

Ταξινομηση

Ταξινόμηση

```
def find_largest(N, L):  
    """Return the N largest values in L in order from  
    smallest to largest."""  
  
    copy = L[:]   
    copy.sort()  
    return copy[:N]
```

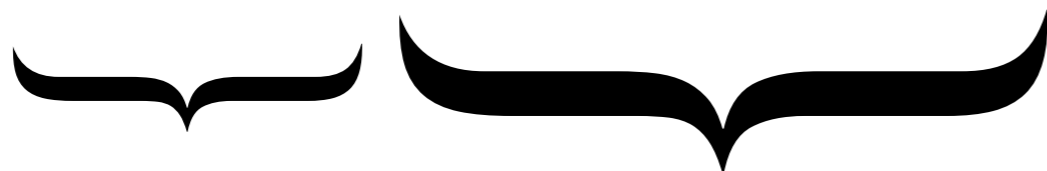
- Πώς πραγματοποιείτε η ταξινόμηση;
- Υπάρχουν διάφοροι αλγόριθμοι
- Απλοί και πιο εύκολοι στην κατανόηση
- Αλλά και πολύπλοκοι και “πιο δύσκολοι”

Ταξινόμηση με επιλογή

Selection Sort

- Σε κάθε βήμα της ταξινόμησης χωρίζουμε τη λίστα σε δύο μέρη ένα **ταξινομημένο** και ένα **μη ταξινομημένο**
- Όλα τα στοιχεία του ταξινομημένου είναι μικρότερα από αυτά του μη ταξινομημένου

1	2	5	10	29	25	12	21	15
---	---	---	----	----	----	----	----	----



ταξινομημένο

μη ταξινομημένο

Ταξινόμηση με επιλογή

- Βρίσκουμε το ελάχιστο στοιχείο του μη ταξινομημένου μέρους

1	2	5	10	29	25	12	21	15
---	---	---	----	----	----	----	----	----

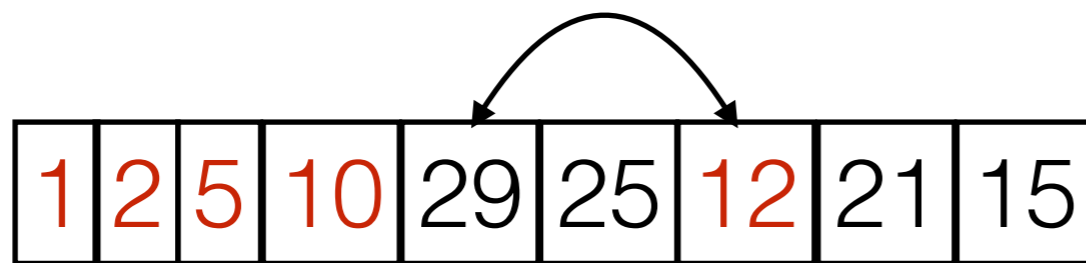


ταξινομημένο

μη ταξινομημένο

Ταξινόμηση με επιλογή

- Αλλάζουμε θέση ανάμεσα στο ελάχιστο στοιχείο και το πρώτο στοιχείο του μη ταξινομημένου μέρους



ταξινομημένο μη ταξινομημένο

Ταξινόμηση με επιλογή

- Το ταξινομημένο μέρος αυξάνετε και ελαττώνετε αντιστοίχα το μη ταξινομημένο

1	2	5	10	12	25	29	21	15
---	---	---	----	----	----	----	----	----



ταξινομημένο

μη ταξινομημένο

Ταξινόμηση με επιλογή

- 1ο βήμα:
Βρίσκουμε το ελάχιστο στοιχείο της λίστας και το βάζουμε πρώτο
- Επόμενα βήματα:
Ξεκινώντας με το ταξινομημένο μέρος (που στην αρχή έχει ένα στοιχείο), βρίσκουμε το ελάχιστο στοιχείο του μη ταξινομημένου και το θέτουμε ως τελευταίο στοιχείο του ταξινομημένου

Ταξινόμηση με επιλογή

```
def selection_sort(L):  
    """Reorder the items in L from smallest to  
    largest."""
```

```
    i = 0 ← μετρητής. Πόσα στοιχεία έχω ταξινομήσει
```

```
    while i != len(L):
```

```
        # Βρες το δείκτη του μικρότερου στοιχείου της του μη ταξινομημένου L[i:].
```

```
        # Αντάλλαξε τη θέση αυτού του στοιχείου με το στοιχείο L[i].
```

```
        i = i + 1 ← Αυξάνω το μετρητή
```

Ταξινόμηση με επιλογή

```
def selection_sort(L):  
    """Reorder the values in L from smallest to  
    largest."""  
  
    i = 0  
    while i != len(L):  
        # Βρες το δείκτη του μικρότερου στοιχείου της του μη ταξινομημένου L[i:].  
        # Καλώ το δείκτη αυτό smallest  
        L[i], L[smallest] = L[smallest], L[i]  
        i = i + 1
```

Ανταλλάσσω τη θέση του στοιχείου με δείκτη smallest στοιχείο L[i]

Ταξινόμηση με επιλογή

```
def selection_sort(L):  
    """Reorder the values in L from smallest to largest."""  
  
    i = 0  
    while i != len(L):  
        smallest = find_min(L, i)  
        L[i], L[smallest] = L[smallest], L[i]  
        i = i + 1
```

← συνάρτηση που δίνει
τον δείκτη του
ελάχιστου στοιχείο
από τη θέση i μέχρι το
τέλος

```
def find_min(L, b):  
    """Return the index of the smallest value in L[b:]."""  
    Γραμμική αναζήτηση από τη θέση b μέχρι το τέλος
```

Ταξινόμηση με επιλογή

```
def selection_sort(L):
    """Reorder the values in L from smallest to largest."""

    i = 0
    while i != len(L):
        smallest = find_min(L, i)
        L[i], L[smallest] = L[smallest], L[i]
        i = i + 1

def find_min(L, b):
    """Return the index of the smallest value in L[b:]."""

    smallest = b # The index of the smallest so far.
    i = b + 1
    while i != len(L):
        if L[i] < L[smallest]:
            # We found a smaller item at L[i].
            smallest = i

        i = i + 1

    return smallest
```