**Assignment 1 :**

***Question 1:*** Tại sao luôn chạy ra giá trị bằng 0

Thread 1 **tăng giá trị** biến toàn cục value lên 1 triệu lần.

Thread 2 **giảm giá trị** biến toàn cục value xuống 1 triệu lần.

Thì luôn luôn ra giá trị bằng 0:

**Vì** : Chạy tuần tự : Thread 1 hoàn thành toàn bộ công việc **tăng giá trị** trước. Sau đó, Thread 2 bắt đầu và **giảm giá trị** xuống đúng bằng số lượng đã tăng. Vì vậy Số lần tăng và giảm **cân bằng tuyệt đối**, vì cả hai thực hiện cùng số lần (1 triệu lần).

* Không có sự chồng chéo : Khi chỉ có một thread chạy tại một thời điểm, không có tranh chấp tài nguyên. Do đó, giá trị biến luôn chính xác và không bị ảnh hưởng bởi lỗi đồng bộ.
* Cơ chế đồng bộ tự nhiên : join() đảm bảo rằng Thread 1 phải hoàn thành trước khi Thread 2 bắt đầu. Điều này giúp quá trình chạy tuần tự, không có tác động chồng chéo lên biến value.

***Question 2 :*** Tại sao luôn chạy ra giá trị khác đáp án

* Điều kiện tranh chấp : Cả hai thread cùng truy cập và sửa đổi biến value tại cùng thời điểm , Khi Thread 1 đọc giá trị value, Thread 2 có thể đã sửa đổi giá trị đó **trước khi Thread 1 ghi lại**. Điều này dẫn đến lỗi **mất cập nhật** khiến tổng giá trị không chính xác.
* Không có đồng bộ : Hai thread không chờ nhau hoàn thành trước khi truy cập biến.
* Lỗi ngẫu nhiên : Kết quả phụ thuộc vào thứ tự thực thi của các thread, vốn không được đảm bảo.
* Vì vậy, mỗi lần chạy chương trình, kết quả có thể khác nhau.