📅 Week 7 – CSIS 2260 (Operating Systems) – Survey Week with Practice

Day 1 – Hardware & System Overview

🔍 Concepts

- CPU, memory, storage, I/O devices
- Microprocessor architecture: ALU, control unit, registers
- Instruction cycle: fetch → decode → execute → store

📖 Theory Notes

- **CPU (Central Processing Unit)**: Brain of the computer; executes instructions.
- **ALU (Arithmetic Logic Unit)**: Performs arithmetic (+, -, *, /) and logic (AND, OR, NOT) operations.
- **Control Unit (CU)**: Directs data flow between CPU, memory, and peripherals.
- **Registers**: Small, fast storage locations inside the CPU.
- **Instruction Cycle**:
    1. **Fetch** – Retrieve instruction from memory.
    2. **Decode** – Determine the operation and required data.
    3. **Execute** – Perform the operation.
    4. **Store** – Save the result.

✍️ Pseudocode Test

```
START
  LOAD instruction from memory
  DECODE instruction
  EXECUTE instruction
  STORE result
REPEAT until program ends
END
```

🧪 Code Simulation (Python Example)

```python
instructions = ["ADD 2 3", "MUL 4 5", "SUB 10 7"]

def execute(instruction):
    parts = instruction.split()
    op, a, b = parts[0], int(parts[1]), int(parts[2])
    if op == "ADD":
        return a + b
    elif op == "SUB":
        return a - b
    elif op == "MUL":
        return a * b

for instr in instructions:
    print(f"FETCH: {instr}")
    print("DECODE")
    result = execute(instr)
    print(f"EXECUTE → result = {result}")
    print("STORE result in register\n")
```

❓ Quiz

1. Which CPU part performs arithmetic? **Answer:** ALU
2. Which step of the instruction cycle comes after fetch? **Answer:** Decode

---

✅ Deliverable for Day 1

- Notes on CPU components & instruction cycle – **Done** ✅
- Completed pseudocode test – **Done** ✅
- Working code simulation in Python or JS – **Done** ✅
- Quiz answers recorded – **Done** ✅

Day 2 – OS Introduction & Roles

🔍 Concepts

- OS as a resource manager
- Types of OS: batch, time-sharing, distributed, real-time
- Popular OS families: Windows, Linux/Unix, Android

📖 Theory Notes

- **Operating System**: Software that manages hardware resources and provides services for application programs.
- **Core roles**:
  - **Resource Management**: CPU scheduling, memory allocation, I/O control.
  - **User Interface**: CLI (Command Line Interface) or GUI (Graphical User Interface).
  - **Security & Access Control**: Manages permissions.
- **Types of OS**:
  - **Batch** – Processes jobs in batches without user interaction.
  - **Time-sharing** – Allows multiple users to share system resources simultaneously.

- **Distributed** – Multiple computers working together as one.
- **Real-time** – Guarantees response within a strict time limit.

✍️ Pseudocode Test

```
IF resource request
  CHECK availability
  IF available THEN allocate
  ELSE wait
ENDIF
```

🧪 Code Simulation (Python Example)

```python
resources = {"CPU": 1, "Printer": 1}
requests = ["CPU", "Printer", "CPU"]

for req in requests:
    print(f"Requesting {req}...")
    if resources.get(req, 0) > 0:
        resources[req] -= 1
        print(f"Allocated {req}")
    else:
        print(f"{req} not available, waiting...")
```

❓ Quiz

1. Name 3 types of operating systems.
2. Which OS type is best for air-traffic control?

✅ Deliverable for Day 2

- Notes on OS roles & types.
- Completed pseudocode test.
- Working code simulation in Python or JS.
- Quiz answers recorded.