**Chương 3 – BT làm thêm**

Bài tập 2:

40 70 20 60 90 10 50 30

**a/ InsertionSort**

B1: Giả sử phần tử đầu tiên đã có thứ tự:

40 70 20 60 90 10 50 30

(j : 0 -> i-1)

Xét i = 1, j = 0 a[i] = 70 > a[j] = a[0] = 40

-> Chèn a[i] vào bên phải a[j]

40 70 20 60 90 10 50 30

Xét i = 2, j = 0 a[i] = 20 < a[j] = a[0] = 40

-> Chèn a[i] vào bên trái a[j]

20 40 70 60 90 10 50 30

Xét i = 3, j = 0 a[i] = 60 > a[j] = 20 -> j++

j = 1 a[i] = 60 > a[j] = 40 -> j++

j = 2 a[i] = 60 < a[j] = 70

-> Chèn a[i] vào bên trái a[j]

20 40 60 70 90 10 50 30

Xét i = 4, j = 0 a[i] = 90 > a[j] = 20 -> j++

j = 1 a[i] = 90 > a[j] = 40 -> j++

j = 2 a[i] = 90 > a[j] = 60 -> j++

j = 3 a[i] = 90 > a[j] = 70

-> Chèn a[i] vào bên phải a[j]

20 40 60 70 90 10 50 30

Xét i = 5, j = 0 a[i] = 10 < a[j] = 20

-> Chèn a[i] vào bên trái a[j]

10 20 40 60 70 90 50 30

Xét i = 6, j = 0 a[i] = 50 > a[j] = 10 -> j++

j = 1 a[i] = 50 > a[j] = 20 -> j++

j = 2 a[i] = 50 > a[j] = 40 -> j++

j = 3 a[i] = 50 < a[j] = 60

-> Chèn a[i] vào bên trái a[j]

10 20 40 50 60 70 90 30

Xét i = 7, j = 0 a[i] = 30 > a[j] = 10 -> j++

j = 1 a[i] = 30 > a[j] = 20 -> j++

j = 2 a[i] = 30 < a[j] = 40

-> Chèn a[i] vào bên trái a[j]

10 20 30 40 50 60 70 90

->Xong. Kết Thúc

# b/ Phương pháp SelectionSort

# 40, 70, 20, 60, 90, 10, 50, 30

Bước 1: I = 0 Duyệt từ 0 đến 7: a[0] đến a[7]

* 40, 70, 20, 60, 90, 10, 50, 30
* Min\_pos = 5
* Đổi chổ a[0] với a[5]
* Ta được dãy mới

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

Bước 2: I = 1 Duyệt từ 1 đến 7: a[1] đến a[7]

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

* Min\_pos = 2
* Đổi chổ a[1] với a[2]
* Ta được dãy mới

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

Bước 3: I = 2 Duyệt từ 2 đến 7: a[2] đến a[7]

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

* Min\_pos = 7
* Đổi chổ a[2] với a[7]
* Ta được dãy mới

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

Bước 4: I = 3 Duyệt từ 3 đến 7: a[3] đến a[7]

* 10, 20, 30, 60, 90, 40, 50, 70
* Min\_pos = 5
* Đổi chổ a[3] với a[5]
* Ta được dãy mới

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

Bước 5: I = 4 Duyệt từ 4 đến 7: a[4] đến a[7]

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

* Min\_pos = 6
* Đổi chổ a[4] với a[6]
* Ta được dãy mới

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

Bước 6: I = 5 Duyệt từ 5 đến 7: a[5] đến a[7]

* 10, 20, 30, 40, 50, 60, 90, 70
* Min\_pos = 5
* Ta vẫn giữ nguyên dãy cũ:

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

Bước 7: I = 6 Duyệt từ 6 đến 7: a[6] đến a[7]

* 10, 20, 30, 40, 50, 60, 90, 70
* Min\_pos = 7
* Đổi chổ a[6] với a[7]
* Ta được dãy mới

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

Vậy sau khi thực hiện 7 bước sắp xếp ta có dãy mới tăng dần là :

# 10, 20, 30, 40, 50, 60, 70, 90

**c/ Phương pháp InterchangeSort**

40 70 20 60 90 10 50 30

Bước 1: i = 0:

j = i + 1 = 1, Xét a[0] = 40 < a[1] = 70 -> Swap

70 40 20 60 90 10 50 30

j = 2, Xét a[0] = 70 > a[2] = 20 -> Không Swap

j = 3, Xét a[0] = 70 > a[3] = 60 -> Không Swap

j = 4, Xét a[0] = 70 < a[4] = 90 -> Swap

90 40 20 60 70 10 50 30

j = 5, Xét a[0]=90 > a[5]=10 -> Ko swap

j = 6, Xét a[0]=90 > a[6]=50 -> Ko swap

j = 7, Xét a[0]=90 > a[7]=30 -> Ko swap

90 40 20 60 70 10 50 30

Bước 2: i=1

j =2, Xét a[1]=40 > a[2]=20 -> Không Swap

j=3, Xét a[1]=40 < a[3]=60 -> Swap

90 60 20 40 70 10 50 30

j = 4, xét a[1]=60 < a[4]=70 -> Swap

90 70 20 40 60 10 50 30

j=5 Xét a[1]=70 > a[5]=10 -> ko swap

j=6 Xét a[1]=70 > a[6]=50 ->Ko swap

j=7 Xét a[1]=70 > a[7]=30 -> Ko swap

90 70 20 40 60 10 50 30

Bước 3: i = 2

j=3 Xét a[2]=20 < a[3]=40 -> Swap

90 70 40 20 60 10 50 30

J=4 Xét a[2]=40 <a[4]=60 -> Swap

90 70 60 20 40 10 50 30

J=5 Xét a[2]=60 > a[5]=10 ->ko swap

J=6 xét a[2]=60 > a[6]=50 -> ko swap

J = 7 xét a[2]=69 > a[7]=30 -> ko swap

90 70 60 20 40 10 50 30

Bước 4: i=3

J=4 Xét a[3]=20 < a[4]=40 ->swap

90 70 60 40 20 10 50 30

J=5 Xét a[3]=40 > a[5]=10 -> ko swap

J=6 Xét a[3]=40 < a[6]=50 -> swap

90 70 60 50 20 10 40 30

J=7 Xét a[3]=50 > a[7]=30 -> ko swap

90 70 60 50 20 10 40 30

Bước 5: i=4

J=5 Xét a[4]=20 > a[5]=10 -> ko swap

J=6 Xét a[4]=20 < a[6]=40 -> swap

90 70 60 50 40 10 20 30

J=7 Xét a[4]=40 > a[7]=30 -> ko swap

90 70 60 50 40 10 20 30

Bước 6: i=5

J =6 Xét a[5]=10 < a[6]=20 -> swap

90 70 60 50 40 20 10 30

J=7 Xét a[5]=20 < a[7]=30 -> swap

90 70 60 50 40 30 10 20

90 70 60 50 40 30 10 20

Bước 7: i=6

J=7 Xét a[6]=10 < a[7]=20-> swap

90 70 60 50 40 30 20 10

Vậy sau khi sắp xếp ta dc: 90 70 60 50 40 30 20 10

**d/ Sắp xếp mãng theo phương pháp BubbleSort**:

Bước 1: i=0.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 30<50.

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

J=6 xét a[i] và a[j-1] ⬄ a[6] và a[5] = 50>10 => swap.

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

J=5 xét a[i] và a[j-1] ⬄ a[5] và a[4] = 50<90.

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

J=4 xét a[i] và a[j-1] ⬄ a[4] và a[3] = 90>60 => swap.

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

J=3 xét a[i] và a[j-1] ⬄ a[3] và a[2] = 90>20 => swap.

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

J=2 xét a[i] và a[j-1] ⬄ a[2] và a[1] = 90>70 => swap.

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

J=1 xét a[i] và a[j-1] ⬄ a[1] và a[0] = 90>40 => swap.

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

Dừng(j=0=i)

Bước 2: i=1.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 30>10 => swap.

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

J=6 xét a[i] và a[j-1] ⬄ a[6] và a[5] = 30<50.

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

J=5 xét a[i] và a[j-1] ⬄ a[5] và a[4] = 50<60.

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

J=4 xét a[i] và a[j-1] ⬄ a[4] và a[3] = 60>20 =>swap.

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

J=3 xét a[i] và a[j-1] ⬄ a[3] và a[2] = 60<70.

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

J=2 xét a[i] và a[j-1] ⬄ a[2] và a[1] = 70>40 => swap.

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

Dừng (j=1=i).

Bước 3: i=2.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 10<30.

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

J=6 xét a[i] và a[j-1] ⬄ a[6] và a[5] = 30<50.

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

J=5 xét a[i] và a[j-1] ⬄ a[5] và a[4] = 50>20 =>swap.

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

J=4 xét a[i] và a[j-1] ⬄ a[4] và a[3] = 50<60.

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

J=3 xét a[i] và a[j-1] ⬄ a[3] và a[2] = 60>40 => swap.

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

Dừng (j=2=i).

Bước 4: i=3.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 10<30.

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

J=6 xét a[i] và a[j-1] ⬄ a[6] và a[5] = 30>20 => swap.

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

J=5 xét a[i] và a[j-1] ⬄ a[5] và a[4] = 30<50.

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

J=4 xét a[i] và a[j-1] ⬄ a[4] và a[3] = 50>40 => swap.

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

Dừng(J=3=i).

Bước 5: i=4.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 10<20.

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

J=6 xét a[i] và a[j-1] ⬄ a[6] và a[5] = 20<30.

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

J=5 xét a[i] và a[j-1] ⬄ a[5] và a[4] = 30<40.

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

Dừng(J=4=i).

Bước 6: i=5.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 10<20.

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

J=6 xét a[i] và a[j-1] ⬄ a[6] và a[5] = 20<30.

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

Dừng(J=5=i).

Bước 7: i=6.

J=7 xét a[i] và a[j-1] ⬄ a[7] và a[6] = 10<20.

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

Dừng(J=6=i).

* Vậy ta được mãng:  **90**, **70**, **60**, **50**, **40**, **30**, **20**, 10.

Độ Phức Tạp của bài toán: O(n2).

**Câu 2.5: QuickSort**

40 70 20 60 90 10 50 30

l = 0, r = 0  
B1: Chọn x = a[(l+r)/2] = a[3] = 60

I = 0 a[0] = 40 < x = 60 => i++  
 I = 1 a[0] = 70 > x = 60 => dừng

J = 7 a[7] = 30 < x = 60 => dừng

Swap a[i] với a[j] ~ a[1] với a[7]

40 30 20 60 90 10 50 70  
=> i++, j-- ; i = 2 ; j = 6

i = 2 a[2] = 20 < x = 60 -> i++  
i = 3 a[3] = 60 > x = 60 -> dừng  
  
j = 6 a[6] = 50 < x = 60 -> dừng

B2:

i = 2 a[2] = 20 < x = 60 -> i++   
i = 3 a[3] = 60 > x = 60 -> dừng

j = 6 a[6] = 50 < x = 60 -> dừng

-> Swap a[i] với a[j] ~ a[3] với a[6]

40 30 20 50 90 10 60 70

-> i++, j-- (i = 4, j = 5)  
Bước 3:

i = 4 a[4] = 90 < x = 60 -> dừng  
j = 5 a[5] = 10 < x = 60 -> dừng

-> Swap a[i] với a[j] (a[4] với a[5])

40 30 20 50 10 90 60 70

->i++, j-- (i = 5, j = 4) - > dừng (i < j)

\*Ta có 2 dãy con: 40, 30, 20, 50, 10 ( dãy 1)

90, 60, 70 (dãy 2)

**Dãy 1: 40, 30, 20, 50, 10**

L=0 , r=4

Chọn x = a[2] = 20

Bước 1:

i = 0 , a[0] = 40 > x=20 🡪 dừng

j = 4 , a[4] = 10 < x = 20 🡪 dừng

* Swap a[i] với a[j] ( a[0] với a[4] )
* 10 30 20 50 40
* i ++ , j- - ( i=1 , j= 3)

Bước 2:

i = 1 , a[1] = 30 > x=20 -> dừng

j = 3 , a[3] = 50 > x=20 -> j- -

j = 2 , a[2] = 20 > x=20 -> dừng

* Swap a[i] với a[j] ( a[1] với a[2] )
* 10 20 30 50 40
* i++, j- - ( i = 2 , j = 1) 🡺 dừng ( i>j )

10, 20, 30, 50, 40

Ta có 2 dãy con: 10, 20 ( dãy 1\*)

30, 50, 40 (dãy 2\*)

Dãy 1\*: 10, 20

L=0 , r=1

Chọn x=a[0]=10

Bước 1:

i = 0 , a[0] = 10 < x = 10 -> dừng

j = 1 , a[1] = 20 > x = 10 -> j++

i ++ = 1 , j- - = 0 ( i> j) 🡺 dừng ( không swap )

* 10, 20 (\*)

Dãy 2\*: 30, 50, 40

L=0 , r= 2

Chọn x= a[1] = 50

Bước 1:

i=0, a[0] = 30 < x=50 -> i++;

i=1, a[1] = 50 < x=50 -> dừng

j=2 , a[2] = 40 < x=50 -> dừng

🡺Swap a[i] với a[j] ( a[1] với a[2] )

🡺30, 40 , 50

* i++ = 2 , j- - = 1 🡺 dừng ( i>j )
* 30, 40, 50 (\*\*)

Từ (\*) và (\*\*) 🡺 10, 20, 30, 40, 50 (+)

Dãy 2: 90, 60, 70

L=0 , r = 2

Chọn x = a[1] = 60

Bước 1:

i=0 , a[0] = 90 > x=60 -> dừng

j=2 , a[2] = 70 > x=60 -> j- -

j=1 , a[1] = 60 > x=60 -> dừng

🡺Swap a[i] với a[j] ( a[0] với a[1] )

* 60, 90, 70
* i++ = 1 , j- - = 0 🡺 dừng ( i>j )
* 60, 90, 70

Ta có 2 dãy con: 60 (1) 🡺 đã có thứ tự

90, 70 (2)

(2) 🡺 90, 70

L= 0 , r= 1

Chọn x = a[0] = 90

Bước 1:

i=0 , a[0] = 90 < x= 90 -> dừng

j=1 , a[1] = 70 < x= 90 –> dừng

🡺Swap a[i] với a[j] ( a[0] với a[1] )

* 70, 90

Từ dãy 2 🡺 60, 70, 90 (++)

**Từ (+) và (++) 🡺 10 20 30 40 50 60 70 90**

**2.6 Dùng phương pháp xếp thứ tự HeapSort**

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)

**Câu 2.7**

Độ phức tạp của thuật toán tìm kiếm tuần tự cho phần tử 90 là : O(n)

Độ phức tạp của thuật toán tìm kiếm nhị phân cho phần tử 90 là : O(log n)