

Bubble Sort – Βασικές διεργασίες

```
int ar[] = {3,5,4,7,2,8,1,0,9,6};
```

"Bubble sort (...) is a simple sorting algorithm that repeatedly steps through the input list element by element, comparing the current element with the one after it, swapping their values if needed. These passes through the list are repeated until no swaps had to be performed during a pass, meaning that the list has become fully sorted. The algorithm, which is a comparison sort, is named for the way the larger elements "bubble" up to the top of the list." *Wikipedia*

- Δύο είναι οι βασικές διεργασίες που εκτελεί
 - Σύγκριση δύο στοιχείων (**comparing** the current element with the one after it)
 - Εναλλαγή θέσης αν η σύγκριση το απαιτεί (**swapping** their values if needed)
- Με την επαναλαμβανόμενη εκτέλεση των δύο αυτών διεργασιών επιτυγχάνει την ταξινόμηση των στοιχείων του πίνακα
 - **repeatedly steps** through the input list element by element
 - **These passes through the list** are repeated

Bubble Sort – A step-by-step Approach

"Bubble sort (...) is a simple sorting algorithm that repeatedly steps through the input list element by element, comparing the current element with the one after it, swapping their values if needed. These passes through the list are repeated until no swaps had to be performed during a pass, meaning that the list has become fully sorted. The algorithm, which is a comparison sort, is named for the way the larger elements "bubble" up to the top of the list." *Wikipedia*

```
int ar[] = {3,5,4,7,2,8,1,0,9,6};
```

- Στόχος μας να αναπτύξουμε την bubbleSort βήμα-προς-βήμα.
 - **Step 1** – Υλοποιεί το κίτρινο μέρος της περιγραφής
Κάνει moveBubbleUp για τον πίνακα ar με στοιχεία numOfElements
 - **Step 2** – Υλοποιεί την επανάληψη του Step 1 για όλους τους υπο-πίνακες του ar μέχρι τον υπο-πίνακα με 2 στοιχεία (πράσινο μέρος περιγραφής).
Στην ουσία επαναλαμβάνει την δουλειά του βήματος 1, δηλαδή το moveBubbleUp, για κάθε υπο-πίνακα του ar μέχρι αυτόν με 2 στοιχεία.
 - **Step 3** – Βελτίωση του βήματος 2 για να αποφύγουμε τις άσκοπες επαναλήψεις

1st Step – moveBubbleUp for ar (1/2)

"Bubble sort (...) is a simple sorting algorithm that repeatedly steps through the input list element by element, comparing the current element with the one after it, swapping their values if needed. (Wikipedia)"

- Ονομάζουμε **moveBubbleUp** τη διεργασία που περιγράφεται παραπάνω με κίτρινο
- Αυτή είναι η πρώτη δουλειά που πρέπει να κάνει η bubbleSort
- Με βάση αυτό η bubbleSort διαμορφώνεται όπως παρακάτω

```
28 void bubbleSort(int ar[], int numElements){  
29     moveBubbleUp(ar, numElements);  
30 }  
31  
32 void moveBubbleUp(int ar[], int numElements){
```

Δώστε το σώμα της moveBubbleUp

1st Step – moveBubbleUp for ar (2/2)

Ο πίνακας

```
int ar[] = {3,5,4,7,2,8,1,0,9,6};
```

Screenshot εκτέλεσης 1^{ης} έκδοσης

C:\Code\courses\I2P2023-24_ X + v

3 5 4 7 2 8 1 0 9 6

Array status: not sorted

Press any key to continue . . .

Iteration No2->3 4 5 7 2 8 1 0 9 6

Iteration No4->3 4 5 2 7 8 1 0 9 6

Iteration No6->3 4 5 2 7 1 8 0 9 6

Iteration No7->3 4 5 2 7 1 0 8 9 6

Iteration No9->3 4 5 2 7 1 0 8 6 9

End of array pass

To iteration είναι η επανάληψη μιας εκτέλεσης της σύγκρισης δυο στοιχείων και εναλλαγής θέσης τους αν απαιτείται.

Press any key to continue . . .

3 4 5 2 7 1 0 8 6 9

Array status: not sorted

Process exited after 15.26 seconds with return value 0

Press any key to continue . . .

Δώστε τον πηγαίο κώδικα για το βήμα αυτό (V1)

Next Step for bubbleSort ?

Ο πίνακας πριν την εκτέλεση του Step 1

```
C:\Code\courses\I2P2023-24_ x + v
3      5      4      7      2      8      1      0      9      6
Array status: not sorted
```

Ο πίνακας μετά την εκτέλεση του Step 1 (moveBubbleUp)

```
3      4      5      2      7      1      0      8      6      9
Array status: not sorted
```

Next step

```
3      4      2      5      1      0      7      6      8
Array status: not sorted
```

Next step

```
3      2      4      1      0      5      6      7
Array status: not sorted
```

Πόσα steps;

2nd Step – MoveBubbleUp for all sub-arrays

Screenshot εκτέλεσης 2ης έκδοσης (1st part)

```
C:\Code\courses\I2P2023-24_ x + v
3      5      4      7      2      8      1      0      9      6
Array status: not sorted
Press any key to continue . . .
Iteration No2->3      4      5      7      2      8      1      0      9      6
Iteration No4->3      4      5      2      7      8      1      0      9      6
Iteration No6->3      4      5      2      7      1      8      0      9      6
Iteration No7->3      4      5      2      7      1      0      8      9      6
Iteration No9->3      4      5      2      7      1      0      8      6      9

End of array pass for 10 elements
Press any key to continue . . .
Iteration No3->3      4      2      5      7      1      0      8      6
Iteration No5->3      4      2      5      1      7      0      8      6
Iteration No6->3      4      2      5      1      0      7      8      6
Iteration No8->3      4      2      5      1      0      7      6      8

End of array pass for 9 elements
Press any key to continue . . .
Iteration No2->3      2      4      5      1      0      7      6
Iteration No4->3      2      4      1      5      0      7      6
Iteration No5->3      2      4      1      0      5      7      6
Iteration No7->3      2      4      1      0      5      6      7

End of array pass for 8 elements
```

Αναβαθμίστε τον πηγαίο κώδικα του βήματος 1 (V1) για να υλοποιήσει την εκτέλεση της λειτουργικότητας moveBubbleUp για όλους τους υπο-πίνακες του ar (V2).

2nd Step – MoveBubbleUp for all sub-arrays

Screenshot εκτέλεσης 2^{ης} έκδοσης (last part)

```
Iteration No1->1      2      0      3      4
Iteration No2->1      0      2      3      4

End of array pass for 5 elements
Press any key to continue . . .
Iteration No1->0      1      2      3

End of array pass for 4 elements
Press any key to continue . . .
End of array pass for 3 elements
Press any key to continue . . .
End of array pass for 2 elements
Press any key to continue . . .
0      1      2      3      4      5      6      7      8      9
Array status: sorted

-----
Process exited after 17.62 seconds with return value 0
Press any key to continue . . .
```

Παρατηρήστε τις άσκοπες επαναλήψεις για 3 και 2 στοιχεία

3rd Step – Avoid meaningless iterations

Screenshot εκτέλεσης 3^{ης} έκδοσης (last part)

```
Press any key to continue . . .
Iteration No2->2      1      3      0      4      5
Iteration No3->2      1      0      3      4      5

End of array pass for 6 elements
Press any key to continue . . .
Iteration No1->1      2      0      3      4
Iteration No2->1      0      2      3      4

End of array pass for 5 elements
Press any key to continue . . .
Iteration No1->0      1      2      3

End of array pass for 4 elements
Press any key to continue . . .
End of array pass for 3 elements
Press any key to continue . . .
0      1      2      3      4      5      6      7      8      9
Array status: sorted

-----
Process exited after 12.04 seconds with return value 0
Press any key to continue . . .
```

Αναβαθμίστε τον πηγαίο κώδικα του βήματος 2 (V2) για να αποφύγετε την εκτέλεση της λειτουργικότητας moveBubbleUp για τους ταξινομημένους υπο-πίνακες του ar (V3).