***BÀI TẬP CHƯƠNG 3 – BÀI 2.6***

***Quá trình mô tả xếp thứ tự HEAP SORT:***

Với một danh sách đặc a[], có n phần tử từ a[0] đến a[6] như sau: a[0], a[1], a[2], a[3], a[4], a[5], a[6],a[7].

7

6

5

4

3

2

1

0

Vị trí:

a[4]

a[3]

a[7]

a[6]

a[5]

a[2]

a[1]

a[0]

Phần tử:

**30**

**50**

**10**

**90**

**60**

**20**

**70**

**40**

Giá trị:

***Bước 1:*** *Tạo HEAP ban đầu:*

6

5

4

3

2

0

1

Chia dãy trên thành 2 đoạn bao gồm:

Nữa dãy bên trái chứa các phần tử sau a[0],…., a[(n/2)-1]: 40, 70, 20, 60

Nữa dãy bên phải chứa các phần tử sau a[n/2],…., a[n-1]: 90, 10, 50, 30

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**50**

**10**

**20**

**40**

**90**

Tại vị trí cuối cùng của nữa dãy con bên trái i=3, so sánh a[i] =a[3] = 60 với hai phần tử tại vị trí 2\*i+1 = 7 và vị trí 2\*i+2 =

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**50**

**10**

**20**

**40**

**90**

swap(a[3], a[7])

Giá trị lớn nhất của a[3], a[7] là a[3] = 60. Ta không đổi chỗ a[3] và a[7]:

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**50**

**10**

**20**

**40**

**90**

Tiếp tục giảm i xuống 1 giá trị (i=2), và so sánh a[2], a[5], a[6]

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**50**

**10**

**20**

**40**

**90**

swap(a[2], a[6])

Giá trị lớn nhất của a[2], a[5], a[6] là a[6] = 50. Thực hiện hoán vị a[2] và a[6]

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**20**

**10**

**50**

**40**

**90**

Tiếp tục giảm i xuống 1 giá trị (i=1), và so sánh a[1], a[3], a[4]

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**20**

**10**

**50**

**40**

**90**

swap(a[1], a[4])

Giá trị lớn nhất của a[1], a[3], a[4] là a[4] = 90. Thực hiện hoán vị a[1] và a[4]

0 1 2 3 4 5 6 7

**60**

**90**

**30**

**20**

**10**

**50**

**40**

**70**

Tiếp tục giảm i xuống 1 giá trị (i=0), và so sánh a[0], a[1], a[2]

0 1 2 3 4 5 6 7

**60**

**90**

**30**

**20**

**10**

**50**

**40**

**70**

swap(a[0], a[1])

Giá trị lớn nhất của a[], a[1], a[2] là a[1] = 90. Thực hiện hoán vị a[0] và a[1]

2

0

1

0 1 2 3 4 5 6 7

**60**

**40**

**30**

**20**

**10**

**50**

**90**

**70**

**Xét lại tính lan truyền tại vị trí a[1],** sau khi a[1] nhận giá trị mới là 40 (thay thế giá trị 90 trước đó; giá trị a[1] = 90 trước đó thỏa tính của Heap so với a[3] và a[4])

0 1 2 3 4 5 6 7

**60**

**40**

**30**

**20**

**10**

**50**

**90**

**70**

swap(a[1], a[4])

Giá trị lớn nhất của a[1], a[3], a[4] là a[4] = 70. Thực hiện hoán vị a[1] và a[4]

Việc tạo Heap (Heap Max) ban đầu hoàn tất. Ta được một Heap sau:

0 1 2 3 4 5 6 7

**60**

**70**

**30**

**20**

**10**

**50**

**90**

**40**

**Bước 2:** *Hoán vị phần tử a[0] và phần tử cuối Heap đang xét. Ta có kết quả sau:*

0 1 2 3 4 5 6 7

**60**

**70**

**90**

**20**

**10**

**50**

**30**

**40**

***Bước 3:*** *Trong dãy đang xét, giới hạn phần tử cuối dãy. Ta được dãy sau:*

0 1 2 3 4 5 6 7

**90**

**20**

**10**

**40**

**60**

**70**

**50**

**30**

Tạo Heap ban đầu lại cho dãy các phần tử đang xét từ a[0], a[1], …,a[6]

Trong trường hợp này thực chất chỉ xét lại vị trí a[0] (và sự lan truyền nếu có), các vị trí còn ại từ a[1],…, a[6] đã thỏa tính chất Heap trước đó.

***Bướ 4:*** Sau khi dãy từ a[0], a[1], …., a[6] là một Heap, hoán vị a[0] và a[6]. Tiếp túc xét lại dãy từ a[0] đến a[5]….*Lặp lại* bước này cho đến khi danh sách được xếp thứ tự tăng dần

***\*Độ phức tập của quá trình xếp thứ tự danh sách trên:*** O( n log n)