40 70 20 60 90 10 50 30

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

Dùng pp InterchangeSort

* i=0 => a[0]=40

j = i+1 => a[1]=70

a[0] < a[1] : Không hoán vị j++ ( j=2 < n-1= 7 )

a[0] > a[2]: Hoán vị 40, 10. j++ ( j=3 < 7 )

20 70 40 60 90 10 50 30

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

a[0] <a[3]: Không hoán vị j++ ( j=4 < 7 )

a[0] <a[4]: Không hoán vị j++ ( j=5 < 7 )

a[0] >a[5]: Hoán vị 20, 10. j++ ( j=6 < 7 )

10 70 40 60 90 20 50 30

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

a[0] <a[5]: Không hoán vị j++ ( j=7 = 7 )

a[0] <a[5]: Không hoán vị j++ ( j=8 > 7 )

Dừng j

10 70 40 60 90 20 50 30

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

* i++ (i=1)

j=i+1 (j =2)

a[1] > a[2]: Hoán vị 70, 40 j++ ( j=3 < 7 )

10 40 70 60 90 20 50 30

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

a[1] <a[3]: Không hoán vị j++ ( j=4 < 7 )

a[1] <a[4]: Không hoán vị j++ ( j=5 < 7 )

a[1] >a[5]: Hoán vị 40, 20. j++ ( j=6 < 7 )

10 20 70 60 90 40 50 30

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

a[1] <a[6]: Không hoán vị j++ ( j=7 = 7 )

a[1] <a[7]: Không hoán vị j++ ( j=8 > 7 )

Dừng j

10 20 70 60 90 40 50 30

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

Lặp lại tương tự với các bước sau

* **B1** Tăng i++ (i=3)
* **B2** Xét phần tử j = i+1

Lặp lại sao cho thỏa j <= n-1

{

Nếu a[i] > a[j] thì hoán vị 2 phần tử

j++

}

* **B3** i++

Nếu i < n -1, lặp lại bước **B2.**

Và kết quả thu được là dãy đã được sắp xếp

10 20 30 40 50 60 70 90