Operating systems

1.5.1 · Computer systems (Paper 1) · revision notes

The purpose and functions of an operating system: the user interface, memory management and multitasking, peripheral management and drivers, user management and file management. Questions ask you to identify a function from a description and to describe or explain how the operating system carries it out.

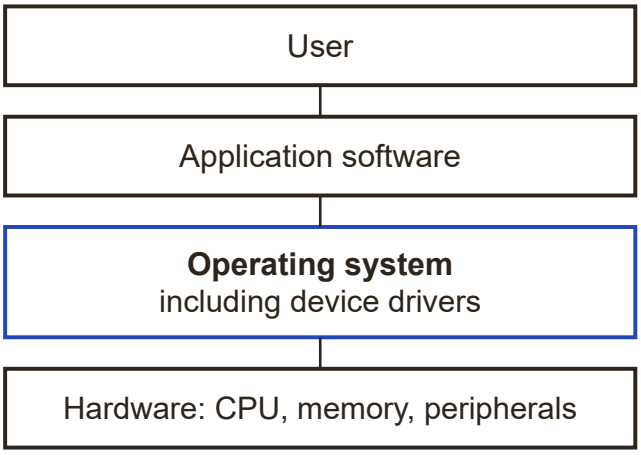
Key facts

- OS: manages hardware and software; provides a user interface and a platform for applications
- GUI: windows, icons, menus, pointer; easy to use but needs more resources
- Memory management: load into RAM, separate areas for each program, free memory, virtual memory
- Driver: translates OS instructions for a specific device
- Functions: user interface, memory management and multitasking, peripheral management and drivers, user management, file management
- Command line: typed commands; fast for experts, can be scripted, few resources
- Multitasking: processor time shared between programs
- User management: accounts, logins, access rights. File management: folders, naming, moving, permissions

Notes

Purpose and user interface

- The **operating system (OS)** manages the computer's hardware and software, and provides a **platform** for applications and an **interface** for the user.

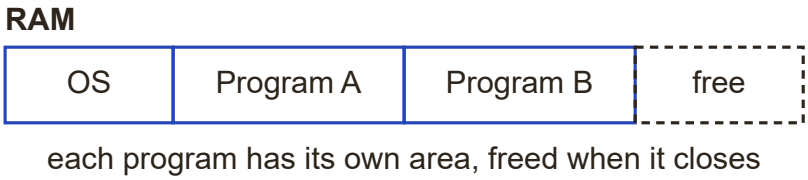


The OS sits between applications and the hardware

- A **GUI (graphical user interface)** uses windows, icons, menus and a pointer. It is easy to learn and suits visual tasks, but uses more memory and processing power.
- A **command-line interface** needs typed commands. It is harder to learn, but fast for experts, can run scripts to automate tasks, and uses few resources.
- Other interfaces include **menu-driven** interfaces (e.g. a cash machine) and **voice** interfaces (e.g. a smart speaker).

Memory management and multitasking

- The OS copies programs and data from secondary storage into **RAM** when they are needed, and gives each program its own area of memory so programs do not overwrite each other.
- It frees memory when a program closes, and moves data to and from **virtual memory** when RAM is full.



Memory management: each program gets its own area of RAM

- **Multitasking**: the OS shares processor time between programs, switching between them so quickly that they appear to run at the same time.

Peripheral management and drivers

- Data is transferred between **peripherals** (e.g. printers, keyboards, monitors) and the processor, and the OS manages this transfer, e.g. by managing a print queue.
- A **device driver** is software that translates the OS's general instructions into commands that a specific device understands. A new device needs the right driver, and drivers are updated to fix bugs or to work with a new version of the OS.

User management and file management

- **User management**: creates and deletes user accounts, checks usernames and passwords when users log in, sets access rights, and keeps each user's files and settings separate.
- **File management**: organises files into folders (directories), and lets users save, name, rename, move, copy, delete and search for files. It also sets file permissions, e.g. read-only.

How to answer each type of question

Identify the OS function being used

1 mark each | **Grade 4**

1. Look for the key idea: 'several programs at once' = memory management and multitasking; 'printer' or 'driver' = peripheral management; 'log in' = user management; 'folders' = file management.
2. Use OCR's names for the functions.

Example. Identify the function of the operating system that is being used in each case.
(a) A user logs in with a username and password. [1]
(b) A photo is moved into a new folder. [1]
(c) A new webcam is plugged in and the OS communicates with it using a driver. [1]

Model answer.

(a) User management (1)
(b) File management (1)
(c) Peripheral management (1)

Describe how the OS manages memory

3 to 4 marks | **Grade 5**

1. Loading programs and data from secondary storage into RAM.
2. Giving each program its own area of memory and keeping them separate.
3. Freeing memory when programs close.
4. Using virtual memory when RAM is full.

Example. Describe how the operating system manages memory. [4]

Model answer. It copies programs and data from secondary storage into RAM when they are needed (1). It allocates an area of memory to each program (1) and keeps the programs separate so they do not overwrite each other's data (1). When a program is closed, the OS frees its memory so it can be used by other programs (1).

Explain the need for a device driver 2 to 3 marks **Grade 5**

1. Say that each device works differently, so the OS cannot control every device on its own.
2. Say that the driver translates the OS's instructions into commands for that device.

Example. Explain why a printer needs a device driver. [2]

Model answer. Each printer model works differently, so the operating system cannot know how to control every one (1). The driver translates the operating system's instructions into commands that this particular printer understands (1).

Compare or justify a user interface 2 to 4 marks **Grade 6**

1. Link the interface to the user: beginner or expert?
2. Give the feature (1 mark) and its benefit for this user (1 mark).

Example. A network manager uses a command-line interface to manage a school's servers. Explain **two** reasons why. [4]

Model answer. The manager can type commands directly without going through menus (1), which is faster for an expert user who knows the commands (1). Commands can be saved in scripts (1), so repetitive tasks, such as creating 200 student accounts, can be automated (1).

Don’t lose marks

- Confusing the OS with applications: word processors and browsers are application software, not the OS.
- Saying a single processor core runs several programs at exactly the same time. The OS switches between them very quickly.
- Describing a driver as hardware. It is software.
- Giving vague answers such as 'it manages files'. Say what it does: folders, naming, moving, deleting, permissions.

More tips

Memory tricks

- The five functions in order: 'Ugly Monkeys Play Under Furniture' = User interface, Memory management, Peripheral management, User management, File management.
- A driver is a translator between the OS and one device.
- GUI = pointing and clicking; command line = typing.

Exam technique

- Use the function names exactly as OCR does: user interface, memory management and multitasking, peripheral management and drivers, user management, file management.
- In 'describe' questions, give separate actions, each worth a mark.
- In scenario questions, link each function to what the user is doing.

What each grade needs

What you need to be able to do, from the first marks up to the top grade.

Grade 3

State the purpose of an operating system

It manages the hardware and software, and provides a user interface and a platform for applications.

Grade 4

Identify the functions of an operating system

User interface, memory management and multitasking, peripheral management, user management, file management.

Grade 5

Describe memory management and multitasking

Loads programs into RAM, gives each its own area, frees memory, and shares the processor between programs.

Grade 5

Explain the role of a device driver

Software that translates the OS's instructions into commands a particular device understands.

Grade 6

Compare a GUI with a command-line interface

A GUI is easier for beginners; a command line uses fewer resources and is faster for experts.

Grade 7

Explain several OS functions in a scenario

E.g. explain how the OS lets a user stream music, write an essay and print at the same time.

Free notes, Full packs, practice papers and mock exams at chhetriacademy.com. Written for this site; not taken from past papers. Stuck? 1-to-1 tutoring with a free 30-minute trial lesson.