**Bài thực hành C10**

**Đa luồng trong Java**

**Bài 1:**

Tạo file **MyThread1.java** thực thi interface Runnable và thực hiện các công việc sau:

* Trong phương thức run() chứa một vòng lặp in ra 10 số tự nhiên đầu tiên và mỗi lần in cách nhau 500 milliseconds.
* Tạo 2 object từ class MyThread1 là Thread1 và Thread2 với thứ tự ưu tiên tương ứng mà MAX\_PRIORITY và MIN\_PRIORITY.
* Gọi phương thức start() cho cả 2 thread.

# Bài 2:

Tạo file **MyThread2.java**, và thực hiện theo yêu cầu:

* Tạo thread 1 và thread 2.
* Thread 1 sẽ in ra các số chẵn nằm trong 1 đến 10.
* Thread 2 sẽ in ra các số lẻ nằm trong 1 đến 10.
* Viết đoạn code sao cho chương trình sẽ chạy xong thread 1 thì chạy tiếp sang thread 2.

**Bài 3:**

Tạo file **MyThread3.java**, và thực hiện theo yêu cầu:

* Thread1: Hiển thị các số từ 1 đến 10
* Thread2: Dựa vào các số được hiển thị ở Thead1 in ra tương ứng là “Chẵn” hoặc “Lẻ”. Sử dụng method **wait()** and **notify()** và từ khóa **synchronized** cho các thread trên.

**Bài 4: Xét bài toán chim bồ câu như sau:**

* Có 5 con chim bồ câu sống trong một tổ chim có 3 lổ.
* Mỗi lổ tại một thời điểm chỉ có thể chứa một con chim bồ câu sử dụng làm chổ ngủ.
* Mỗi con chim ngủ 1 giây sau đó bay ra khỏi lổ để đi tìm thức ăn trong 2 giây và sau đó quay về tổ để tìm lổ nào còn trống để ngủ.
* Nếu không còn lổ trống thì chim phải chờ.
* Giả sử lúc tình trạng ban đầu là 5 con chim cùng bay về tổ để tìm chổ ngủ.
* Tạo file **MyThread4.java** minh họa bài toán này và có xuất ra các câu thông báo (ví dụ) như sau:
  + Chim 1 vào lổ 1 để ngủ
  + Chim 4 vào lổ 2 để ngủ
  + Chim 2 vào lổ 3 để ngủ
  + Chim 3 đang chờ
  + Chim 5 đang chờ
  + Chim 1 bay ra lổ 1
  + Chim 5 vào lổ 1 để ngủ
  + Chim 4 bay ra lổ 2
  + Chim 3 vào lổ 2 để ngủ
  + Chim 1 đang chờ
  + Chim 2 bay ra lổ 3