Computer System

What is computer system?

- Digital electronics device
- Able to programmed to accept inputs
- Process the data as per instructions
- Provide the output
- Hardware + Software

Computer

Microcomputer

Minicomputer

Personal computer

Supercomputer

Laptop

Tablet

Desktop

Laptop

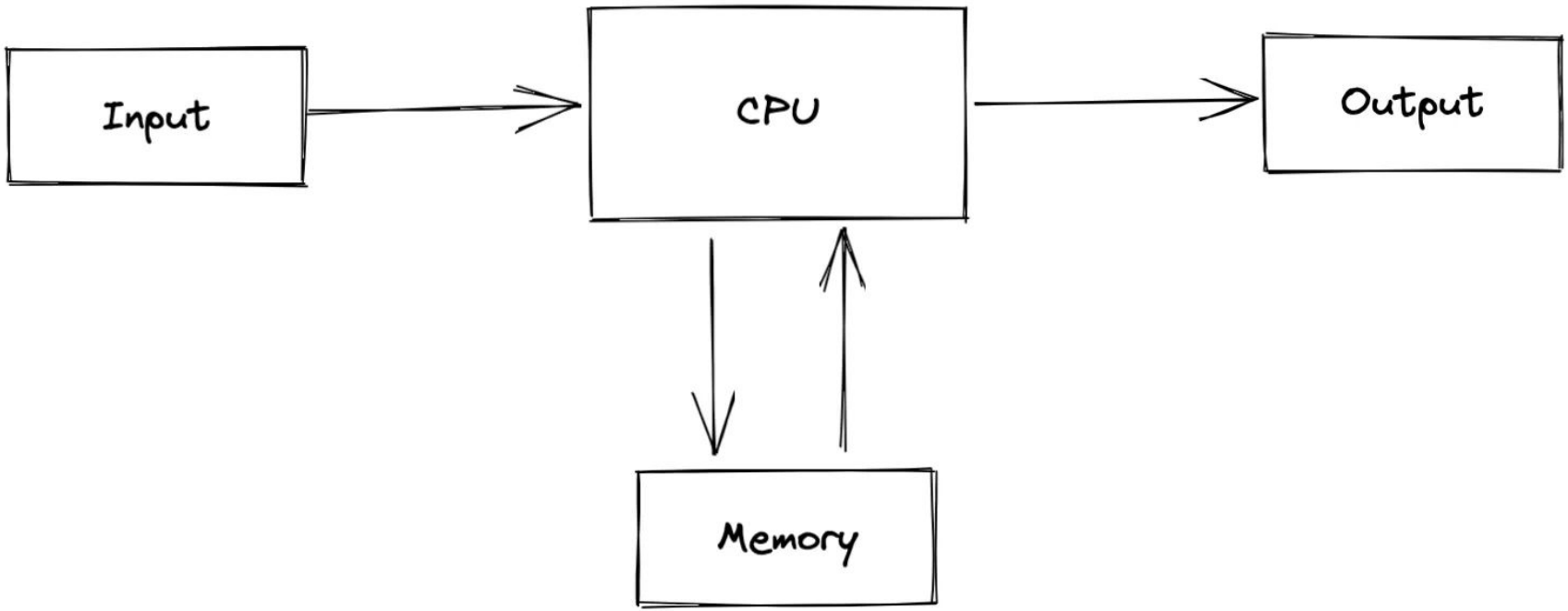
Personal Digital Assistant

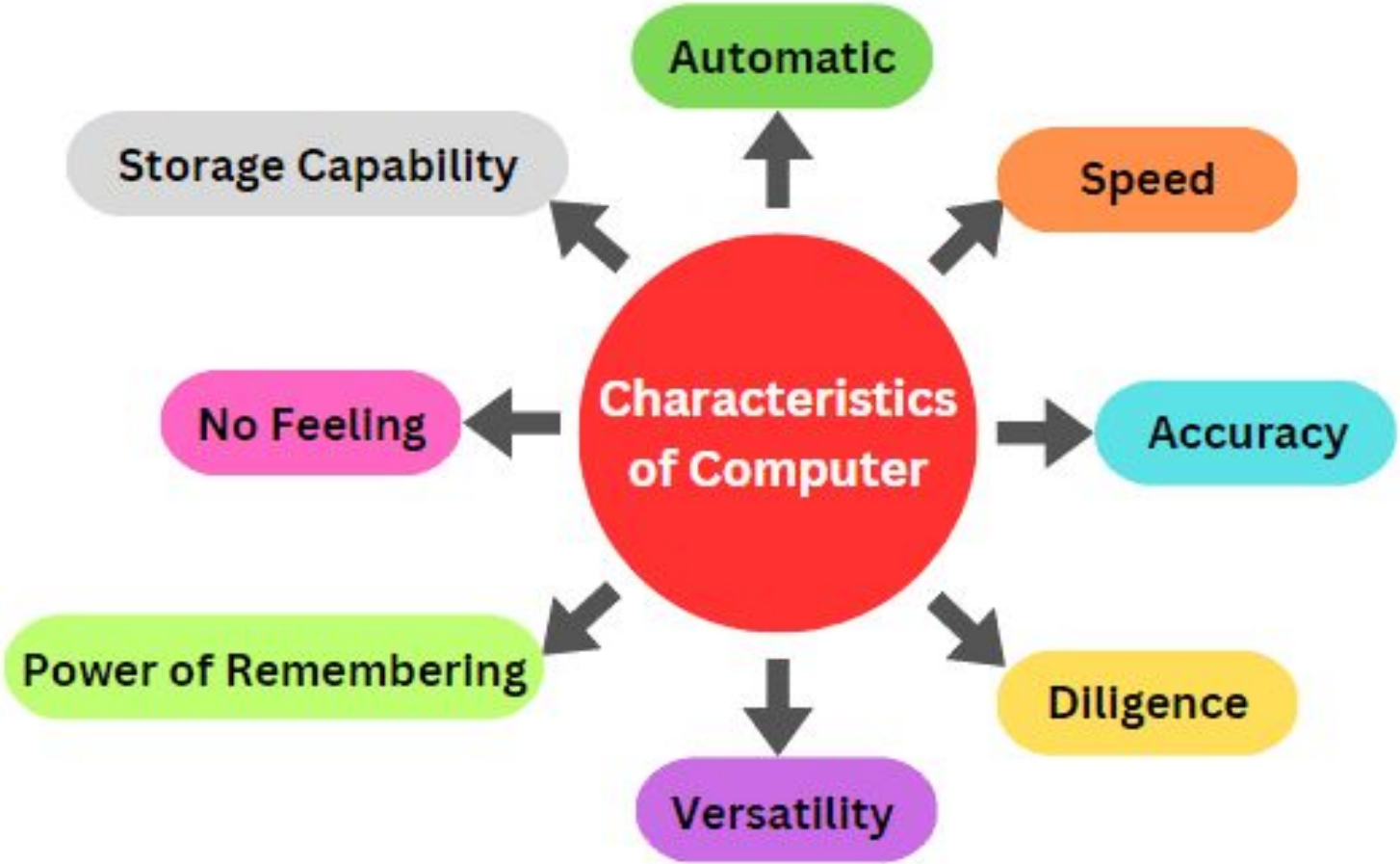
SmartPhone

Netbook

Mainframe

Embedded


Input
CPU
Output
Memory


Automatic
Storage Capability
Speed
Characteristics of Computer
No Feeling
Accuracy
Power of Remembering
Diligence
Versatility

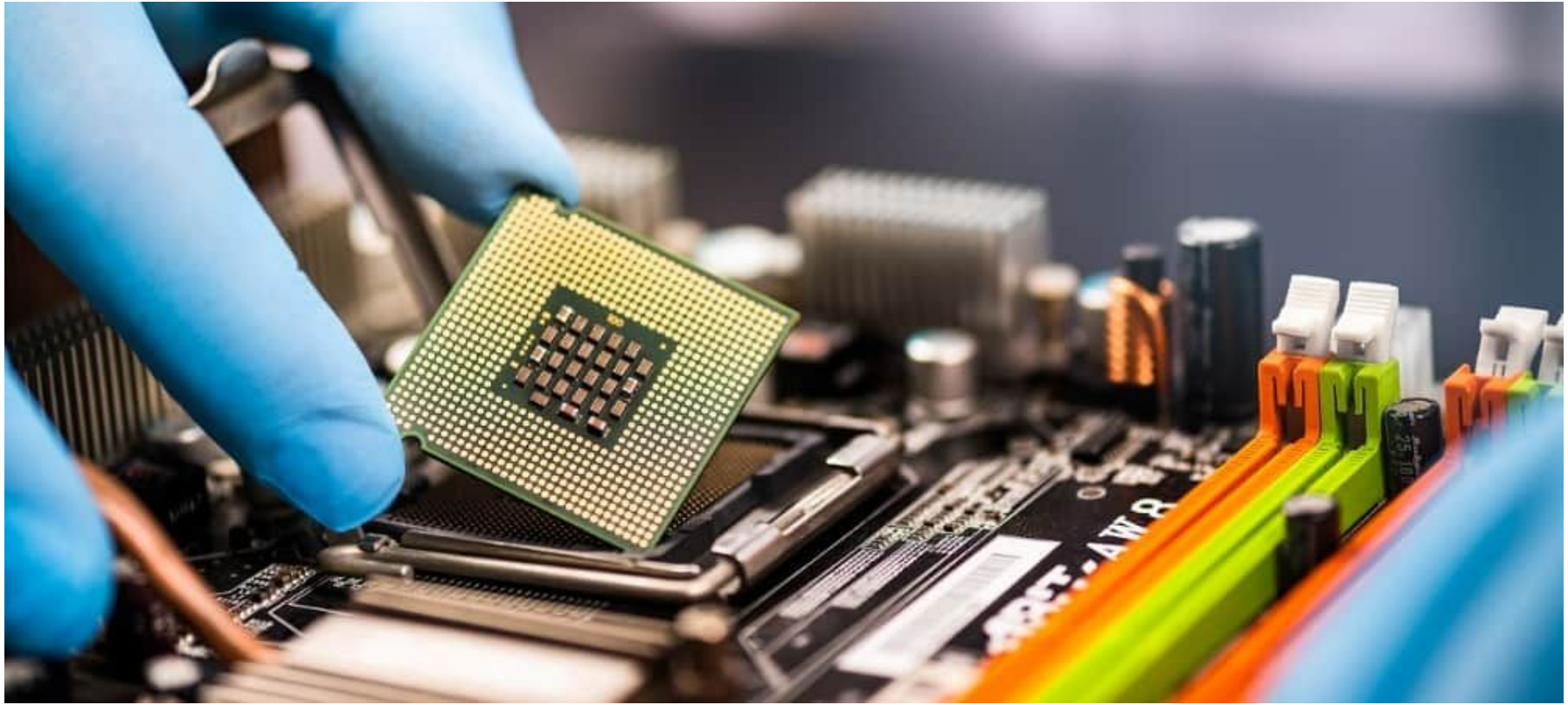
Hardware

Type of hardware

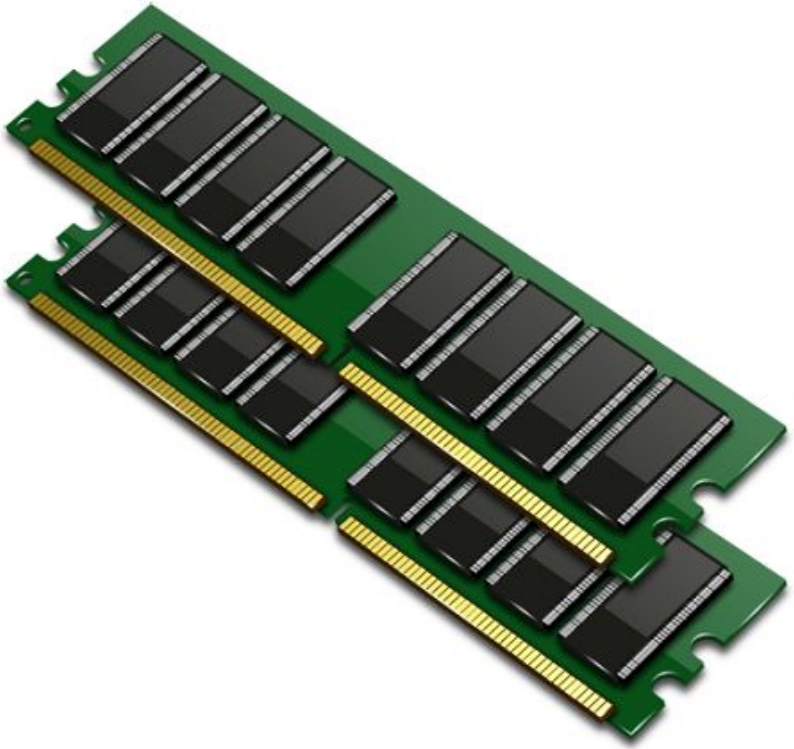
- Internal components

- External components

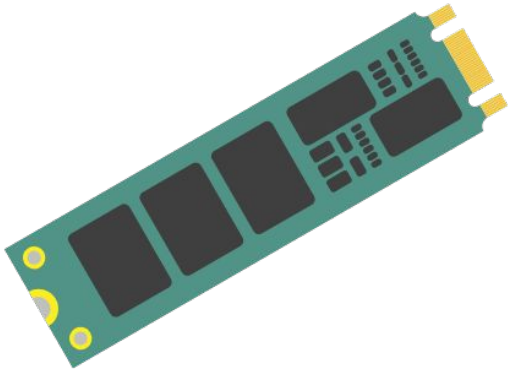
CPU (Central Processing Unit)



Main Memory



Secondary Memory



Input Devices

Output Devices

Monitor

Printer

Speakers

Head Set

Projector

Plotter

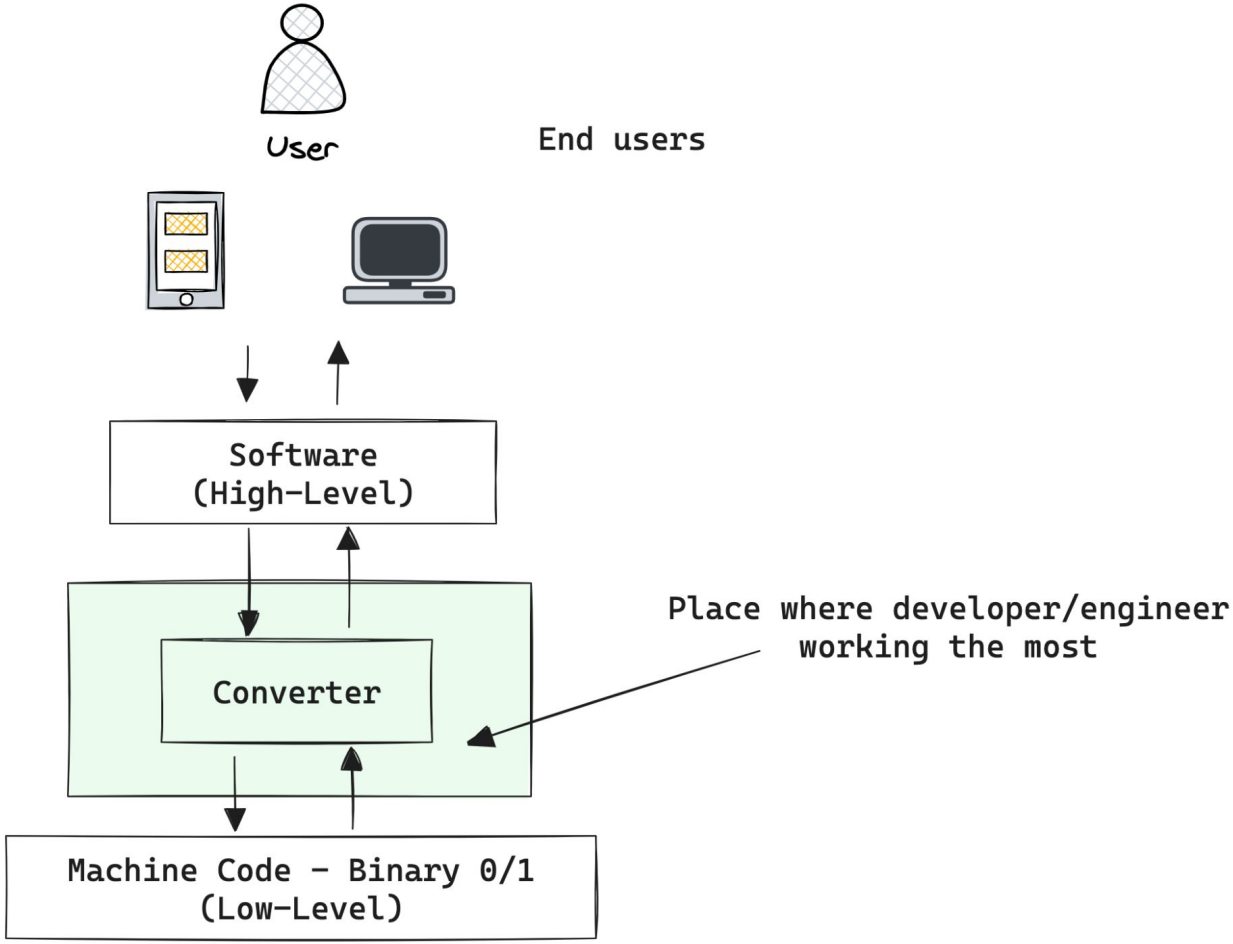
Software

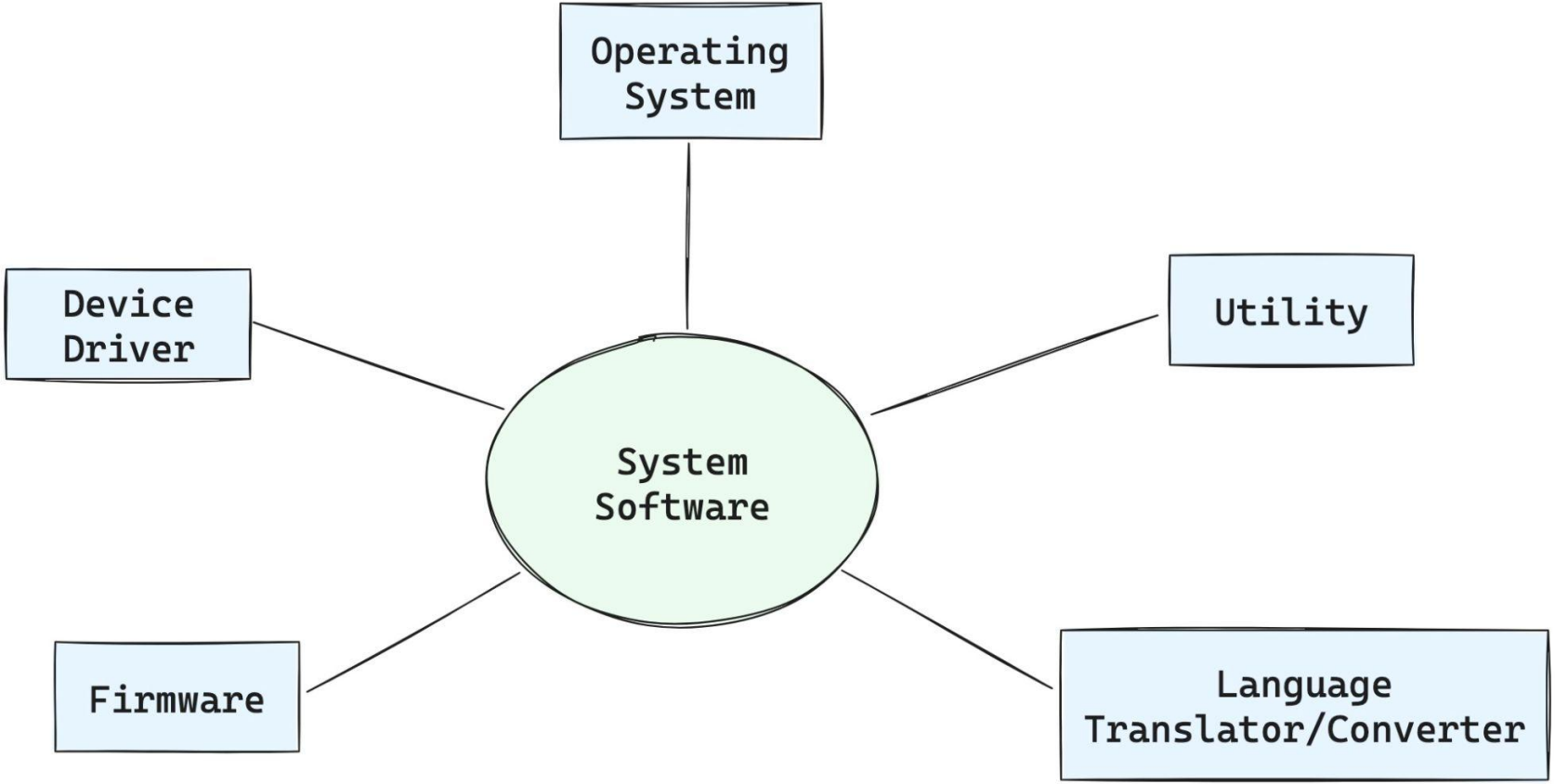
Computer Program

Thanaphoom Babparn

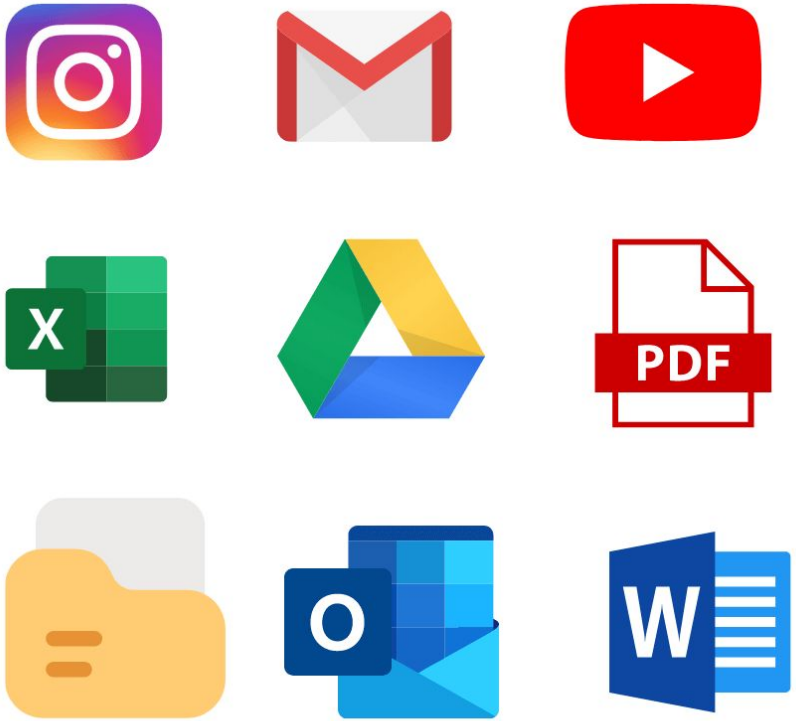
```rust
pub async fn register_user_handler(
    State(data): State<Arc<AppState>>,
    Json(body): Json<RegisterUserSchema>,
) -> Result<impl IntoResponse, (StatusCode, Json<serde_json::Value>)> {
    let user_exists: Option<bool> =
        sqlx::query_scalar("SELECT EXISTS(SELECT 1 FROM users WHERE email = $1)")
            .bind(body.email.to_owned().to_ascii_lowercase())
            .fetch_one(&data.db)
            .await
            .map_err(|e| {
                let error_response = json!({
                    "status": "fail",
                    "message": format!("Database error: {}", e),
                });
                (StatusCode::INTERNAL_SERVER_ERROR, Json(error_response))
            })?;

    if let Some(exists) = user_exists {
        if exists {
            let error_response = json!({
                "status": "fail",
                "message": "User with that email already exists",
            });
            return Err((StatusCode::CONFLICT, Json(error_response)));
        }
    }

    let salt = SaltString::generate(&mut OsRng);
    let hashed_password = Argon2::default()
        .hash_password(body.password.as_bytes(), &salt)
        .map_err(|e| {
            let error_response = json!({
                "status": "fail",
                "message": format!("Error while hashing password: {}", e),
            });
            (StatusCode::INTERNAL_SERVER_ERROR, Json(error_response))
        })
        .map(|hash| hash.to_string())?;
```

```java
package lc.blind75.easy;

import java.util.Arrays;

public class ValidAnagram_242 {

  public boolean isAnagram(String s, String t) {
    // Unicode size (2^16)
    int size = (int) Math.pow(2, 16);
    int[] count1 = new int[size];
    for (char c : s.toCharArray()) {
      count1[c]++;
    }
    int[] count2 = new int[size];
    for (char c : t.toCharArray()) {
      count2[c]++;
    }
    // Time/Space complexity: O(s + t)
    return Arrays.equals(count1, count2);
  }
}
```

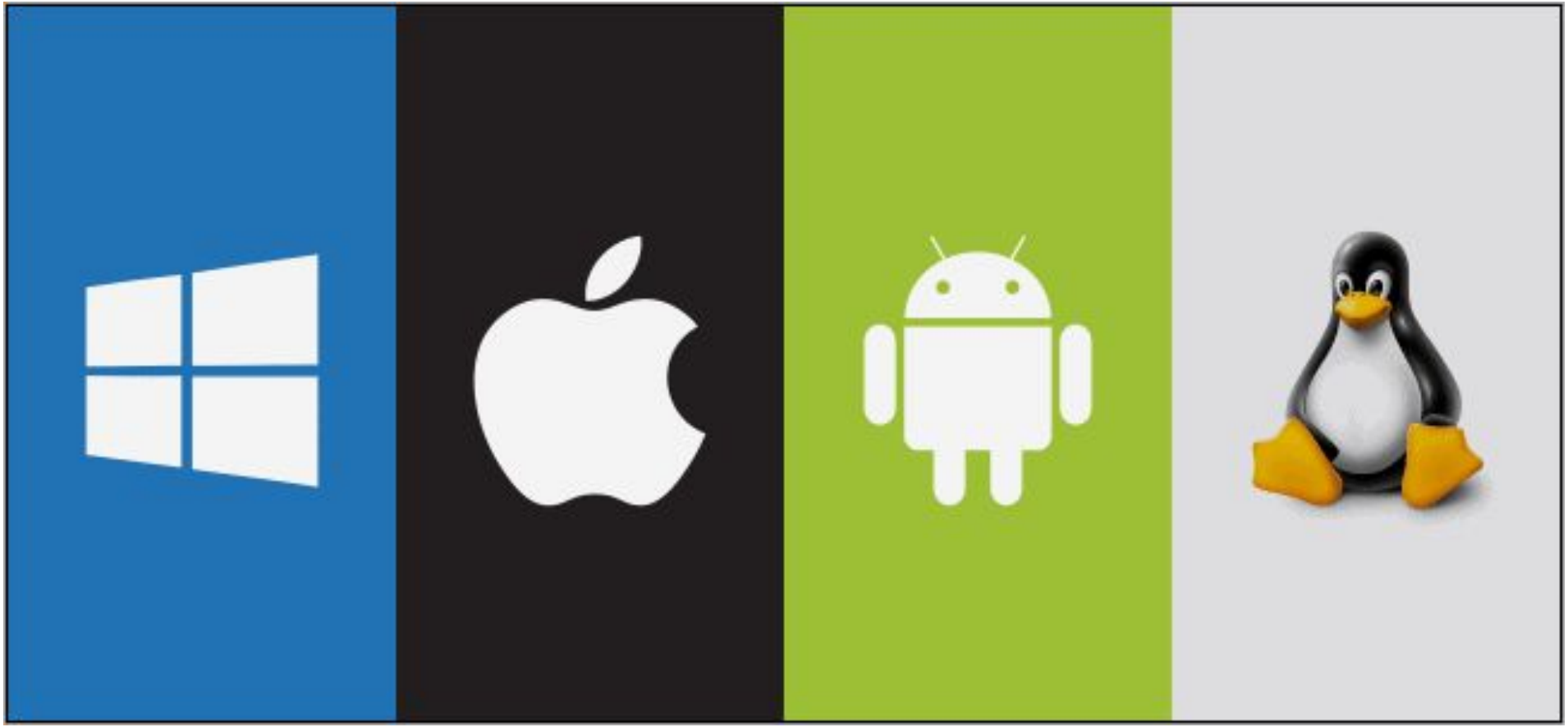

User
End users
Software
(High-Level)
Converter
Place where developer/engineer
working the most
Machine Code - Binary 0/1
(Low-Level)

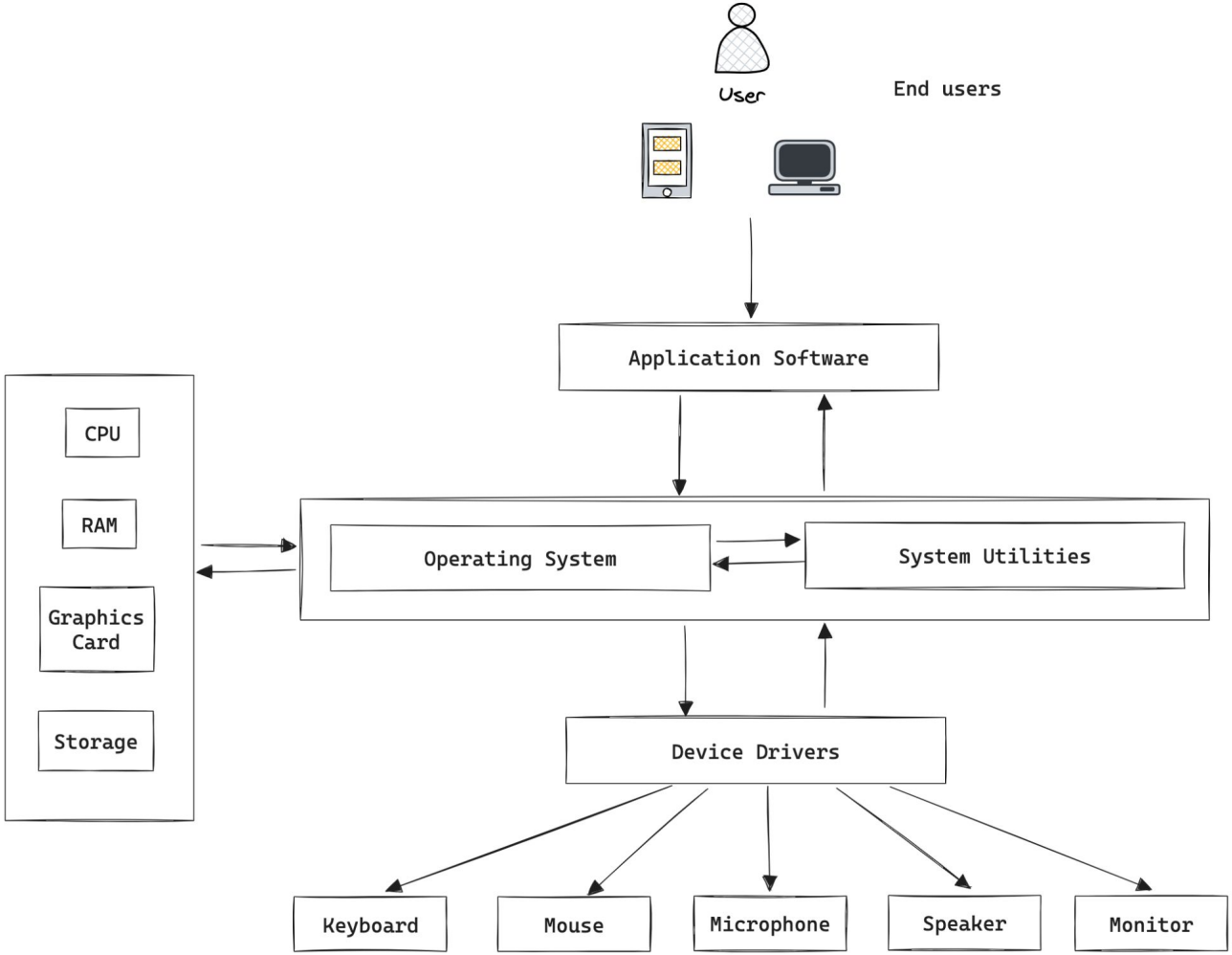

Operating System
Device Driver
Utility
System Software
Firmware
Language Translator/Converter

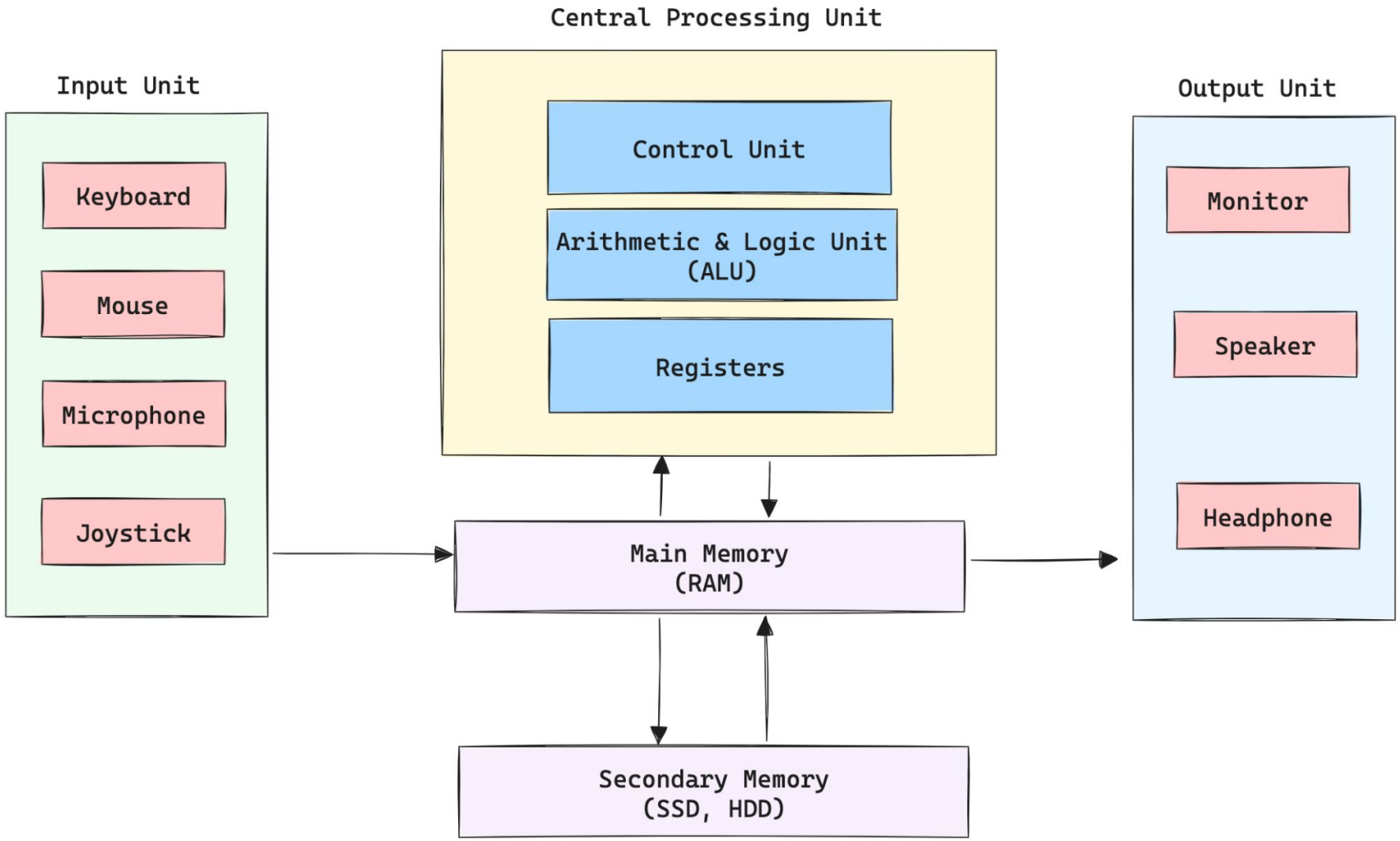
Application Software

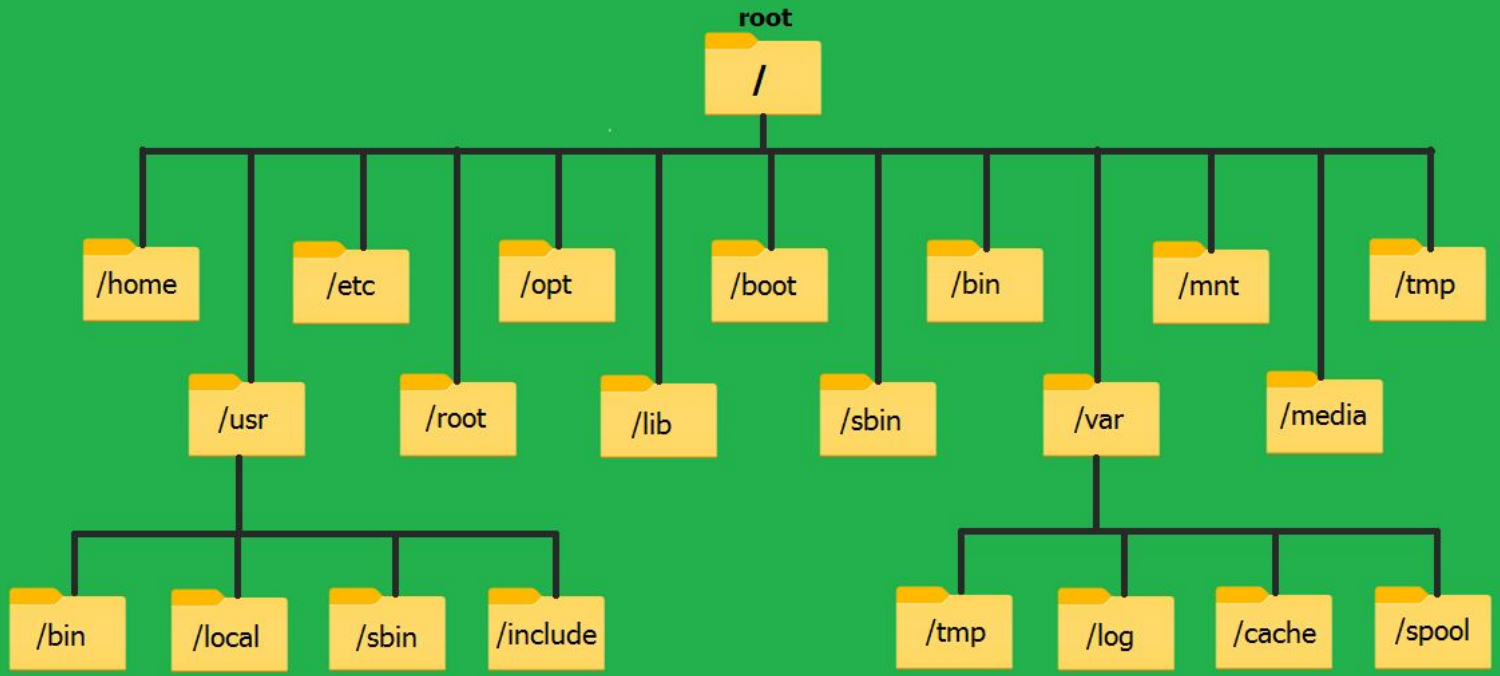


Operating System (OS)




User
End users
Application Software
CPU
RAM
Graphics Card
Storage
Operating System
System Utilities
Device Drivers
Keyboard
Mouse
Microphone
Speaker
Monitor


Central Processing Unit
Input Unit
Output Unit
Control Unit
Arithmetic & Logic Unit
(ALU)
Registers
Keyboard
Mouse
Microphone
Joystick
Monitor
Speaker
Headphone
Main Memory
(RAM)
Secondary Memory
(SSD, HDD)



File System Hierarchy(FHS) of Linux

Homework

1. Describe situation when you are working with input/output devices
2. 5 Example of application software
3. 5 Example of system software
4. Benefit of operating system over computer system
5. Benefit why we have programming languages

End of presentation