ANDIJAN STATE PEDAGOGICAL INSTITUTE

Lecture materials · 2026–2027 academic year

SUBJECT

# Information Systems and Digital Technologies

## Lecture 12 · Application Software and Software Development Lifecycle

Part 1 of 2 · 2 hours

Lecturer: **Ismailov Alisher**

E-mail: alisherismailov1991@gmail.com

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

LECTURE 12 · 2 HOURS · PART 1/2

FUNDAMENTALS OF COMPUTERS · CHAPTER 10

# Computer Software

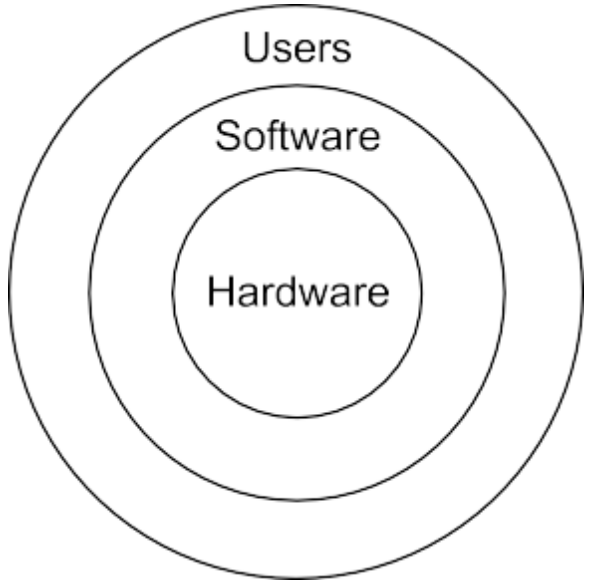
The programs that give the hardware life — and how we plan them

**System management**

**System development**

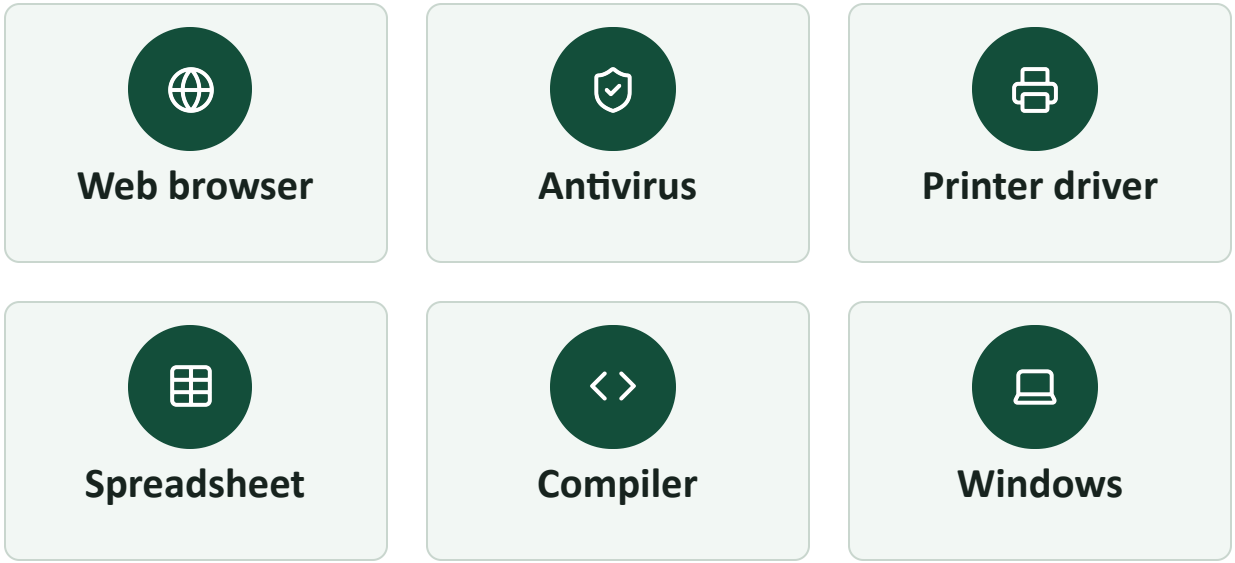
**Standard applications**

**Unique applications**




*Software sits between users and hardware — textbook Fig. 10.1*

# What can a computer do with no software?



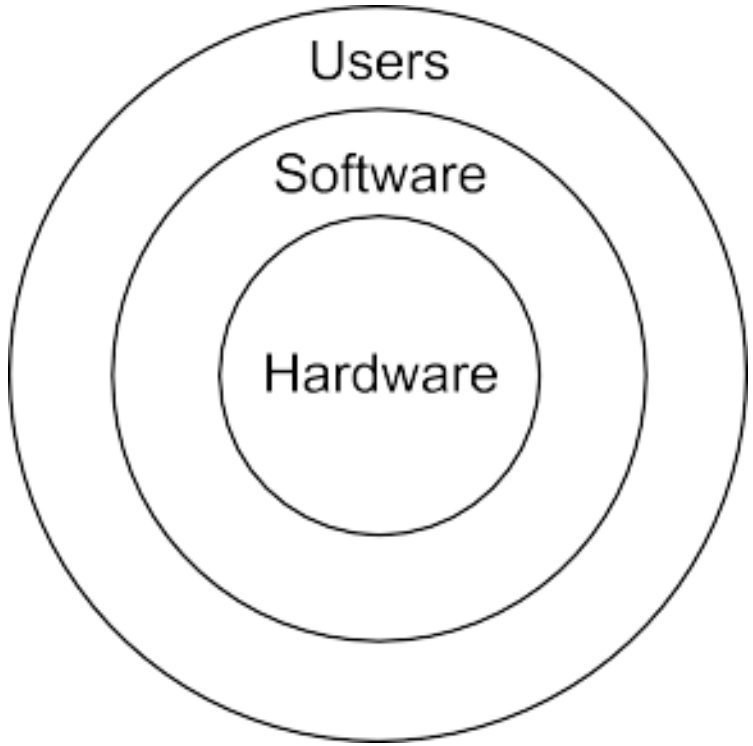

**Think – Pair – Share (2 min)**

1. Which of these did you use today?
2. Which ones work "behind the scenes" and are never opened by you?
3. Split them into two groups. Explain your rule.

# By the end of this lesson, you can…

1. Classify software as system management, system development, standard application or unique application.
2. Explain what utility programs, device drivers, translators, linkers, debuggers and editors do.
3. Describe word processors, spreadsheets, DBMS and desktop publishing.
6. Classify application software by purpose and platform (desktop, web, mobile), with real use cases.

# Hardware is the heart, software is the soul




*Textbook Fig. 10.1*

### Hardware

The physical equipment: reads and processes data, stores results, gives output.

### Software

A set of computer programs that make the hardware work.

### Program

Logical instructions, written in a programming language, telling the computer how to do a task.

**Software is the interface between the hardware and the user.**

# Two jobs, two kinds of software

## System software

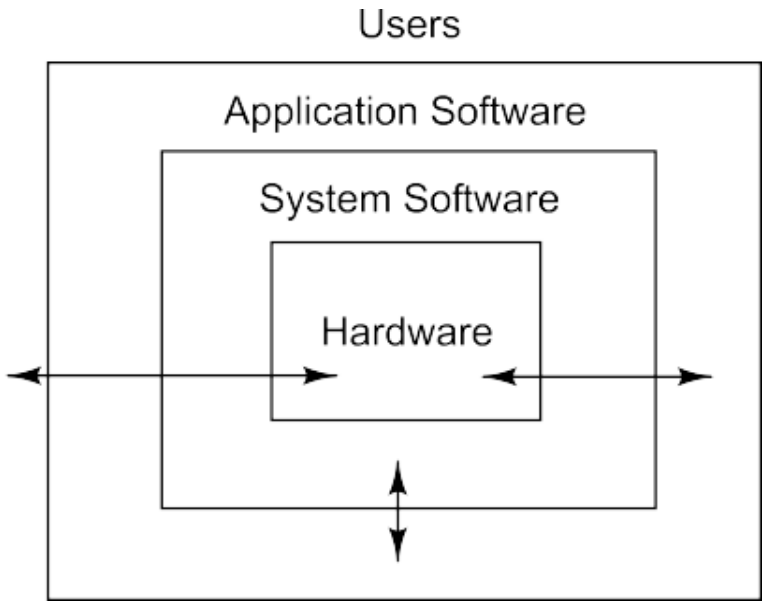
Controls and coordinates the hardware and manages its performance.

**Essential — the computer cannot work without it.**

## Application software

Helps users do their own tasks: documents, databases, graphics.

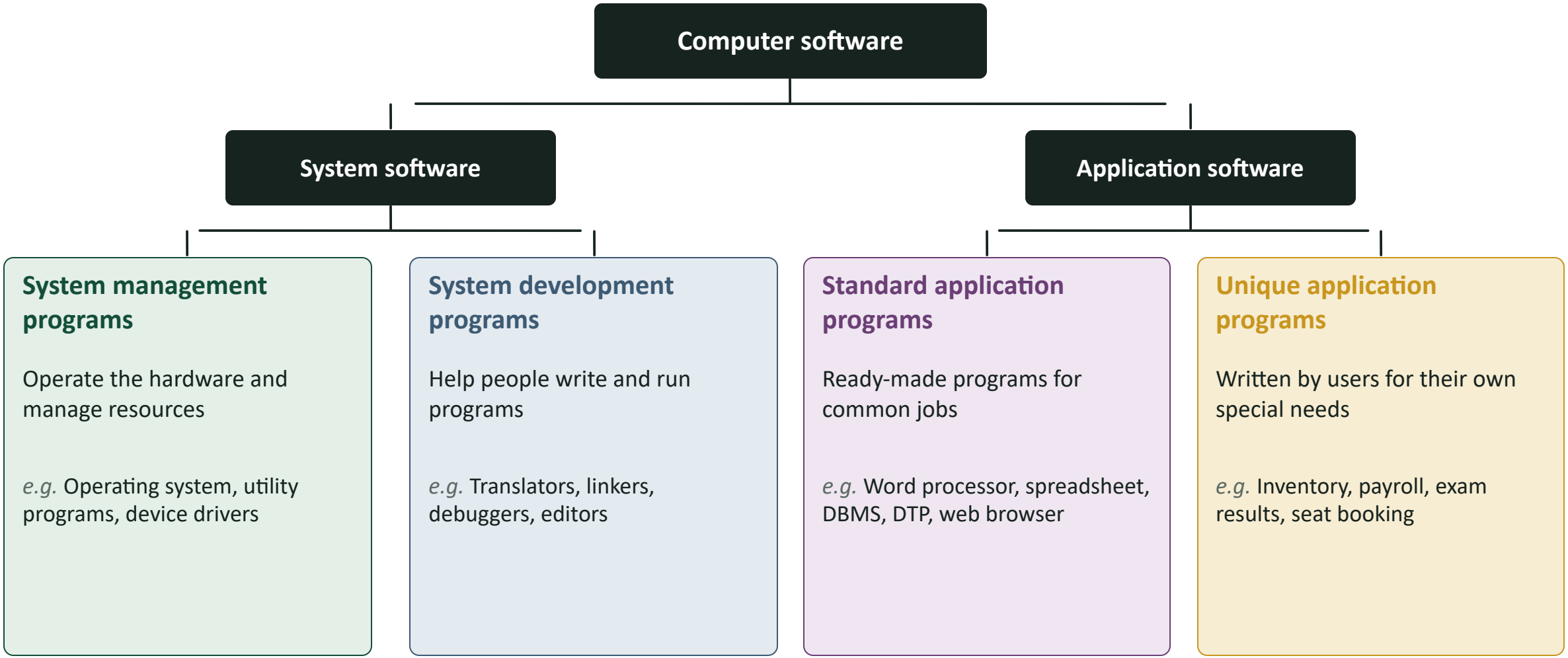
**Additional — added for a specific job.**




*Layers of software — textbook Fig. 10.2*

*System software also lets the hardware work with applications and users.*

# Four families of software



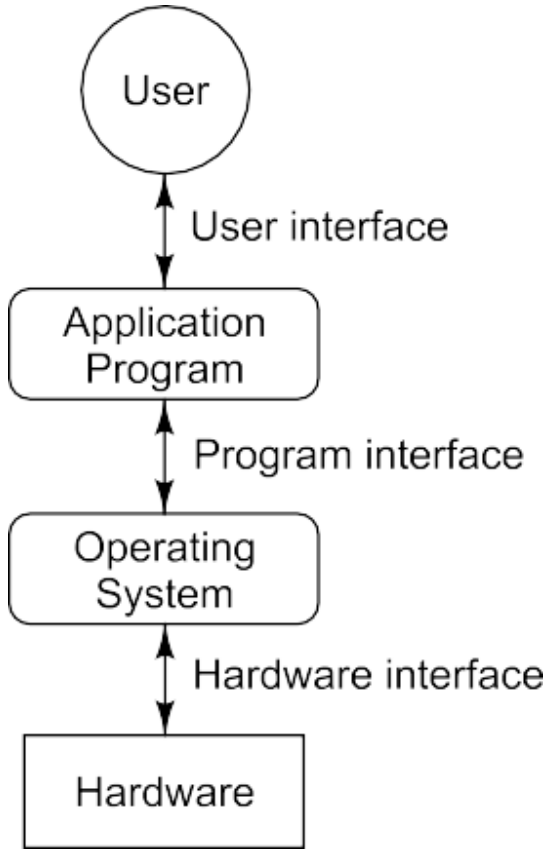
CHECK FOR UNDERSTANDING

# Which family?

| | | | |
|---|---|---|---|
| MS Word | Linker | Operating system | Payroll program for one company |
| Printer driver | MS Excel | Virus scanner | Compiler |

| System management | System development | Standard application | Unique application |
|---|---|---|---|
| | | | |

# The operating system: the principal system software




*Textbook Fig. 10.5*

**MAIN FUNCTIONS**

- Schedules and runs all processes
- Allocates main memory and storage
- Assigns hardware devices to programs
- Creates, stores and handles files
- Keeps the order in which programs run
- Interprets commands and instructions
- Gives a friendly user interface
- Keeps access to resources secure

Supplied by the hardware maker — rarely written by users, as it is so complex.

# Utility programs: handy extras for the OS

### Search & replace

find files by criteria

### Print

start printing

### Disk defragmenter

store data in one place, not scattered

### System profiler

list installed hardware and software

### Encryption

scramble files sent over a network

### Virus scanner

detect viruses and bugs

### Backup

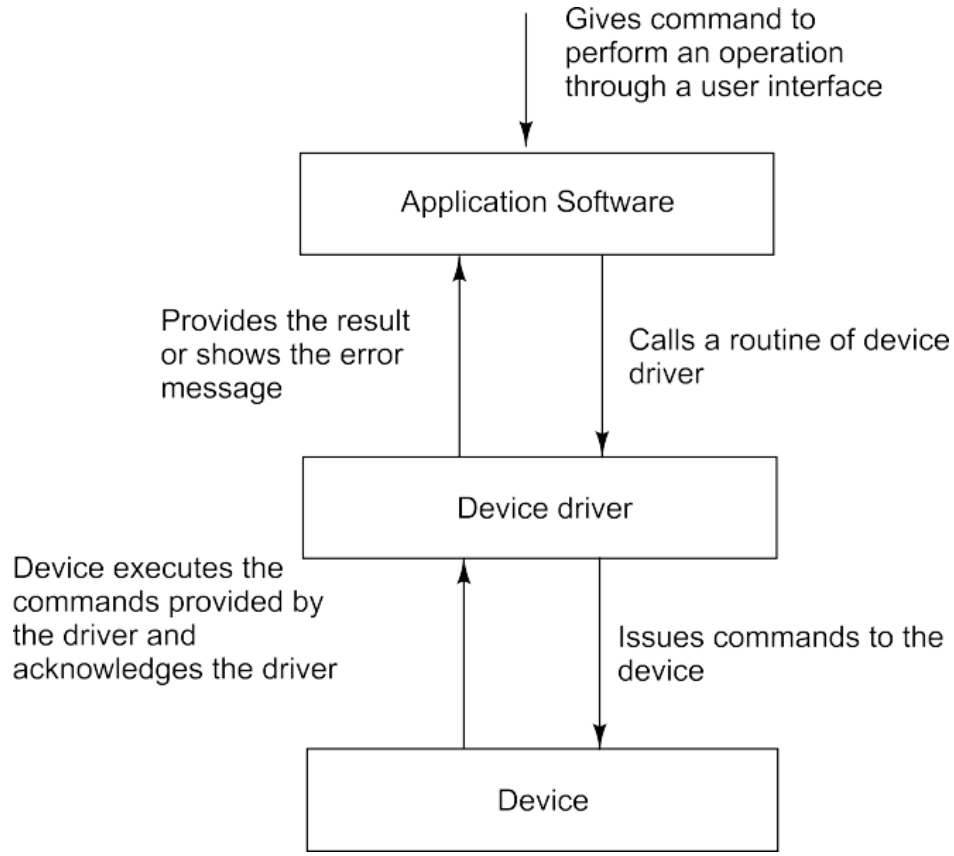
copy files to disk or tape

### Data recovery

get back lost data

Not essential: the OS can run most programs without them — they just add features.

# Device drivers: translators for hardware




*Working of a device driver — textbook Fig. 10.6*

1. A program needs a device and sends a command to the OS.
2. The OS calls the device driver's routine.
3. The driver sends control and data signals to the device through buses and cables.
4. The device acknowledges, does the task and reports back.
5. The driver returns the result to the program.

*e.g. keyboard, monitor, printer drivers*

CHECK FOR UNDERSTANDING

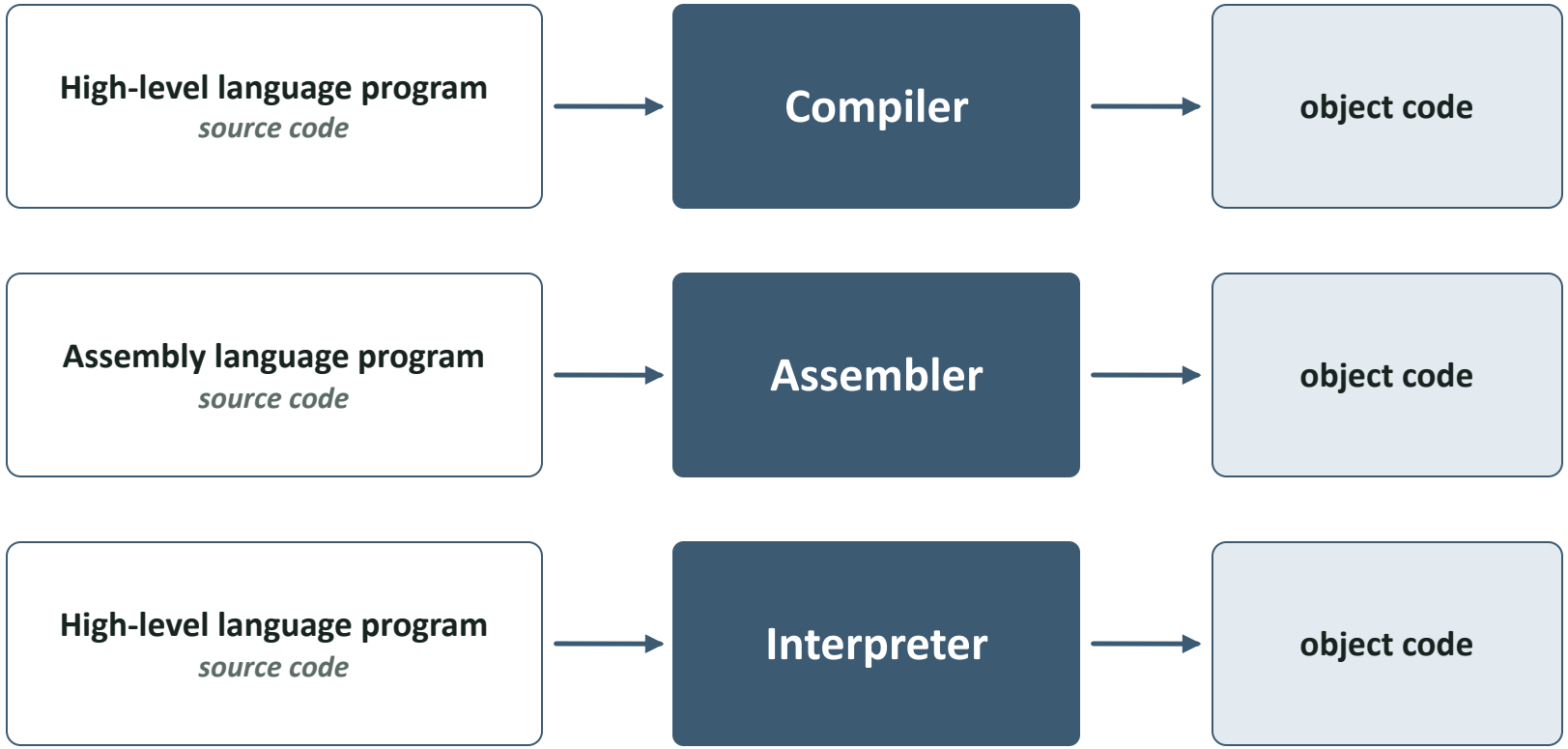
# Which program helps?

- Your USB stick was damaged and you need your photos back.
- Files are scattered in pieces across the disk, so it runs slowly.
- You plug in a new printer and Windows cannot talk to it yet.
- You must send an exam paper over the Internet so no one can read it.
- You want a list of all hardware and software on your computer.

## Choose from

- **Data recovery**
- **Disk defragmenter**
- **Device driver**
- **Encryption**
- **System profiler**

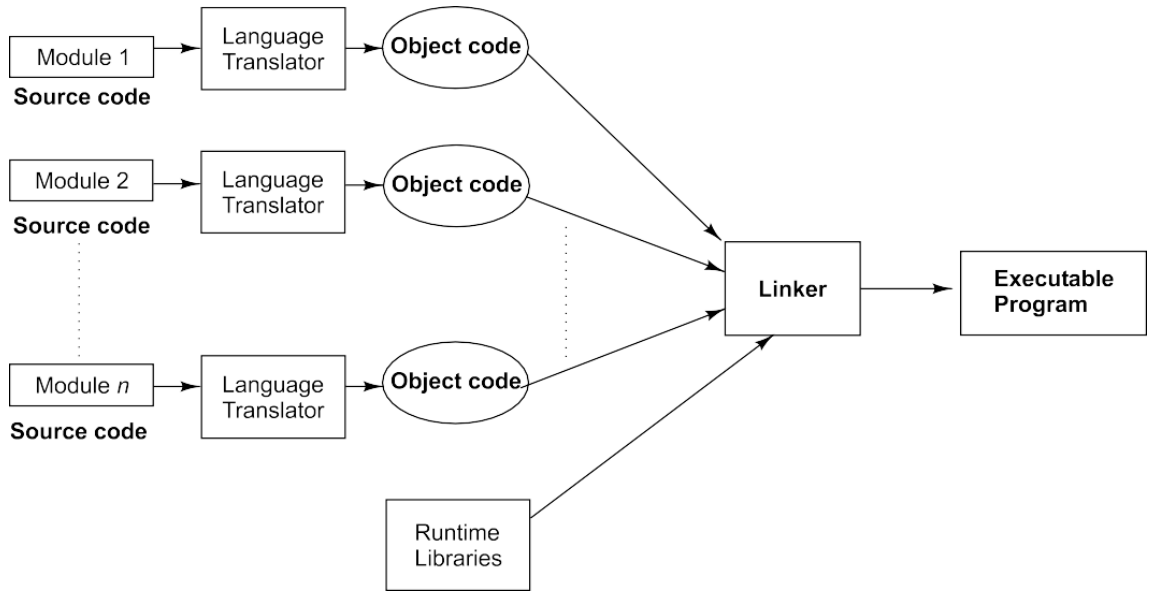
# Language translators: turning code into machine code




*After textbook Fig. 10.7*

**Source code = what the programmer writes · Object code = the low-level result**

# Linkers join the pieces; debuggers find the bugs




*Working of a linker — textbook Fig. 10.8*

## Linker

Combines the object code of all modules, plus runtime libraries, into one executable program.

## Debugger

Finds errors (bugs) and stops the program where an error occurs.

**Machine-level:** checks the object code

**Symbolic:** shows the bug in the original high-level code

- Step-by-step execution
- Backtracking to earlier steps
- Breakpoints and tracking variables

# Editors: six kinds for six kinds of content

### Text editor

plain text; used to update configuration files

### Digital audio editor

music and sound signals

### Graphics editor

graphical and animation objects

### Binary file editor

data made of 0s and 1s

### HTML editor

web pages

### Source code editor

programs in C, C++, Java

*All editors let you copy, cut and paste while editing.*

CHECK FOR UNDERSTANDING

# Which development tool?

**A** Translates a high-level program one line at a time, running each line straight away.

**B** Combines the object code of five modules into one executable program.

**C** Converts an assembly language program into low-level code.

**D** Stops the program at the line where an error happens.

**E** Translates the whole high-level program in one go.

**F** Is used to write the source code of a Java program.

# Standard applications: ready-made for common jobs

### Word processor

letters, reports

### Spreadsheet

numbers and formulas

### Database manager

records of products, staff

### Desktop publisher

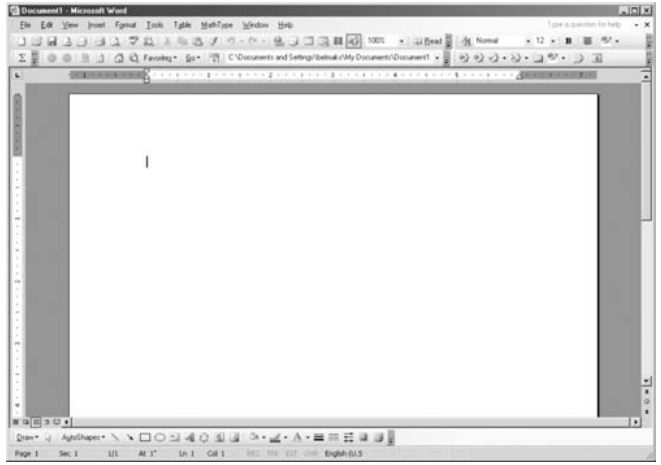
page layouts for print

### Web browser

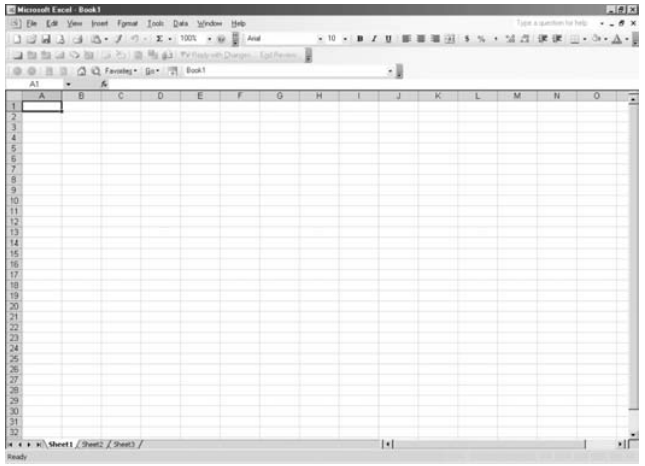
viewing web pages

**Also called general-purpose programs.** Bought ready-to-use from hardware or software vendors. They raise productivity and save time and money compared with writing your own.

# Word processors and spreadsheets



*MS Word — textbook Fig. 10.9*



*MS Excel — textbook Fig. 10.10*

## Word processor

- Type and format text: bold, italic, colour, style
- Cut-paste to move text; copy-paste to copy it
- Example: MS Word

## Spreadsheet

- Worksheet of cells with numbers or text
- Formulas calculate values
- Change a cell → linked results update

## Try it: automatic update

| | A | B | C |
|---|---|---|---|
| 1 | 5 | 3 | =A1+B1 |

C1 shows **8**

**Now change A1 to 7. What does C1 show?**

*Why do people keep money records in spreadsheets?*

# Database Management System (DBMS)

**Database:** records stored in a structured way. **Record:** values about one entity, e.g. a student.

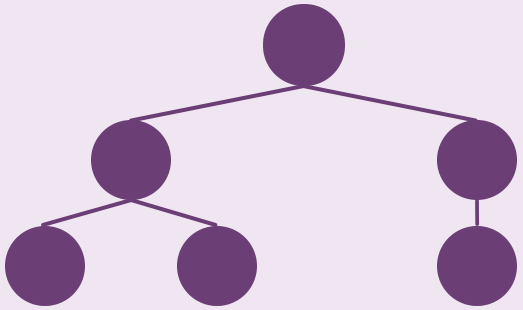
A DBMS lets you store, search, update and delete records easily.

## Relational model

| ID | Name | Class |
|---|---|---|
| | | |
| | | |

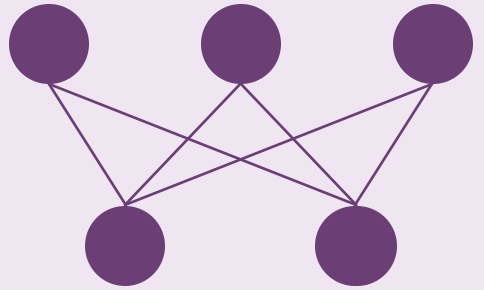
records in tables of rows and columns

## Hierarchical model



a tree: each record has one parent, one or more children

## Network model



each node can have several parents and children

**Advantages:** organises information · keeps data consistent · protects data integrity · password security · access over a network

# Inside a DBMS — and desktop publishing

## DATA STRUCTURES

| ID | Name | Class |
|---|---|---|
| 101 | Aziza | 9A |
| 102 | Bekzod | 9B |

**File** = the whole table (many records)

**Record** = one row (highlighted)

**Field** = one column of same-type values

## TRANSACTIONS

**A**tomicity · **C**onsistency · **I**solation · **D**urability

*Plus a query language to search and update data.*



## Desktop publishing (DTP)

- Lays out pages: margins, justification, text, graphics, images
- WYSIWYG — "What You See Is What You Get": you see the page as it will look
- Used in publishing for print and display media

**Examples:** Quark XPress, InDesign, Microsoft Publisher, Apple Pages

# Unique application programs: made for one need

Written by organisations or professionals — scientists, engineers, accountants, teachers — usually in C, C++ or Java.



## Inventory management

- Keeps records of items in warehouses and storerooms, using a DBMS
- Each item has a unique ID — the primary key
- The ID is printed as a barcode and read by a barcode reader: no typing
- Reports on stock, availability, shipping dates



## Payroll system

- Used by the HR department to calculate salaries
- Enter attendance, tax deductions and so on
- Removes manual calculation
- Custom-built or off-the-shelf; can print or e-mail salary slips

CHECK FOR UNDERSTANDING

# Which application?

- A shop owner wants totals to update by themselves when a price changes.
- The school designs its magazine pages, seeing each page exactly as it will print.
- A warehouse scans each box’s barcode to update its stock records.
- A library stores thousands of book records and searches them by author.
- HR enters attendance and tax, and the program works out each salary.

## Bonus

For each one, is it a STANDARD or a UNIQUE application?

# Types of application software and their uses

### Productivity

Word, Excel, Google Docs — reports, budgets, class notes.

### Communication

Telegram, Gmail, Zoom — messages, e-mail, online lessons.

### Graphics and multimedia

Photoshop, Canva, VLC — posters, photo and video editing.

### Education

Moodle, Duolingo, Khan Academy — courses, tests, languages.

### Business and enterprise

1C, SAP, CRM systems — accounting, stock, customers.

### Entertainment

Games, Spotify, YouTube — play, music and video.

# Where apps run: desktop, web and mobile

| | Desktop app | Web app | Mobile app |
|---|---|---|---|
| **How you get it** | Install on a PC or laptop | Open in a browser — no install | Download from an app store |
| **Internet needed?** | Usually not | Yes | Sometimes |
| **Updates** | You install them | Automatic, on the server | Through the app store |
| **Strength** | Powerful; uses full hardware | Works on any device | Always with you; camera, GPS |
| **Examples** | Photoshop, AutoCAD, MS Word | Google Docs, Gmail, Canva | Telegram, Click, Yandex Go |

# Review: can you answer these?

01 What is the difference between system and application software?

02 Name the four families of software.

03 Name three utility programs and what they do.

04 Why does a computer need device drivers?

05 How is a compiler different from an interpreter?

06 What does a linker do?

07 What do file, record and field mean in a DBMS?

08 Give one example each of a desktop, a web and a mobile app.

# Exit ticket

Answer on a slip of paper before you leave.

1. Sort into system or application software: MS Excel, device driver, compiler.

2. Which program joins object code modules into one executable program?

3. Name one standard and one unique application program.