Grade 12 A/L ICT · Lesson 5

5.2 Files, Directories and Storage Allocation

Syllabus 5.2. The OS organises your data as **files** and **directories**, using a **file system** (for example FAT-style). In this part you learn **file types**, the **hierarchy**, **security** (passwords and privileges), how space is given out with **contiguous, linked and indexed allocation**, why **defragmentation** helps, and why disks are **formatted**.

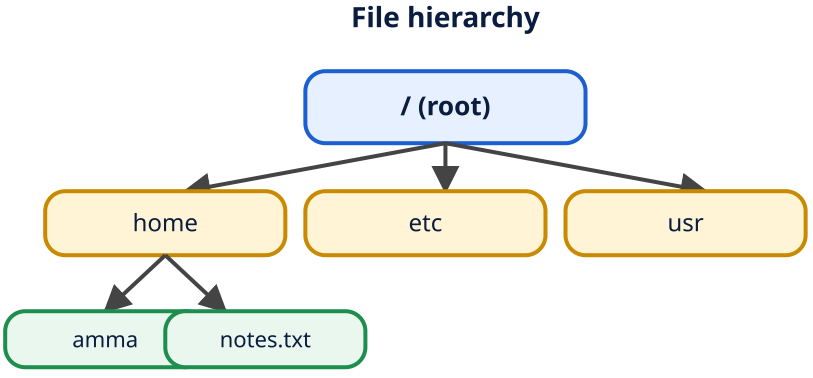
1. Files, types and attributes

A **file** is a named collection of related bytes. The **extension** (.exe, .jpg, .txt, .docx, .mp4) hints at the **type** so the OS and apps know how to open it. **Attributes** include name, size, type, location, creation/modification time, and access permissions.

> 💡 Extensions are conventions. Renaming `photo.jpg` to `photo.txt` does not turn pixels into text – it only changes the label the OS shows.

2. Directory hierarchy and file systems

Directories form a **tree** from a **root**. A path like `/home/amma/notes.txt` walks from root to the file. A **file system** (FAT, NTFS, ext4, APFS, …) is the on-disk layout of directories, files and free space. Older syllabuses often mention **FAT** as a classic example.



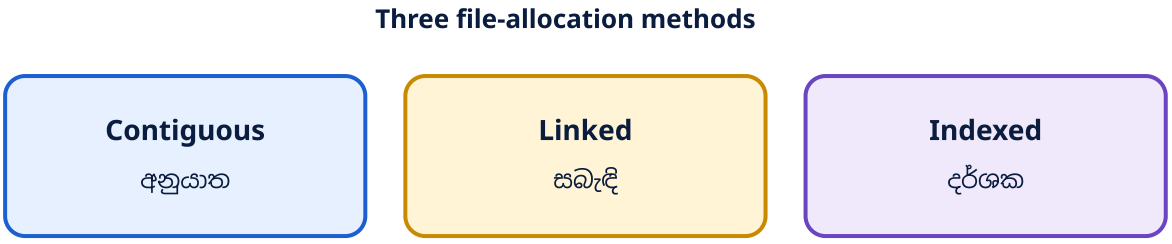
A tree of directories and files

3. File security

- **Passwords / accounts** control who can log in.
- **Access privileges** (read, write, execute; owner/group/others) limit what each user may do to a file.
- The OS **enforces** these rules on every open, read and write.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

4. Storage allocation methods



Contiguous, linked and indexed allocation

Contiguous
File occupies a run of neighbouring blocks. Fast sequential read; suffers **external fragmentation**.

Linked
Each block holds a pointer to the next. Easy to grow; slow random access.

Indexed
An **index block** lists pointers to all data blocks. Good for random access.

🧱 **Worked example: A disk has 10 blocks. Files A=2, B=3, C=1 blocks are stored contiguously from block 0. Where does each file live, and how many blocks stay free?**
A: blocks **0–1**; B: **2–4**; C: **5–5**; free = **4** blocks.

File	Start block	End block
A	0	1
B	2	4
C	5	5

🧱 **Worked example: A file needs 5 data blocks with one-level indexed allocation. How many blocks are used in total (index + data)?**
1 index block + 5 data blocks = **6** blocks.

5. Defragmentation and formatting

Fragmentation spreads a file's blocks around the disk so the head jumps more. **Defragmentation** packs related blocks closer to speed sequential reads (more relevant to HDDs than SSDs). **Formatting** prepares a disk with a file system structure (and usually erases previous contents) so the OS can store files.

💡 On SSDs, heavy classic defragmentation is usually unnecessary and can add wear; the OS still maintains free-space maps.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

🚀 **Beyond the syllabus (2026 update)** *Not required for the exam – read for interest.* In 2026 school PCs still meet NTFS (Windows), APFS (Mac) and ext4 (Linux). USB sticks may still use exFAT/FAT32 for compatibility. Cloud drives (OneDrive, Google Drive) look like folders but the blocks live in a data centre – the OS still uses the same open/read/write calls.

Key Words

File – Named collection of related bytes.	**File type / extension** – .exe .jpg .txt etc. hint how to open the file.
Directory / folder – Container in the file hierarchy.	**File system** – On-disk organisation of files and free space (e.g. FAT).
FAT – File Allocation Table – a classic simple file system.	**Access privileges** – Read/write/execute rights for users or groups.
Contiguous allocation – File stored in neighbouring blocks.	**Linked allocation** – Blocks chained by pointers.
Indexed allocation – Index block lists all data-block addresses.	**Fragmentation** – File pieces scattered so access slows.
Defragmentation – Rearranging blocks to pack a file closer together.	**Formatting** – Preparing a disk with a file-system structure.

Sample Q&A

1. Why do file types / extensions matter?

> **Answer:** They hint which program should open the file and how the bytes should be interpreted.

2. What is a file system?

> **Answer:** The on-disk method of organising directories, files and free space (e.g. FAT, NTFS, ext4).

3. Name three storage allocation methods.

> **Answer:** Contiguous, linked and indexed.

4. What problem does contiguous allocation often face?

> **Answer:** External fragmentation – free holes that are too small for new files.

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

5. What does defragmentation do?

> **Answer:** Moves scattered blocks of a file closer together to improve sequential access.

6. Why format a disk?

> **Answer:** To create a fresh file-system structure the OS can use (usually erasing old data).

MCQ

1. A directory is

A. a folder that organises files in a hierarchy
B. a type of RAM only
C. a password algorithm
D. a network cable
E. a CPU register

2. FAT stands for

A. File Average Time
B. File Allocation Table
C. Firmware Auto Test
D. Fast Access Tape
E. Folder Access Token

3. Contiguous allocation stores a file in

A. neighbouring disk blocks
B. random non-linked sectors with no map
C. a single bit
D. the CPU cache alone
E. only in RAM registers

4. Linked allocation uses

A. no pointers at all
B. page tables for RAM only
C. pointers from each block to the next
D. DNS names
E. only contiguous runs forever

5. Indexed allocation keeps

A. no list of blocks
B. an index block of pointers to data blocks
C. the file only in registers
D. passwords in the boot sector always
E. nothing on disk

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

6. External fragmentation is typical of

A. pure linked allocation with no holes concern
B. CPU scheduling
C. GUI themes
D. contiguous allocation
E. sound volume

7. Defragmentation mainly helps

A. speeding sequential access by packing blocks
B. increasing screen brightness
C. changing the keyboard language
D. stopping all I/O
E. removing the OS

8. Formatting a disk usually

A. changes the CPU model
B. adds more RAM chips
C. repairs the monitor
D. installs a game only
E. creates a file-system structure (and may erase data)

9. Access privileges control

A. who may read/write/execute a file
B. the clock speed only
C. the fan noise only
D. only the desktop colour
E. DNS servers only

10. Path /home/amma/notes.txt starts at

A. a CPU register
B. the root directory
C. a random empty sector with no name
D. the printer
E. the BIOS only

11. With contiguous allocation A=2,B=3,C=1 from block 0 on a 10-block disk, free blocks =

A. 10
B. 1
C. 6
D. 4
E. 0

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

12. One-level index + 5 data blocks uses how many blocks?

A. 5
B. 10
C. 1
D. 4
E. 6

13. Which term means: “Named collection of related bytes”?

A. Directory / folder
B. Access privileges
C. Linked allocation
D. Defragmentation
E. File

14. Which term means: “.exe .jpg .txt etc. hint how to open the file”?

A. Formatting
B. File type / extension
C. Indexed allocation
D. Contiguous allocation
E. File system

15. Which term means: “Container in the file hierarchy”?

A. Fragmentation
B. FAT
C. File
D. Directory / folder
E. Linked allocation

16. Which term means: “On-disk organisation of files and free space (e.g. FAT)”?

A. Defragmentation
B. Access privileges
C. File system
D. File type / extension
E. Indexed allocation

17. Which term means: “File Allocation Table – a classic simple file system”?

A. Directory / folder
B. Contiguous allocation
C. Formatting
D. FAT
E. Fragmentation

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

18. Which term means: “Read/write/execute rights for users or groups”?

A. File
B. Linked allocation
C. Access privileges
D. Defragmentation
E. File system

19. Which term means: “File stored in neighbouring blocks”?

A. FAT
B. Indexed allocation
C. Contiguous allocation
D. Formatting
E. File type / extension

20. Which term means: “Blocks chained by pointers”?

A. Access privileges
B. Directory / folder
C. Fragmentation
D. File
E. Linked allocation

Structured Questions

1. Explain why file types are needed and name three extensions. *(3 marks)*

> Tell OS/apps how to interpret bytes (1); e.g. .exe .jpg .txt (1); mismatch can open wrongly (1).

2. Describe a directory hierarchy with a short path example. *(3 marks)*

> Tree from root (1); path names each step (1); example /home/user/file (1).

3. Compare contiguous and linked allocation (one advantage each, one drawback each). *(2 marks)*

> Contiguous: fast sequential / fragmentation (2); linked: easy grow / slow random (2).

4. A 10-block disk stores A=2,B=3,C=1 contiguously from 0. Give ranges and free count. *(2 marks)*

> A 0–1, B 2–4, C 5–5 (2); free 4 (1).

5. Explain defragmentation and formatting briefly. *(2 marks)*

> Defrag packs scattered blocks (1); format builds file-system structure / may erase (2).

© 2026 schoolict.net (Ruwan Mangala Suraweera) · All rights reserved

Summary

- Files have types/extensions and attributes; directories form a hierarchy.
- File systems (e.g. FAT) organise data on disk.
- Security uses passwords and access privileges.
- Allocation: contiguous, linked, indexed – each with trade-offs.
- Defragmentation packs blocks; formatting prepares a file system.

MCQ answer key – 5.2

1. **A** – Hierarchy container.
2. **B** – Classic file system.
3. **A** – One run of blocks.
4. **C** – Block chain.
5. **B** – Index + data.
6. **D** – Holes between contiguous files.
7. **A** – Pack file pieces.
8. **E** – Prepare file system.
9. **A** – Permissions.
10. **B** – Root of the tree.
11. **D** – 2+3+1=6 used; 10−6=4.
12. **E** – 1+5=6.
13. **E** – File – Named collection of related bytes.
14. **B** – File type / extension – .exe .jpg .txt etc. hint how to open the file.
15. **D** – Directory / folder – Container in the file hierarchy.
16. **C** – File system – On-disk organisation of files and free space (e.g. FAT).
17. **D** – FAT – File Allocation Table – a classic simple file system.
18. **C** – Access privileges – Read/write/execute rights for users or groups.
19. **C** – Contiguous allocation – File stored in neighbouring blocks.
20. **E** – Linked allocation – Blocks chained by pointers.

© 2026 schoolict.net (Ruwan Mangala Suraweera). All rights reserved.