Ταξινόμηση

λίστα ονομάτων

Ταξινόμηση

- L=['John Lennon', 'Albert Einstein', 'Steve Jobs', 'John Kennedy', 'Mata Hari', 'Marie Curie’].

- Ταξινόμηση με βάση μια σχέση διάταξης

- Σχέση διάταξης:

    - Διάταξη αριθμών

    - Λεξικογραφική

    - Άλλη διάταξη που θεωρούμε

Ταξινόμηση

Λεξικογραφική

- ‘a’<‘b’: True
- ‘Aa’<‘Ba’: True
- ‘Aac’<‘Aab’: False

Ταξινόμηση

Λεξικογραφική

Αλγόριθμος Python για ταξινόμηση λίστας αριθμών

```
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:]."""
```

Μεταβολή για να εφαρμοστεί σε string

Ταξινόμηση

Λεξικογραφική

Αλγόριθμος Python για ταξινόμηση λίστας strings

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

```
def find_min(L, b):  ←—— Μεταβολή για να εφαρμοστεί σε string
    """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]:  ←—— Έλεγχος διάταξης
            # We found a smaller item at L[i].
            smallest = i

        i = i + 1

    return smallest
```

Ταξινόμηση

Άλλη διάταξη

- Tuple (x,y), x,y αριθμοί
- Μια διάταξη που μπορούμε να ορίσουμε είναι: (x1,y1) μικρότερο του (x2,y2) αν |x1|+|y1|<|x2|+|y2|
- L=[(1,2),(0,1),(-1,2),(3,0),(4,-1)]

Ταξινόμηση

Άλλη διάταξη

Αλγόριθμος Python για ταξινόμηση λίστας tuples

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

```
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 my_less(L[i],L[smallest]):
            # We found a smaller item at L[i].
            smallest = i

        i = i + 1

    return smallest
```

Μεταβολή για να εφαρμοστεί σε άλλη διάταξη

Έλεγχος διάταξης

Ταξινόμηση

Άλλη διάταξη

Αλγόριθμος Python για ταξινόμηση λίστας tuples

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

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 my_less(L[i],L[smallest]):
            # We found a smaller item at L[i].
            smallest = i
        i = i + 1
    return smallest
def my_less(a,b):
    x=abs(a[0])+abs(a[1])
    y=abs(b[0])+abs(b[1])
    return x<y
```

Μεταβολή για να εφαρμοστεί σε άλλη διάταξη

Έλεγχος διάταξης

Επιστρέφει True ή False

Ταξινόμηση

Άλλη διάταξη

- Θέλουμε να ταξινομήσουμε ένα λεξικό d, με κλειδιά ονόματα και με τιμές ‘Gold’, ‘Silver’, ‘Bronze’

- Π.χ.

```
d={'Mike':'Gold', 'John':'Silver', 'Harris':'Gold', 'Jenifer':'Bronze'}
```

- Το λεξικό δεν υπάρχει η έννοια της διάταξης.

- Αυτό που μπορούμε να πάρουμε είναι μια λίστα με ταξινομημένα στοιχεία του λεξικού.

Ταξινόμηση

- Η εντολή d.keys() και d.values() δίνουν δύο λίστες με τις αντίστοιχες τιμές.

- Οι λίστες έχουν στοιχεία με αντίστοιχες τιμές, δηλαδή το στοιχείο στη θέση i του d.keys() έχει τιμή στο λεξικό d αυτή που βρίσκεται στο d.values() στη θέση i.

- Ταξινομώντας μόνο τη μια από τις λίστες χάνουμε την παραπάνω αντιστοιχία.

Ταξινόμηση

- Στο παραπάνω λεξικό έχουμε
list(d.values())=[‘Gold’, 'Gold', 'Silver', 'Bronze', ‘Silver']

- Η εντολή sorted() ταξινομεί μια λίστα
sorted(d.values())=[‘Bronze’, 'Gold', 'Gold', 'Silver', ‘Silver']

Πλέον το στοιχείο ‘Bronze’ δεν είναι τιμή του στοιχείου στη θέση 0 του d.keys().
Παρατήρηση: Η d.values() δεν έχει τροποποιηθεί

Ταξινόμηση

- Οι τιμές του λεξικού είναι ‘Gold’, ‘Silver’, ‘ Bronze’

- Θέλουμε η διάταξη να γίνει με τη σειρά ‘Bronze’, ‘Silver’, ‘Gold’

Ταξινόμηση

- Ας θεωρήσουμε για παράδειγμα τώρα ένα λεξικό με τιμές αριθμούς

```
d={'Mike':111, 'John':222, 'Harris':555, 'Jennifer':44}
```

- Θέλουμε να δημιουργήσουμε μια λίστα με ταξινομημένους τους αριθμούς που είναι οι τιμές του d, αλλά και τα κλειδιά (keys) να βρίσκονται στις αντίστοιχες θέσεις.

Ταξινόμηση

Λεξικού

Αλγόριθμος Python για ταξινόμηση λεξικού με αριθμούς

```
def selection_sort(d):
    """Create two lists, the values in d from smallest to largest."""
    i = 0;L1=list(d.keys());L2=list(d.values());
    while i != len(L2):
        smallest = find_min(L2, i)
        L2[i], L2[smallest] = L2[smallest], L2[i]
        L1[i], L1[smallest] = L1[smallest], L1[i]
        i = i + 1
    return L1,L2


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

← Η συνάρτηση που θεωρήσαμε στη ταξνόμηση λιστας αριθμών

Ταξινόμηση

Άλλη διάταξη

Αλγόριθμος Python για ταξινόμηση λεξικού με τιμές string

```
def selection_sort(d):
    """Create two lists, the values in d from smallest to largest."""

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 my_less(L[i],L[smallest]):
            # We found a smaller item at L[i].
            smallest = i
        i = i + 1
    return smallest
def my_less(a,b):
    if a==‘Bronze’:
       return True
    elif a==‘Silver’ and b==‘Gold’:
       return True
    else
    return False
```

ταξινόμηση λεξικού σύμφωνα με τον προηγούμενο προβλημα

Μεταβολή για να εφαρμοστεί σε άλλη διάταξη

Έλεγχος διάταξης

Επιστρέφει True ή False