kernel
`CONFIG.SYS`	configures hardware
`COMMAND.COM`	command interpreter
`AUTOEXEC.BAT`	runs at start-up

WINDOWS RELEASES (TEXTBOOK TABLE 11.1)

NT 3.1	1993	for business & science users
95	Aug 1995	plug and play, multitasking
98	Jun 1998	improved GUI, Windows Update
2000	Feb 2000	encrypted file system
ME	Sep 2000	system restore
XP	Oct 2001	Windows firewall
Server 2003	Apr 2003	better web services (IIS)
Vista	Nov 2006	multilingual UI, better search

Later: Windows 7, 8, 10 and 11 (modern note).

UNIX: kernel, shell and a tree of files

Kernel

The central part: controls hardware–software communication; manages processes, memory and devices.

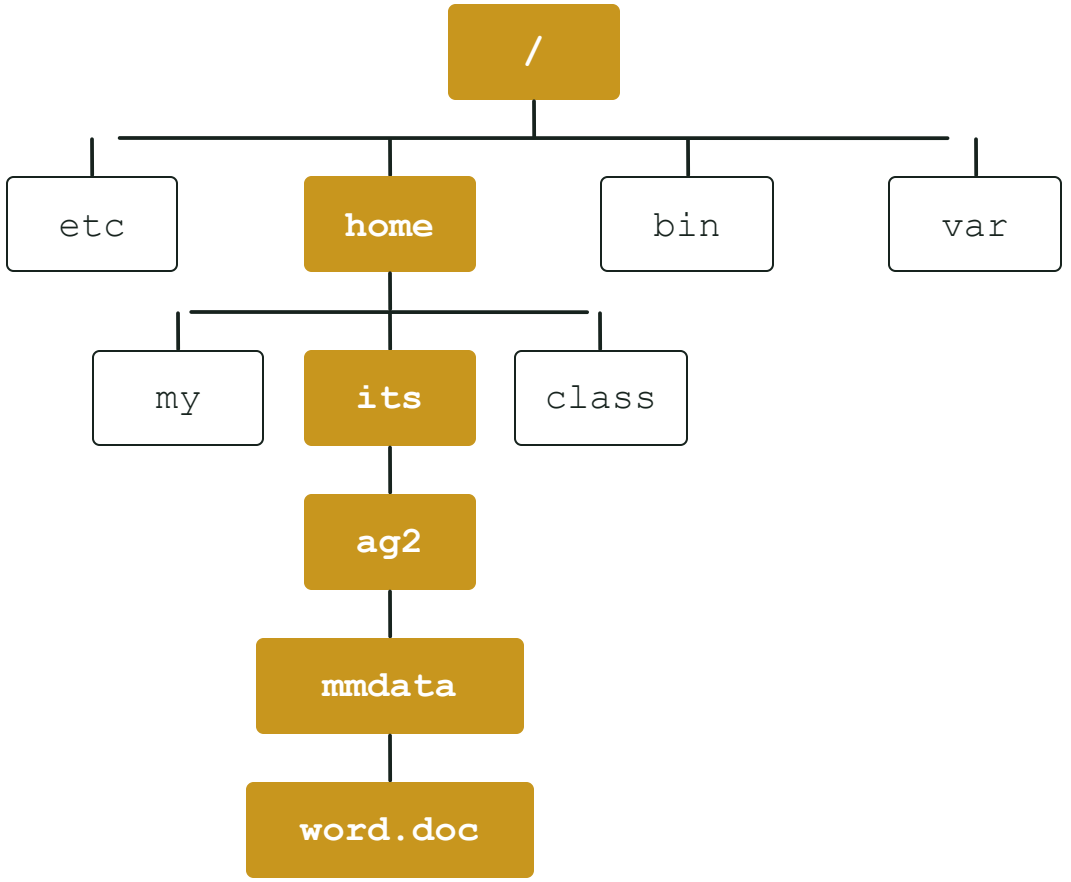
Shell

The user interface between user and kernel: takes commands and interprets them.

Files & processes

Everything is a file or a process. Each process has a unique PID.

- Many users at once
- Many programs at once
- Virtual memory: runs programs bigger than RAM




```
/home/its/ag2/mmdata/word.doc
```

Open source vs proprietary

Open source

source code is public

- Anyone may use, study, change and share it
- Usually free of charge
- Built by communities and companies
- Linux (Ubuntu, Debian, Fedora), Android (AOSP), FreeBSD

Proprietary

source code is secret

- Owned by one company; licence limits use
- Usually paid or bundled with a device
- Only the owner can fix or change it
- Windows, macOS, iOS

Key idea: “free” in open source means freedom to see and change the code, not only zero price.

Open source runs much of the world

Smartphones

Android, built on the Linux kernel, is the most used mobile OS.

Web servers and cloud

Most websites and cloud services run on Linux.

Supercomputers

All of the world's top 500 supercomputers run Linux.

Desktops and schools

Ubuntu, Linux Mint; ChromeOS laptops are based on Linux.

Small devices

Routers, smart TVs, cars and Raspberry Pi boards.

How it is made

Thousands of developers and companies improve the code together.

CHECK FOR UNDERSTANDING

Services, modes and licences

01 A game asks the OS to save your progress to a file.

02 The OS shows “Printer out of paper”.

03 A new webcam does not work until you install software from its maker.

04 The mode bit is 0 and any instruction can run.

05 Your browser runs with limited rights.

06 You download Ubuntu, change it and share your version.

Choose from:

- System call
- Error detection
- Device driver
- Kernel mode
- User mode
- Open source

Operating systems around you



TEXTBOOK

School laptop running Windows

→ **Multitasking OS with a GUI**



TEXTBOOK

Linux server with many users

→ **Multi-user OS**



TODAY

Your phone (Android or iOS)

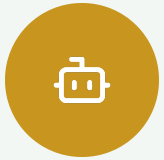
→ **Multitasking OS, touch GUI**



TODAY

Smart TV or washing machine

→ **Embedded OS**



TEXTBOOK

Factory robot arm

→ **Real-time OS**



TEXTBOOK

Typing commands at a prompt

→ **Command Line Interface**

Review: can you answer these?

01 Why is the OS called the resource manager?

02 Name the seven process states.

03 What is a PCB, and where does the OS keep all of them?

04 How do SJF and Round Robin choose the next process?

05 Give the three conditions that can cause deadlock.

06 What is the difference between swapping in and swapping out?

07 What is the job of a device driver?

08 Give one advantage of a GUI and one disadvantage of a CLI.

Exit ticket

Answer on a slip of paper before you leave.

1. Name three of the five functions of an operating system.

2. A process has been created and loaded into memory, but has not run yet. Which state is it in?

3. An OS controls a medical robot that must react within a set time. Which type of OS is it?