

Ασκήσεις εργαστηρίου

Υπενθύμιση θεωρίας

Μεταβλητές κειμένου (strings)

Στις μεταβλητές κειμένου μπορούμε να εντοπίζουμε ξεχωριστά κάθε χαρακτήρα χρησιμοποιώντας παρόμοιο συμβολισμό με αυτόν που χρησιμοποιούμε στις λίστες, δηλαδή αν $a = \text{"name"}$ το $a[0]$ είναι το n . Έτσι, για να μετρήσουμε το πλήθος των χαρακτήρων σε μια μεταβλητή κειμένου μπορεί να χρησιμοποιηθεί η εντολή `len()`. Δηλαδή $\text{len}(a) = 4$.

Μία άλλη χρήσιμη εντολή για μεγαλύτερα κείμενα είναι η εντολή `.split()`. Με αυτή την εντολή χωρίζεται αυτόματα ένα κείμενο σε λέξεις οι οποίες αποθηκεύονται σε μία νέα λίστα. Για παράδειγμα το κείμενο: $a = \text{"This is a new text"}$ με την εντολή $b = a.\text{split}()$ μετατρέπεται στη λίστα $b = [\text{"This"}, \text{"is"}, \text{"a"}, \text{"new"}, \text{"text"}]$.

Σε αυτή την περίπτωση μπορούμε και πάλι να εντοπίσουμε κάθε χαρακτήρα του κάθε στοιχείου της λίστας αλλά θα χρειαστεί ένας επιπλέον δείκτης, δηλαδή το 2^ο γράμμα της πρώτης λέξης είναι το $b[0][1]$.

Όταν θέλουμε να κάνουμε αναζήτηση σε ένα κείμενο μας βολεύει μια επιπλέον δυνατότητα που έχει η εντολή `for` να συντάσσεται ως εξής: `for i in a` ώστε να διαβάζονται επαναληπτικά όλοι οι χαρακτήρες του κειμένου που βρίσκεται στη μεταβλητή a μέσω της μεταβλητής i . Εναλλακτικά θα μπορούσε να γραφεί ως: `for i in range(len(a))` αλλά τότε θα ελέγχονταν το $a[i]$.

Λίστες

Δύο χρήσιμες εντολές για τις λίστες είναι η `.append()` και η `.pop()`. Με την πρώτη εντολή προσθέτουμε ένα καινούριο στοιχείο στο τέλος της λίστας ενώ με τη δεύτερη αφαιρούμε ένα στοιχείο από το τέλος της λίστας. Για παράδειγμα αν η λίστα είναι $a = [1, 2, 3, 4]$ με το $a.\text{append}(6)$ η λίστα γίνεται $a = [1, 2, 3, 4, 6]$ ενώ αν πρώτα γραφόταν το $a.\text{pop}()$ η λίστα γινόταν $a = [1, 2, 3]$.

Ασκήσεις εργαστηρίου

Άσκηση 1

Γράψτε έναν κώδικα σε Python ο οποίος να διαβάζει ένα ονοματεπώνυμο που έχει καταχωρηθεί σε μία μεταβλητή κειμένου, να τα διαχωρίζει και να τα εμφανίζει με πρώτο το επίθετο. Αν το όνομα έχει πάνω από 5 γράμματα να εμφανίζει μόνο το πρώτο γράμμα.

Άσκηση 2

Γράψτε έναν κώδικα σε Python ο οποίος να διαβάζει ένα κείμενο (τουλάχιστον 1 πρόταση), να μετράει τις προτάσεις, τις λέξεις και τα γράμματα και να εμφανίζει επιπλέον τα εξής στατιστικά στοιχεία για το κείμενο: λέξεις/πρόταση και γράμματα ανά λέξη. Επίσης να εμφανίζει πόσες φορές υπάρχει στο κείμενο το γράμμα "a".

Άσκηση 3

Γράψτε έναν κώδικα σε Python ο οποίος να διαβάζει ένα password και να ελέγχει αν πληροί ορισμένες προϋποθέσεις για είναι ισχυρό, δηλαδή να έχει πάνω από 8 ψηφία, να συνδυάζει αριθμούς, κεφαλαία και μικρά γράμματα και σύμβολα.

Ασκήσεις εργαστηρίου

Άσκηση 4

Γράψτε έναν κώδικα σε Python ο οποίος να διαβάζει μια σειρά από ονόματα και επώνυμα, να τα διαχωρίζει και να τα εισάγει σε δύο λίστες, μία για τα ονόματα και μία για τα επώνυμα. Στη συνέχεια να μετράει πόσα επώνυμα αρχίζουν από φωνήεν και πόσα από σύμφωνο.

Άσκηση 5

Γράψτε έναν κώδικα σε Python ο οποίος να ξεκινάει να διαβάζει αριθμούς που δίνει ο χρήστης και να τους καταχωρεί σε λίστα όταν δοθεί η λέξη “Start” (αλλιώς να δίνει το μήνυμα “Give START to begin”) και να σταματάει τη διαδικασία όταν ο χρήστης γράψει “STOP”. Στο τέλος πρέπει να εμφανίζει το πλήθος των αριθμών που δόθηκαν.

Άσκηση 6

Γράψτε έναν κώδικα σε Python ο οποίος να προσθέτει δύο διανύσματα των οποίων οι τιμές είναι καταχωρημένες σε λίστα και να αποθηκεύει το αποτέλεσμα σε νέα λίστα.

Ασκήσεις εργαστηρίου

Άσκηση 7

Γράψτε έναν κώδικα σε Python ο οποίος να αφαιρεί τα N τελευταία στοιχεία μιας λίστας και να τα γράφει σε μια καινούρια λίστα.

Άσκηση 8

Γράψτε έναν κώδικα σε Python ο οποίος να διαβάζει μια λίστα και να εισάγει σε μια καινούρια λίστα τα στοιχεία της πρώτης λίστας αν είναι πολλαπλάσια του 3 ή αριθμοί πάνω από 10.

Άσκηση 9

Γράψτε έναν κώδικα σε Python ο οποίος να διαβάζει αριθμούς από μία λίστα και να καταχωρεί σε μία νέα λίστα μόνο όσους αριθμούς δεν υπάρχουν ήδη στη λίστα, δηλαδή να μην έχει «αντίγραφα» των ίδιων αριθμών.

Άσκηση 10

Γράψτε έναν κώδικα σε Python ο οποίος να υπολογίζει τον «κινούμενο» μέσο όρο (δηλαδή μέσο όρο με βάση λιγότερα σημεία) για τους αριθμούς που βρίσκονται σε μια λίστα.

Λυμένες ασκήσεις

Άσκηση 1

Αρχικά πρέπει να γραφεί το ονοματεπώνυμο με κενό στη μέση π.χ. a="Name Surname". Για να διαχωριστεί το κείμενο αυτό θα χρησιμοποιηθεί η εντολή .split(). Αφού χρειάζεται να ελεγχθεί το μέγεθος του επιθέτου θα χρησιμοποιηθεί η εντολή if και η εντολή len. Οπότε ο κώδικας θα γραφεί ως εξής:

a="Name Surname"	←	Ορισμός ονόματος και επωνύμου
b=a.split()	←	Διαχωρισμός των λέξεων της μεταβλητής a και καταχώρηση στη λίστα b
if (len(b[0])>5):	←	Υπολογισμός γραμμάτων του ονόματος που είναι το πρώτο στοιχείο της λίστας b άρα b[0]
print(b[1],b[0][0])	←	Εμφάνιση επωνύμου (b[1]) και πρώτου χαρακτήρα του ονόματος που είναι το πρώτο στοιχείο της λίστας άρα b[0][0]
else:		
print(b[1],b[0])	←	Εμφάνιση πλήρους επωνύμου και ονόματος γιατί το όνομα δεν έχει πάνω από 5 χαρακτήρες

Λυμένες ασκήσεις

Άσκηση 2

Αρχικά γράφουμε ένα μεγαλύτερο κείμενο σε μια μεταβλητή. Το κείμενο περιέχει προτάσεις που τελειώνουν με τελεία. Για να μετρήσουμε τις λέξεις θα χρησιμοποιήσουμε την εντολή `.split()` και μετά `len()` στη λίστα που θα δημιουργηθεί. Για να μετρήσουμε τις προτάσεις χρειάζεται να ελέγξουμε (με `for` και `if`) όλους τους χαρακτήρες του κειμένου και όταν εντοπίσουμε τελείες να αυξάνεται η τιμή μιας μεταβλητής που θα μετράει τον αριθμό των προτάσεων. Ο συνολικός αριθμός των γραμμάτων μπορεί να προκύψει με διάφορους τρόπους. Ο ένας είναι να μετρηθεί ο αριθμός χαρακτήρων κάθε λέξης που είναι αποθηκευμένη στη λίστα και να αφαιρεθεί ο αριθμός των επιπλέον χαρακτήρων π.χ. τελείας ή κόμματος που έχουμε στο κείμενο. Για την εύρεση του γράμματος “a”, θα ακολουθήσουμε την ίδια διαδικασία με την αναζήτηση της τελείας στο κείμενο.

Οπότε ο κώδικας γράφεται ως εξής:

```
a="This is a new text."
```

```
b=a.split() ← Διαχωρισμός λέξεων κειμένου
```

```
lexeis=len(b) ← Καταμέτρηση λέξεων που είναι αποθηκευμένες στη λίστα b
```

```
protaseis=0
```

Λυμένες ασκήσεις

Άσκηση 2

```
grammata=0
```

```
atim=0
```

```
for i in a:
```

```
    if i==".": ←
```

Έλεγχος για προτάσεις μέσω καταμέτρησης των τελειών

```
        protaseis=protaseis+1 ←
```

Καταμέτρηση προτάσεων

```
    if i=="a": ←
```

Έλεγχος εμφανίσεων του γράμματος "a"

```
        atim=atim+1 ←
```

Καταμέτρηση εμφανίσεων του "a"

```
for i in range(len(b)):
```

```
    grammata=grammata+len(b[i]) ←
```

Καταμέτρηση συνολικών χαρακτήρων μέσω μέτρησης των χαρακτήρων κάθε λέξης της λίστας b.

```
grammata=grammata-protaseis ←
```

Τα γράμματα υπολογίζονται ως οι χαρακτήρες της λίστας b μείον τις τελείες. Αν έχετε βάλει επιπλέον σημεία στίξης πρέπει να το λάβετε υπόψη.

```
print(lexeis/protaseis," λέξεις ανά πρόταση")
```

```
print(grammata/lexeis," γράμματα ανά λέξη")
```

```
print(atim," φορές υπάρχει το α στο κείμενο")
```

Λυμένες ασκήσεις

Άσκηση 6

Αρχικά θα δημιουργήσουμε δύο λίστες με αριθμητικές τιμές για τα δύο διανύσματα. Η πρόσθεση των δύο διανυσμάτων χρειάζεται δομή επανάληψης γιατί πρέπει να γίνει ξεχωριστά για κάθε στοιχείο που βρίσκεται στην ίδια θέση της κάθε λίστας. Επίσης πρέπει να ορίσουμε μια κενή λίστα για να αποθηκεύεται το κάθε αποτέλεσμα ώστε σε κάθε επανάληψη με χρήση της εντολής `.append()` να γίνεται η αποθήκευση. Οπότε ο κώδικας γράφεται ως εξής:

```
a=[1,2,4]
b=[1,3,7]
c=[]
for i in range(len(a)):
    c.append(a[i]+b[i])
```

Ορισμός διανυσμάτων

Δημιουργία κενής λίστας

Το αποτέλεσμα της πρόσθεσης καταχωρείται σε καινούριο στοιχείο της λίστας c

Λυμένες ασκήσεις

Άσκηση 7

Για την αφαίρεση στοιχείων από λίστα θα χρησιμοποιήσουμε την εντολή `.pop()`.
Οπότε ο κώδικας γράφεται ως εξής:

```
a=[1,2,4,5,7,8,9]
```

```
N=3
```

```
b=[]
```

```
for i in range(N):
```

```
    c=a.pop()
```

```
    b.append(c)
```

Αφαίρεση του τελευταίου στοιχείου της λίστας a και αποθήκευση στην μεταβλητή c

Καταχώρηση της τιμής του c στη λίστα b, άρα το τελευταίο στοιχείο του a τελικά εισάγεται στη λίστα b