

## **CSC10007 – HỆ ĐIỀU HÀNH**

### **BÀI TẬP LÝ THUYẾT**

#### **BTLT-04: ĐỒNG BỘ HOÁ VÀ BẾ TẮC**

#### **I. Thông tin chung**

Mã số bài tập:	BTLT-04
Thời lượng dự kiến:	~ 1H
Quy định nộp bài:	Nộp trên Moodle cuối buổi học
Hình thức:	Bài tập cá nhân
GV phụ trách:	Vũ Thị Mỹ Hằng
Thông tin liên lạc với GV:	<a href="mailto:vtmhang@fit.hcmus.edu.vn">vtmhang@fit.hcmus.edu.vn</a>

#### **II. Mục tiêu và Quy định**

**Mục tiêu:** Bài tập nhằm giúp sinh viên ôn tập lại kiến thức các chủ đề liên quan đến: Tiểu trình, Đồng bộ hoá dữ liệu (**Chương 4**) và Bế tắc (**Chương 5**). **Lưu ý:** Sinh viên *chỉ cần nộp bài tập được yêu cầu trên lớp (không cần nộp tất cả các bài có trong file này)*.

**Quy định:** Sinh viên tuân thủ các quy định bên dưới.

- Sinh viên làm bài **cá nhân** trên **giấy và chụp lại hình** (chữ viết phải rõ ràng) hoặc làm trực tiếp **trên máy tính**.
- Sinh viên nộp lại bài vào **cuối giờ trên Moodle**.
- Trên giấy hoặc trong file bài làm phải **ghi rõ MSSV và Họ tên**.
- Nếu có nhiều file nộp cần nén lại thành một folder (**.zip**).

#### **III. Yêu cầu bài tập**

1. Consider an online shopping process consisting of the following steps:

-> makeOrder  
-> receiveOrder  
-> requireAdditionalInformation  
-> provideRequiredInformation  
-> verifyReceivedInformation  
-> deliverProduct  
-> receiveProduct

The following processes Client and Shopping allow performing the above steps:

<b>Client {</b> makeOrder(); provideRequiredInformation(); receiveProduct(); } <b>}</b>	<b>Shopping {</b> receiveOrder(); requireAdditionalInformation(); verifyReceivedInformation(); deliverProduct(); } <b>}</b>
---	--

Synchronize processes Client and Shopping Stored by using semaphores to ensure the order of the above shopping process.

2. To form a H<sub>2</sub>O molecule, one oxygen atom and two Hydrogen atoms are needed. Let us suppose three processes P1, P2 and P3 allowing to create an Oxygen atom, a Hydrogen, and a H<sub>2</sub>O molecule, respectively, as below:

P1 { create1Oxygen(); }	P2 { create1Hydrogen(); }	P3 { create1H2o(); }
-------------------------------	---------------------------------	----------------------------

Write pseudo-code for these three processes and use semaphores to synchronize them so that:

- Different processes P1 and P2 can coexist at the same time in the system.
  - Only one process P3 performs its function at a time in the system.
  - If one Oxygen atom and two Hydrogen atoms have been produced, all instances of P1 and P2 will be blocked until all existing atoms assembled into H<sub>2</sub>O molecule (P3). P3 will be blocked if there are not enough oxygen and hydrogen atoms.
3. Consider a system consisting of 4 resources of the same type that are shared by 2 processes, each of which needs at most 3 resources. Can deadlock occur? Explain your answer.
4. Consider a system consisting of m resources of the same type that are shared by 4 processes, each of which needs at most 3 resources. What is the minimum value of m that ensures no deadlock?
5. Consider a system having 6 six tape drives sharing by n processes, each of which needs at most 2 tape disks.
- What is the minimum value of n that can lead the system to deadlock situation?
  - What is the maximum value of n that can ensure no deadlock?
6. Consider a system has 4 processes P1, P2, P3, and P4 sharing a resource R. P1, P2, P3, and P4 needs at most 3, 2, 1, and 4 units of the resource R, respectively. What is the minimum number of units of R that can ensure no-deadlock?
7. Consider the following snapshot of a system.

	Allocation					Max					Need (Max – Allocation)				
	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5	R1	R2	R3	R4	R5
P1	2	3	3	1	1	4	5	7	1	1					
P2	1	0	2	1	4	2	2	3	1	5					
P3	5	2	2	2	2	5	3	7	2	3					
P4	4	1	0	1	0	5	2	1	2	2					
P5	1	1	3	1	1	4	5	3	1	1					

Available				
R1	R2	R3	R4	R5
3	2	2	1	1

- Fill in the matrix Need.
- Is the system in a safe state **Safe**? Explain your answer.
- If P2 requests for (1, 2, 2, 1, 0), can the request be granted? Explain your answer.
- If P1 requests for (2, 2, 2, 0, 0), can the request be granted? Explain your answer.