***BÀI TẬP LÀM THÊM CHƯƠNG 3***

***Họ và tên*** *: Nguyễn Thị Thanh Hằng*

***MSSV:*** *1654050032*

***Nhóm*** *: 02*

**Đề bài** :

***Câu 2.3****: Sắp xếp các phần tử trong danh sách đặc sau :* ***40 , 70 , 20 , 60 , 90 , 10 , 50 , 30***  *bằng phương pháp xếp thứ tự Interchange Sort. Tính độ phức tạp của quá trình xếp thứ tự danh sách trên*

**Bài làm** :

* ***Sắp xếp danh sách theo thứ tự tăng dần:***

40

70

20

60

30

50

10

90

**a[i] 0 1 2 3 4 5 6 7**

**Lần 1** : *xét i = 0, j đi từ 1 đến 7*

* Xét j = 1 : a[0] = 40 và a[j]=a[1]= 70, a[0]< a[1] => Không hoán vị , tăng j (j= 2)

40

70

20

60

30

50

10

90

* Xét j = 2 : a[0] = 40 và a[j]=a[2]= 20, a[0] > a[2] => Hoán vị (a[2],a[0]), tăng j (j= 3)

20

70

40

60

30

50

10

90

* Xét j = 3: a[0] = 20 và a[j]= a[3]= 60, a[0] < a[3] => Không hoán vị , tăng j (j=4)
* Xét j = 4: a[0] = 20 và a[j]= a[4]= 90, a[0] < a[4] => Không hoán vị , tăng j (j=5)
* Xét j = 5 : a[0] = 20 và a[j]=a[5]= 10, a[0] > a[5] => Hoán vị (a[5],a[0]), tăng j (j= 6)

10

70

40

60

30

50

20

90

* Xét j = 6: a[0] = 10 và a[j]= a[6]= 50, a[0] < a[6] => Không hoán vị , tăng j (j=7)
* Xét j = 7: a[0] = 10 và a[j]= a[7]= 30, a[0] < a[7] => Không hoán vị , tăng j (j=8) , j > n-1 => Dừng j = 7, tăng i lên ( i = 1). Kết quả lần 1:

***10***

70

40

60

30

50

20

90

**Lần 2**: *xét i = 1, j đi từ 2 đến 7*

* Thực hiện tương tự lần 1 => Ta được kết quả sau lần 2 :

***10***

***20***

70

60

30

50

40

90

**Lần 3**: *xét i = 2, j đi từ 3đến 7*

* Thực hiện tương tự lần 2 => Ta được kết quả sau lần 3 :

***10***

***20***

***30***

70

40

50

60

90

**Lần 4**: *xét i = 3, j đi từ 4 đến 7*

* Thực hiện tương tự lần 3 => Ta được kết quả sau lần 4 :

***10***

***20***

***30***

***40***

50

60

70

90

**Lần 5**: *xét i = 4, j đi từ 5 đến 7*

* Thực hiện tương tự lần 4 => Ta được kết quả sau lần 5 :

***10***

***20***

***30***

***40***

60

70

90

***50***

**Lần 6**: *xét i = 5, j đi từ 6 đến 7*

* Thực hiện tương tự lần 5=> Ta được kết quả sau lần 6:

***10***

***20***

***30***

***40***

70

90

***60***

***50***

**Lần 6**: *xét i = 6, j = 7*

* Thực hiện tương tự lần 5=> Ta được kết quả sau lần 6:

***10***

***20***

***30***

***40***

***90***

***70***

***60***

***50***

* ***Kết quả sau khi sắp xếp dãy ta được dãy sau:***

***10***

***20***

***30***

***40***

***90***

***70***

***60***

***50***

* Tính độ phức tạp của thuật toán Interchange Sort

Chương trình :

Void Interchange\_Sort ( int a[], int n)

{

for ( int i = 0; i< n -1;i++)

{

for ( int j = i+ 1; j < n ; j++)

{

if(a[i] > a[j])

Swap(a[i],a[j]);

}

}

}

Mỗi vòng lặp for có độ phức tạp là O(n), đoạn code chương trình sắp xếp theo phương pháp Interchange Sort có 2 vòng lặp for => Độ phức tạp là O(n2).