2.4 Dùng phương pháp xếp thứ tự BubbleSort, 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

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

Ta có vị trí 40 cho j=0 🡪 vị trí 30 j=7

Xét j=7

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

Ta thấy j=7 và vị trí j-1 là j=6 lớn hơn nên ta tiến hành đổi chổ

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

Ta xét tiếp vị trí j=6 và j=5

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

Ta thấy j6 lớn hơn nên giữ nguyên

Ta tiếp tục xét j=5 và j=4

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

j = 4 lớn hon ta đổi chổ

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

Xét j = 4 và j = 3 ta thấy j =3 lớn hơn ta đổi chổ vs j =4

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

Xét j=3 và j = 2 ta thấy j = 2 lớn hơn ta đổi chổ vs j = 3

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

Xét j=2 và j=1 ta thấy j=1 lớn hơn ta đổi chổ vs j=2

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

Xét j=1 và j=0 ta thấy j=0 lớn hơn ta đổi chổ vs j =1

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

Dùng tương tự từng cặp số từ vị trí j=7🡪 vị trí j=1 ta được

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

Dùng tương tự từng cặp số từ vị trí j=7🡪 vị trí j=2 ta được

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

Dùng tương tự từng cặp số từ vị trí j=7🡪 vị trí j=3 ta được

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

Dùng tương tự từng cặp số từ vị trí j=7🡪 vị trí j=2 ta được

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

Ta thấy dãy số không còn sự thay thế thì trình tự sắp xếp đã thành công

Đáp án cuối cùng là 10,20, 30, 40, 50, 70, 60,90

2.tính độ phức tạp :

void BubbleSort(int []a, int n)

{

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

for(int j=n-1;j>i; j--)

if(a[j-1] >a[j]); // xét điều kiện phần tử sau nhỏ hơn phần tử trước

swap(a[j],a[j-1]) *// hoán vị a[j] với a[j-1]*

}

Độ phức tạp của thuật toán: **O(n2)** do có 2 vòng lặp for