**Bài 2:** Một danh sách các phần tử được lưu trữ trong một danh sách đặc, có các phần tử sau: 40, 70, 20, 60, 90, 10, 50, 30. Yêu cầu:

2.6 Dùng phương pháp xếp thứ tự HeapSort, mô tả từng bước quá trình xếp thứ tự dãy số trên (không lập trình). Tín độ phức tạp của quá trình xếp thứ tự danh sách trên.

Bước 1:

Dãy 1: a[0]->a[3] : 40 70 20 60

Dãy 2: a[4]->a[7] : 90 10 50 30

i=3, so sánh a[3]=60, a[7]=30 => Không swap

40 70 20 60 | 90 10 50 30

i=2, so sánh a[2]=20, a[5]=10, a[6]=50 => Swap (a[2], a[6]) => Heap max

40 70 50 60 | 90 10 20 30

i=1, so sánh a[1]=70, a[3]=60, a[4]=90 => Swap (a[1], a[4]) => Heap max

40 90 50 60 | 70 10 20 30

i=0, so sánh a[0]=40, a[1]=90, a[2]=50 => Swap (a[0], a[1]) => Heap max

90 40 50 60 | 70 10 20 30

i=1, so sánh a[1]=40, a[3]=60, a[4]=70 => Swap (a[1], a[4]) => Heap max

90 70 50 60 | 40 10 20 30

i=2, so sánh a[2]=50, a[5]=10, a[6]=20 => Không swap

90 70 50 60 | 40 10 20 30

i=3, so sánh a[3]=60, a[7]=30 => Không swap

90 70 50 60 | 40 10 20 30

90 70 50 60 | 40 10 20 30

* 30 70 50 60 | 40 10 20 90

Bước 2 :

40 30 50 60 20 10 70 90

Bước 3:

10 40 50 30 20 60 70 90

Bước 4:

20 40 10 30 50 60 70 90

Bước 5:

30 20 10 40 50 60 70 90

Bước 6:

10 20 30 40 50 60 70 90

**Dãy sau khi sắp xếp :**

**10 20 30 40 50 60 70 90**

Tại vị trí i chọn hai phần tử có vị trí 2i+1 và 2i+2 so sánh i và max(2i+1,2i+2) nếu i nhỏ hơn thì hoán vị, tới vị trí i=0, hoán vị vị trí i=0 và i=n. Ta xét từ vị trí n/2-1

Độ phức tạp dãy đã cho có thứ tự trung bình=> Độ phức tạp là O(nlogn)