Larbi Ben M'hidi University – Oum El Bouaghi

Faculty of Exact Sciences and Nature and Life Sciences

Department of Mathematics and Computer Science

Module: Operating Systems II

Lecturer: Dr. Moustafa Sadek KAHIL

2025 – 2026

Tutorial Series 2

Exercise 1:

In a computer system, there are four files: F1, F2, F3, and F4, and three programs: A, B, and C, having the following structures:

Program A	Program B	Program C
Actions A1	Actions B1	Actions C1
Actions A2 (*Read* F1)	Actions B2 (Write F2)	Actions C2 (Read F3)
Actions A3 (*Read* F4)	Actions B3 (Write F3)	Actions C3 (Write F2)
Actions A4 (*Read* F2)	Actions B4 (Write F1)	Actions C4 (Write F4)
Actions A5	Actions B5	Actions C5

Each file cannot be read or modified simultaneously by multiple processes. Furthermore, a file cannot be modified by multiple processes at the same time.

1. Identify the critical sections for each file.
2. Identify the sections that are mutually exclusive.

Exercise 2:

Two processes, P1 and P2, share a critical resource. They must ensure that they do not simultaneously enter their critical section. The solution uses a global lock variable, which takes the values 1 (available) or 0 (busy).

lock = 1; // The lock is initialized to 1 (resource available)

P1	P2

while (true) {
// Non-critical section
while (lock == 0); // Active waiting if the resource is locked
lock = 0; // Take the lock
// Beginning of the critical section
// Use of the common resource
lock = 1; // Free the lock
// End of the critical section
}
	while (true) {
	// Non-critical section
	while (lock == 0); // Active waiting if the resource is locked
	lock = 0; // Take the lock
	// Beginning of the critical section
	// Use of the common resource
	lock = 1; // Free the lock
	// End of the critical section
	}

- Study the feasibility of mutual exclusion

Exercise 3:

At a train station, a single ticket window sells tickets. Passengers must wait their turn to be served. A mechanism is needed to:

1. Manage the passenger queue.
2. Synchronize access to the ticket window to ensure that only one passenger is served at a time.
3. Release the ticket window once service is complete.

Question: Suggest a solution to this problem using a semaphore.

Exercise 4:

A road intersection has two intersecting roads:

1. The North-South (NS) road.
2. The East-West (EW) road.

To avoid collisions, only vehicles traveling on one road may cross the intersection at a time.

- Vehicles coming from both directions arrive at random times.
- Each vehicle must wait its turn to cross the intersection.

- A traffic signal is used to control access to the intersection.

Question: Write a semaphore-based algorithm to manage vehicle access to the intersection in a safe manner.

Exercise 5:

In a laundromat, a solution is needed to allocate M washing machines among customers. Each washing machine is identified by a number. To obtain a number, each customer must use the function $allocate()$. After using the machine, they must use the function $free()$.

- Solve this problem using monitors.

Exercise 6:

A railway line connecting two cities, A and B, includes a single-track section. Trains are represented by processes.

Two types of processes exist:

- Process A to B: represents trains traveling from city A to city B.
- Process B to A: represents trains traveling from city B to city A.

Two trains arriving from opposite directions cannot cross the single-track section at the same time.

Question: Solve the problem using the monitors.