UNIVERSITAS
GADJAH MADA

LOCALLY ROOTED,

GLOBALLY RESPECTED

Meeting 3

Demo

Fundamentals of Programming
TKU211131

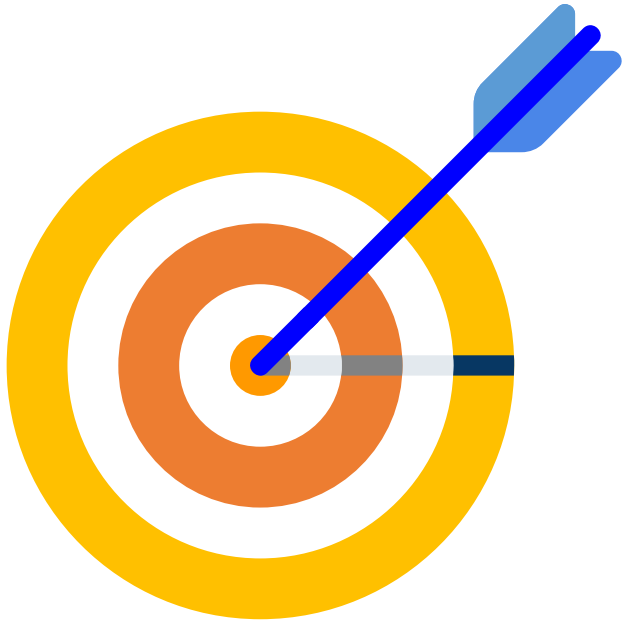
DTETI FT UGM

Bimo Sunarfri Hantono
bhe@ugm.ac.id

Topics

UNIVERSITAS GADJAH MADA

- Command Line Interface (CLI)
- Demo session: VS Code (editor) + CLI.
  - Implement Computational Thinking using:
    - Elements of a C++ program
    - Data types and operators that C++ can process
    - Basic operations in fundamental programming:
    - Performing calculations
    - Inputting data
    - Displaying results



UNIVERSITAS
GADJAH MADA

Command Line Interface

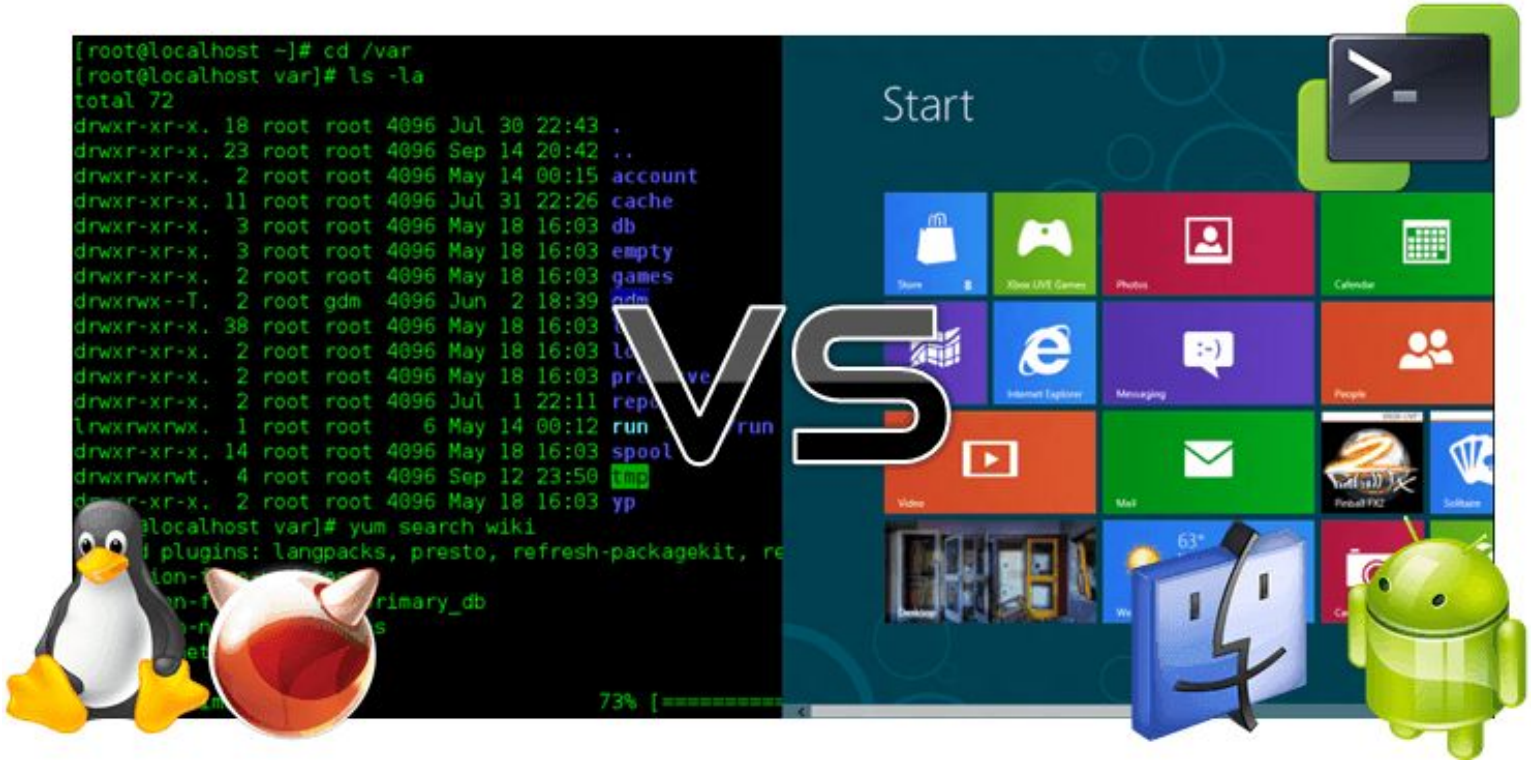
ugm.ac.id

LOCALLY ROOTED, GLOBALLY RESPECTED

What is CLI?

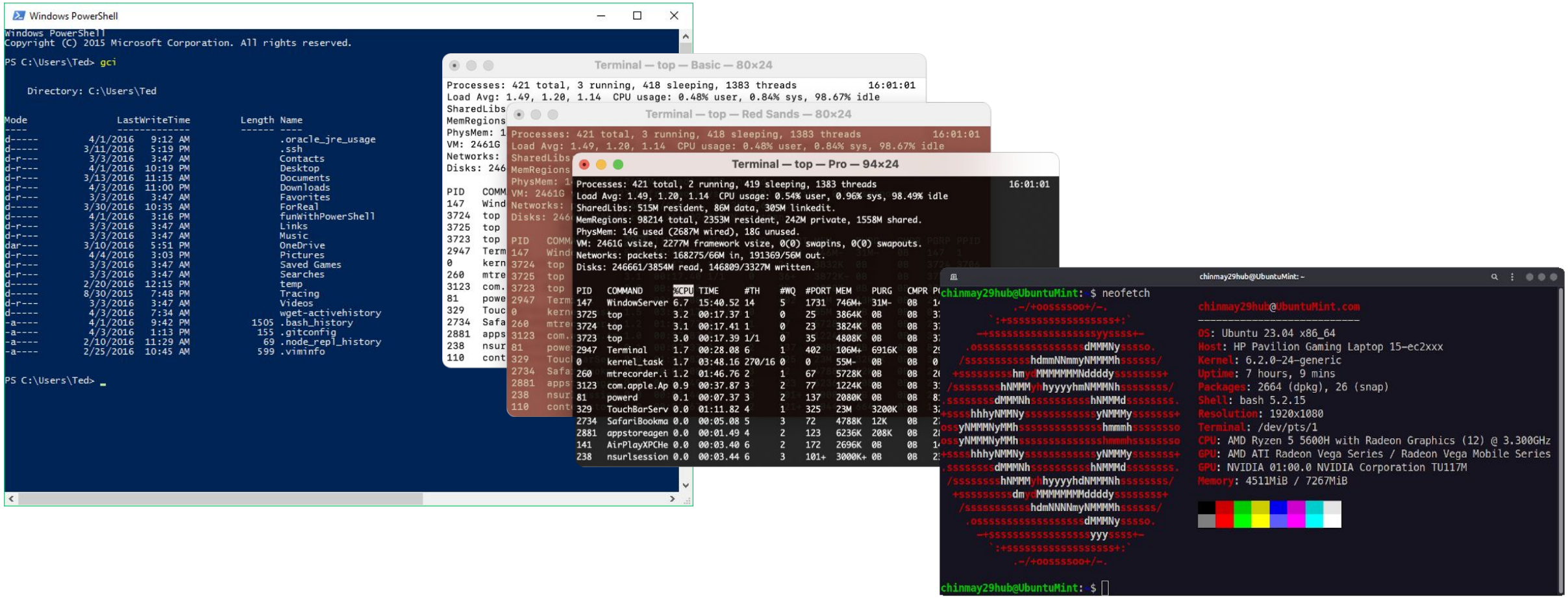
UNIVERSITAS GADJAH MADA

A text-based way to interact with the computer



Example of CLI in Different OS

UNIVERSITAS GADJAH MADA



CLI vs GUI

UNIVERSITAS GADJAH MADA

Aspect	CLI (Command Line Interface)	GUI (Graphical User Interface)
Interaction	Text commands typed by user	Icons, menus, and buttons
Ease of Use	Steeper learning curve (need to memorize commands)	Intuitive, easy for beginners
Speed	Very fast for experienced users	Slower due to multiple clicks
Resource Usage	Low (lightweight)	High (needs more CPU & memory)
Automation	Easy to script & automate	Limited automation
Flexibility	High (fine-grained control)	Moderate (depends on GUI features)
Examples	Terminal, PowerShell, Bash	Windows Explorer, macOS Finder

Example of CLI Command

UNIVERSITAS GADJAH MADA

- cd → change directory
- ls / dir → list files
- g++ file.cpp -o file.exe → compile C++ program
- ./file.exe → run program

CLI Demo

UNIVERSITAS
GADJAH MADA

VS Code Tools

UNIVERSITAS GADJAH MADA

Main tools in **Visual Studio Code**:

- Explorer → see project files
- Editor → write code
- Terminal → compile & run code
- Extensions → add support for C++

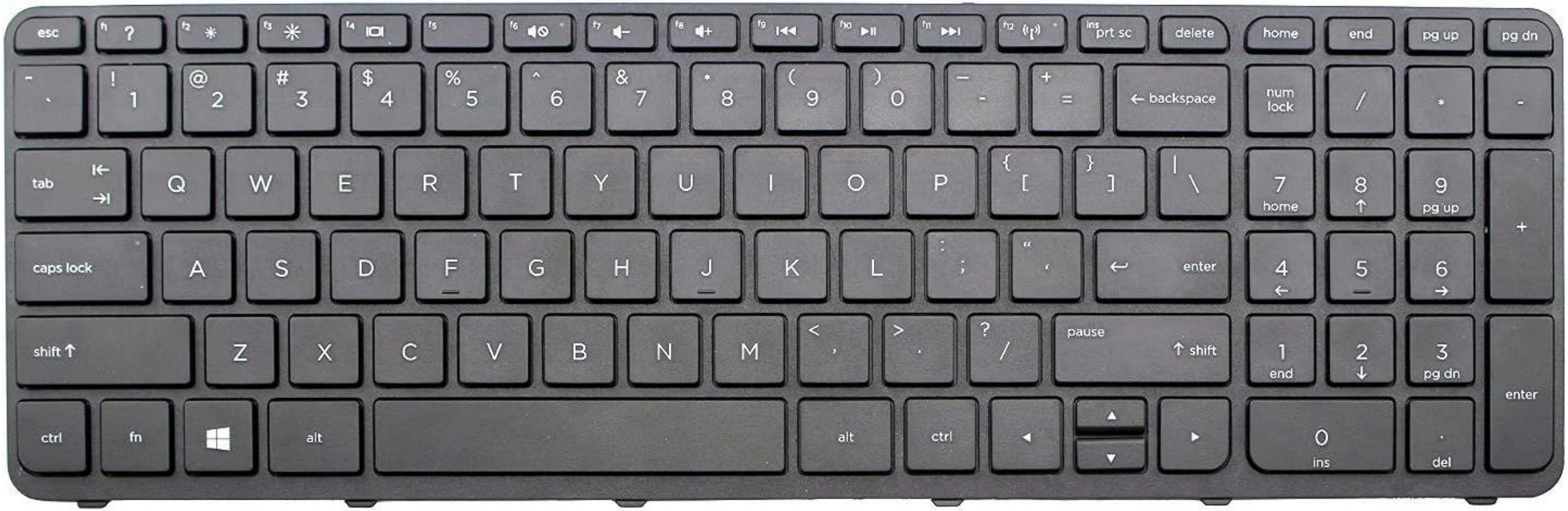
Useful VS Code Shortcuts

UNIVERSITAS GADJAH MADA

- Ctrl + N → new file
- Ctrl + S → save file
- Ctrl + / → comment/uncomment line
- Ctrl + ` → open terminal
- Ctrl + Shift + B → build program
- F5 → run & debug program

Keyboard

UNIVERSITAS GADJAH MADA



VS Code Demo

UNIVERSITAS
GADJAH MADA

UNIVERSITAS
GADJAH MADA

Demo Session

ugm.ac.id

LOCALLY ROOTED, GLOBALLY RESPECTED

Case: Car Breakdown with Low Fuel

UNIVERSITAS GADJAH MADA

Problem Definition

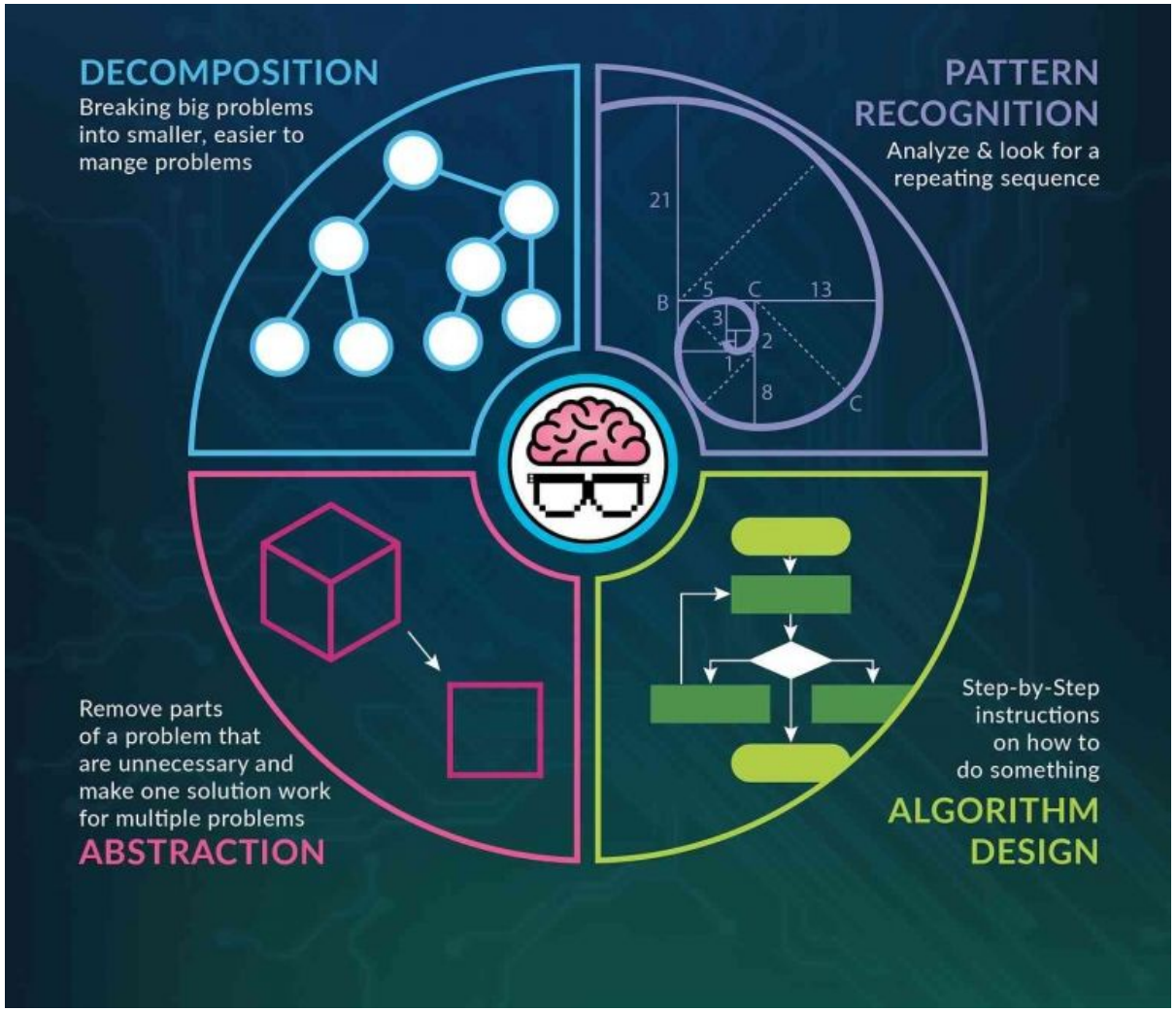
- **Problem:** Car stops in the middle of the road; fuel is almost empty.
- **Goal:** Get the car running again or select a safe alternative.
- **Constraints:** Time, roadside safety, limited tools.

Computational Thinking Steps

UNIVERSITAS GADJAH MADA

CT pillars:

1. **Decomposition** – break into checks & actions
2. **Pattern Recognition** – common car-stop causes
3. **Abstraction** – focus on fuel/battery essentials
4. **Algorithm Design** – clear, testable steps

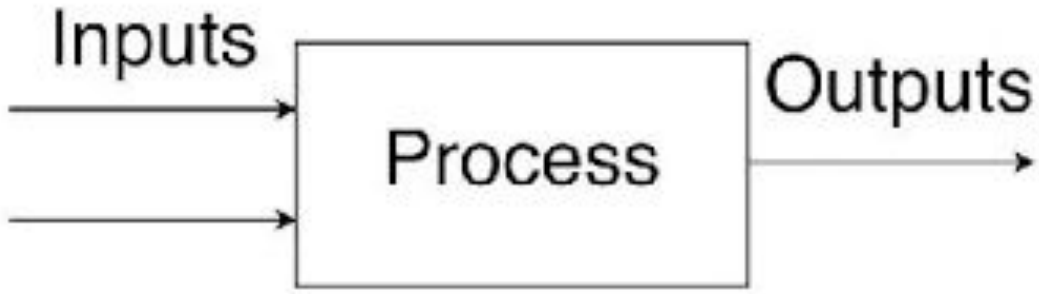


IPO Diagram (Input – Process – Output)

UNIVERSITAS GADJAH MADA

Input

- fuel_level (0 = empty, 1 = low, 2 = ok)
- battery_status (0 = weak, 1 = ok)
- has_spare_battery (true/false)



Process

1. Ensure safety (hazard lights)
2. If fuel > 0 and (battery weak or spare available) → replace battery
3. Try starting engine; if fail, recheck or call assistance
4. If fuel == empty → refuel/assistance

Output

- Decision & action: **drive to gas station** OR **call assistance**

Algorithm Design: Pseudocode

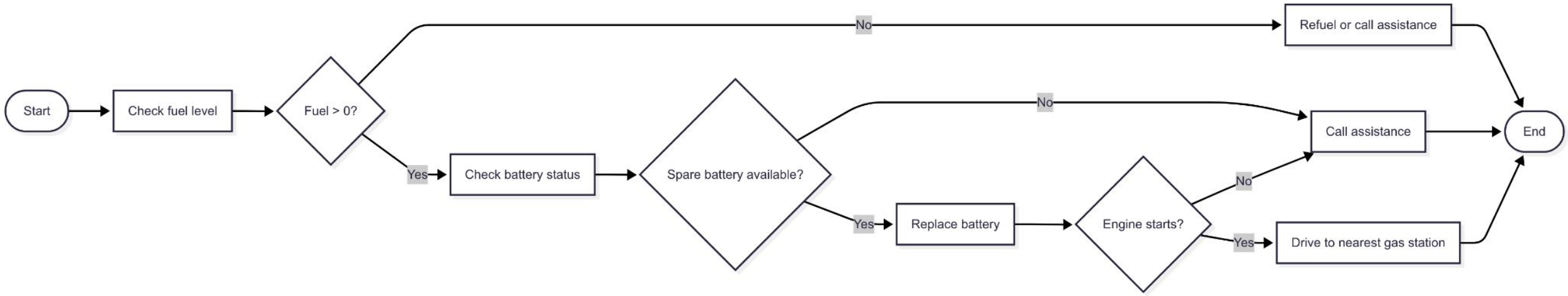
UNIVERSITAS GADJAH MADA

Definition: A simplified, plain-language description of the steps in an algorithm, not tied to any programming language.

```
IF fuel > 0 AND spare battery available
    Replace battery
    IF engine starts
        Drive to gas station
    ELSE
        Call assistance
ELSE
    Refuel or call assistance
```

Algorithm Design: Flowchart

UNIVERSITAS GADJAH MADA



C++ Program (1)

UNIVERSITAS GADJAH MADA

Elements of a C++ Program

- Preprocessor: #include <iostream>
- Namespace: using namespace std;
- Entry point: int main() { … }

```cpp
#include <iostream>
using namespace std;

int main() {
    // program starts here
    return 0;
}
```

C++ Program (2)

UNIVERSITAS GADJAH MADA

Data Types and Operators that C++ Can Process

- Data types used: `int`, `bool`
- Operators shown: assignment `=`, comparison `==`, logical `||` (OR)

```cpp
int fuel_level;       // 0=empty, 1=low, 2=ok
int battery_status;   // 0=weak, 1=ok
bool has_spare_battery; // true/false
```

C++ Program (3)

UNIVERSITAS GADJAH MADA

Basic Operations in Fundamental Programming

- Read → Process → Decide
- Illustrate control flow with `if / else`

```
// basic decision structure
if (fuel_level == 0) {
    // handle empty fuel
} else {
    // handle non-empty fuel
    if (battery_status == 0 || has_spare_battery) {
        // handle weak battery or replace with spare
    }
    // attempt to start engine...
}
```

C++ Program (4)

UNIVERSITAS GADJAH MADA

Performing Calculations

- Compute/derive a state for decision
- Example: simple start condition

```
// naive demo calculation: assume engine can start if fuel is at least low
bool engine_starts = (fuel_level >= 1);  // derived state for decision
```

C++ Program (5)

UNIVERSITAS GADJAH MADA

Inputting Data

- Use `cin` to read from keyboard
- Prompt user clearly

```
cout << "Fuel level (0=empty, 1=low, 2=ok): ";
cin  >> fuel_level;

cout << "Battery status (0=weak, 1=ok): ";
cin  >> battery_status;

cout << "Has spare battery? (0=no, 1=yes): ";
cin  >> has_spare_battery;
```

C++ Program (6)

UNIVERSITAS GADJAH MADA

Displaying Results

- Use `cout` to display actions/outcomes
- Print clear guidance

```
cout << "\n-- Actions --\n";
cout << "Turn on hazard lights and place safety triangle.\n";

if (fuel_level == 0) {
    cout << "Fuel empty. Refuel or call assistance.\n";
} else {
    if (battery_status == 0 || has_spare_battery) {
        cout << "Replace or secure battery connections.\n";
    }
    cout << "Attempting to start engine...\n";
    bool engine_starts = (fuel_level >= 1); // from previous slide
    if (engine_starts) {
        cout << "Engine started. Drive carefully to the nearest gas station.\n";
    } else {
        cout << "Engine failed to start. Call assistance.\n";
    }
}
```

Compile & Execute

UNIVERSITAS GADJAH MADA

- Windows
  - g++ demo_car.cpp -o demo_car.exe
  - demo_car.exe

- MacOS / Linux
  - g++ demo_car.cpp -o demo_car
  - ./demo_car

Quick Debug Tips

UNIVERSITAS GADJAH MADA

- Missing `;` → check compile errors
- Wrong types → verify variable declarations
- Input confusion → echo inputs back to user
- Logic branches → print checkpoints (e.g., *entered fuel==0 branch*)

UNIVERSITAS
GADJAH MADA

Any Question?

UNIVERSITAS
GADJAH MADA

Thank You!
See you, next week, stay safe!