

## Lab 5:

Xây dựng chương trình phân tán sau sử dụng RPC/ Java RMI.

Chương trình gồm 1 server và một client.

Server cung cấp một phương thức từ xa, tên là **SapXep()** . Phương thức này cho phép sắp xếp lại một mảng số nguyên dương sao cho *các số chẵn được chuyển về đầu dãy, các số lẻ chuyển về cuối dãy*.

Client sử dụng phương thức từ xa đó để sắp xếp các mảng số nguyên dương của nó(các mảng này có thể được sinh ngẫu nhiên hoặc nhập từ bàn phím).

## Lab 4:

Xây dựng chương trình phân tán sau sử dụng socket trên UDP/TCP.

Chương trình gồm một server và một hoặc nhiều client.

Client(liên tục) gửi các dãy số nguyên dương ngẫu nhiên lên server, mỗi số phân tách nhau bằng ký tự trắng.

Server có nhiệm vụ **sắp xếp các dãy vừa nhận sao cho các số chẵn được chuyển về đầu dãy, các số lẻ chuyển về cuối dãy**, sau đó trả về cho client dãy số mới đó.

## Lab 3:

Sử dụng các cơ sở đồng bộ hóa (semaphore, monitor, barrier, ...) để xây dựng chương trình đa luồng thực hiện theo yêu cầu :

- Luồng 1: cứ sau một khoảng thời gian x ngẫu nhiên sẽ sinh ra 1 số nguyên ngẫu nhiên trong khoảng từ 1 đến 500.
- Luồng 2: cứ sau một khoảng thời gian y ngẫu nhiên sẽ lấy lần lượt các số mà Luồng 1 sinh ra rồi kiểm tra số đó có phải số nguyên tố không và hiển thị ra màn hình "<Số>:YES" hoặc "<Số>:NO".

## Lab 2:

Xây dựng thuật toán Bakery.

## Lab 1:

Xây dựng chương trình đồng thời với các yêu cầu sau:

- Tạo một mảng A gồm N số nguyên dương ngẫu nhiên( $N \geq 50$ )
- Sinh ra hai luồng  $T_1$ ,  $T_2$ .
- Luồng  $T_1$  sẽ tìm và in ra các số chia hết cho 3 trong mảng A.
- Luồng  $T_2$  sẽ tìm và in ra các số chia hết cho 5 trong mảng A.
- Hiển thị các số tìm được của 2 luồng đan xen nhau.
- Hiển thị hết các số chia hết cho 5, sau đó mới hiển thị các số chia hết cho 3.