## Bài 1: Hoàn thiện bài ngăn xếp, hàng đợi

## Bài 2: Thêm phương thức xóa cho cây nhị phân tìm kiếm

## 2.1. Không đệ quy

## 2.2. Đệ quy

## Bài 3: Tiếp tục hoàn thiện các bài dạng 3

### Công thức:

(1) Giả thiết;

(2) Kiểm tra giả thiết: // Giả thiết đúng ==> tăng đếm // Giả thiết sai ==> Cập nhật lại giả thiết; Reset giá trị đếm về 1

Ví dụ: Hãy xác định giá LN và in ra số lần xuất hiện cùng các ví trị xuất hiện của nó

int [] A={12, 3, 434, 12, 43, 434, 38,55 };

int [] pos = new int[A.length];

(1) int max=A[0]; int cnt=1; // Giả thiết A[0] LN và xuất hiện 1 lần

(2) pos[cnt-1]=0;

for (int i=1; i<A.length;i++) {

if (max==A[i]) { cnt++; pos[cnt-1]=i;

}

else if (max<A[i]) { max=A[i]; cnt=1; pos[cnt-1]=i;

}

}

System.out.println("Max = "+ max);

System.out.println("So lần xuất hiện: "+ cnt);

System.out.println("Xuất hiện ở các vị trí sau: ");

for (int i=0;i<cnt;i++) System.out.print(pos[i]+" ");

/\* Thay đổi đoạn in kết quả theo yêu cầu sau:

(1) Nếu chỉ xuất hiện 1 lần: Max = ... ở vị trí: ....

(2) Nếu xuất hiện nhiều lần:

Max = ... Số lần: ....

Ở các vị trí: 3, 5, 6, ...10. \*/

### Bài tập áp dụng:

(1) Cho mảng int [] A={5, 3, 7, 8, 4, 6, 8, 3}. // Hãy xác định và in tất cả các dãy con tăng dần dài nhất cùng các vị trí bắt đầu xuất hiện của nó

(2) Cho mảng int [] A={5, 1, 9, 10,16, 4,6, 8, 3}. // Hãy xác định và in tất cả các dãy con tăng dần dài nhất có tổng các phần tử lớn hơn 100 cùng các vị trí bắt đầu xuất hiện của nó

(3) Cho mảng int [] A={5, 1, 9, 10,16, 4,6, 8, 3}. // Hãy xác định và in ra tất cả các dãy con dài nhất gồm các phần tử là các số chẵn.