

Σημειωματάριο Δευτέρας 23 Οκτ. 2017

Ένα μεγάλο πρόγραμμα διαχείρισης πληροφορίας σε λίστες

Σήμερα γράψαμε ένα μεγάλο πρόβλημα απλής διαχείρισης μιας μικρής βάσης δεδομένων. Η πληροφορία με την οποία ξεκινάμε περιέχεται ολόκληρη στις δύο λίστες L και M που φαίνονται στην αρχή του προγράμματος.

Κάθε στοιχείο της λίστας L περιέχει το όνομα ενός φοιτητή και μια λίστα με τα ονόματα των μαθημάτων που έχει πάρει.

Κάθε στοιχείο της λίστας H περιέχει το όνομα ενός μαθήματος και τη λίστα με τις ώρες της εβδομάδας στις οποίες διδάσκεται (μια ώρα της εβδομάδας αναπαρίσταται από μια λίστα με δύο στοιχεία, το όνομα της μέρας και τον αύξοντα αριθμό της ώρας, όπου 0 είναι η ώρα 9:00-10:00, 1 η ώρα 10:00 και, τέλος, 9 είναι η τελευταία ώρα 18:00-19:00).

Σκοπός μας στο πρόγραμμα αυτό είναι να εξάγουμε διάφορες πληροφορίες από τις λίστες αυτές.

Δείτε τα σχόλια μέσα στο πρόγραμμα για το τι ακριβώς κάνουμε.

```
In [8]: # Χρησιμοποιούμε τη συνάρτηση pprint από το module ("βιβλιοθήκη") pprint για να τυπώνου
με ωραία μια λίστα
from pprint import pprint

# Λίστα με τους φοιτητές και τα μαθήματά τους
L = [
    ["Γιάννης", ["ΑπΙ", "ΠρογρΙ", "ΑπΙΙΙ"]],
    ["Μαρία", ["ΑπΙ", "ΠρογρΙ", "Γραλ"]],
    ["Ελένη", ["ΑπΙ", "ΠρογρΙ", "Γραλ", "ΑναλΙ"]],
    ["Μανώλης", []],
    ["Παντελής", ["ΠρογρΙ", "ΑναλΙ", "ΑπΙΙΙ", "ΦυσΙ"]],
    ["Μιχάλης", ["ΑναλΙ"]]
]

# Λίστα με τα μαθήματα και τις ώρες που διδάσκονται μέσα στη βδομάδα
H = [
```

```

["ΑπΙ", [ ["Τρ", 0], ["Τρ", 1], ["Πε", 0], ["Πε", 1] ] ],
["ΠρογρΙ", [ ["Δε", 2], ["Δε", 3], ["Τε", 4], ["Τε", 5] ] ],
["ΑπΙΙΙ", [ ["Τε", 0], ["Τε", 1], ["Πα", 0], ["Πα", 1] ] ],
["ΑναλΙ", [ ["Τε", 4], ["Τε", 5], ["Πα", 4], ["Πα", 5] ] ],
["Γραλ", [ ["Δε", 3], ["Δε", 4], ["Πα", 3], ["Πα", 4] ] ],
["ΦυσΙ", [ ["Τρ", 4], ["Τρ", 5], ["Πα", 4], ["Πα", 5] ] ],
]

# Στη λίστα courses μαζεύουμε τα ονόματα όλων των μαθημάτων
courses = []
for l in L: # Για να τη συμπληρώσουμε διανύουμε τη λίστα L
    for c in l[1]: # Για κάθε ένα από τα μαθήματα που αναφέρονται σε κάποιο στοιχείο της L
        if c not in courses: # αν δεν το έχουμε ήδη βάλει στη λίστα μας
            courses.append(c) # το βάζουμε

print("Τα μαθήματα είναι:"); print()
pprint(courses)
print() # μια έξτρα γραμμή

# Στη λίστα students μαζεύουμε τα ονόματα όλων των φοιτητών
students = []
for l in L: # Διανύουμε τη λίστα L και μαζεύουμε τα πρώτα στοιχεία των στοιχείων της L
    students.append(l[0])

print("Οι φοιτητές είναι:"); print()
pprint(students)
print()

# Τώρα αντιστρέφουμε την πληροφορία που είναι στην L και την παρουσιάζουμε κατά μάθημα.
# Στη λίστα M θα έχουμε όλα τα μαθήματα με τους φοιτητές τους.
# Η πληροφορία στην M είναι οργανωμένη όπως στην L. Κάθε στοιχείο της M είναι μια λίστα με δύο
# στοιχεία. Το πρώτο στοιχείο είναι το όνομα του μαθήματος και το δεύτερο είναι η λίστα
# όλων των
# φοιτητών που παίρνουν το μάθημα. Δείτε την εκτύπωση της λίστας στο output του προγράμματος.
M = []
for c in courses: # για κάθε μάθημα c
    M.append([c, []]) # το προσθέτουμε στη λίστα M με κενή λίστα φοιτητών αρχικά.

```

```

for l in L: # για κάθε φοιτητή στη λίστα L
    name = l[0] # έστω name το όνομα του φοιτητή
    thecourses = l[1] # και έστω thecourses η λίστα των μαθημάτων του φοιτητή
    for c in thecourses: # για κάθε μάθημα c του φοιτητή που εξετάζουμε
        for m in M: # διανύουμε τη λίστα M για να βρούμε το στοιχείο της M που αφορά το
            μάθημα αυτό
                if m[0]==c: # αν το πρώτο στοιχείο του στοιχείου m της M είναι το c τότε βρ
                    ήκαμε το σωστό στοιχείο
                        m[1].append(name) # προσθέτουμε τον φοιτητή στη λίστα φοιτητών (δεύτερο
                            στοιχείο του m)
                                break # σταματάμε το "for m in M:" παραπάνω αφού δε χρειάζεται πλέον.

print("Η λίστα των μαθημάτων με τους φοιτητές τους είναι:"); print()
pprint(M); print()

# Για κάθε ζεύγος φοιτητών θέλουμε να βρούμε τη λίστα των κοινών τους μαθημάτων

# πρώτα βρίσκουμε όλα τα ζεύγη φοιτητών, στη λίστα pairs, η οποία θα περιέχει λίστες μή
    κους 2
pairs = []
for i in students: # Για κάθε φοιτητή i
    for j in students: # Για κάθε φοιτητή j
        if i == j: # Δε μας ενδιαφέρουν ζεύγη με ίδια στοιχεία
            continue # αυτό σημαίνει συνέχισε να πάρεις το επόμενο j
        if [j, i] not in pairs: # αν το ζεύγος που βρήκαμε, [i, j], δεν είναι ήδη στη λ
            ίστα με ανάποδη σειρά
                pairs.append([i, j]) # το βάζουμε στη λίστα pairs

# Στη λίστα cpp (courses per pair) θα αποθηκεύσουμε όλα τα ζεύγη φοιτητών με τα κοινά τ
    ους μαθήματα.
# Κάθε στοιχείο της cpp είναι μια λίστα με δύο στοιχεία. Το πρώτο στοιχείο είναι το ζεύ
    γος φοιτητών (μια
# λίστα δηλ. μήκους 2) και το δεύτερο στοιχείο είναι η λίστα των κοινών τους μαθημάτων.
    Δείτε το output.
cpp = []
for p in pairs: # Για κάθε ζεύγος (τα έχουμε ήδη βρει πριν)
    a = p[0]; b = p[1] # ονομάζουμε a τον πρώτο φοιτητή του ζεύγους και b το δεύτερο
        for l in L: # Για κάθε φοιτητή στη λίστα L
            if l[0]==a: # αν ο φοιτητής είναι ο a τότε στη μεταβλητή ca κρατάμε τη λίστα τω
                ν μαθημάτων του
                    ca = l[1]

```

```

        if l[0]==b: # αν ο φοιτητής είναι ο b τότε στη μεταβλητή cb κρατάμε τη λίστα των
        μαθημάτων του
            cb = l[1]
        cab = [] # έχουμε βρει τα μαθήματα του a (λίστα ca) και τα μαθήματα του b (λίστα c
        b)
        # και θέλουμε να βρούμε τα κοινά στοιχεία της λίστα ca και της cb στη λίστα cab.
        for h in ca: # για κάθε μάθημα του a
            if h in cb: # αν αυτό είναι και μάθημα του b
                cab.append(h) # τότε το προσθέτουμε στην cab (και μόνο τότε)
        crr.append([p, cab]) # έχουμε φτιάξει τη λίστα cab του ζεύγους p και το προσθέτουμε
        στη λίστα crr

print("Τα ζεύγη των φοιτητών με τα κοινά τους μαθήματα είναι:"); print()
pprint(crr); print()

# Από δω και κάτω θα βρούμε και θα τυπώσουμε το ωρολόγιο πρόγραμμα της εβδομάδας, το συ
# νολικό (σε ποιες
# ώρες της εβδομάδας γίνονται μαθήματα) και για κάθε φοιτητή χωριστά.
# Αν σε μια ώρα της εβδομάδας γίνονται μαθήματα τότε τυπώνουμε σε εκείνη τη θέση το str
ing '**' αλλιώς το ' '
day = ["Δε", "Τρ", "Τε", "Πε", "Πα"] # Οι εργάσιμες μέρες της εβδομάδας

print("        Συνολικό ωρολόγιο πρόγραμμα")
s = "    Δε Τρ Τε Πε Πα" # Η πρώτη γραμμή που θα τυπωθεί
print(s)
# θα τυπωθούν ακόμη 10 γραμμές, όσες και οι ώρες κάθε μέρας
# Σε κάθε γραμμή θα υπάρχουν 5 ώρες, μια για κάθε εργάσιμη μέρα. Δείτε το output.
for i in range(10): # Για κάθε ώρα της μέρας (για κάθε γραμμή που θέλουμε να τυπώσουμε)
    line = "{x:02d}".format(x=i+9)+" " # Στην αρχή της γραμμής τυπώνουμε την ώρα που αν
    τιστοιχεί (αρχίζει από i+9)
    # Φροντίζουμε ακόμη και το 9 να τυπωθεί διψήφιο για να είναι σωστή η στοίχιση.
    # Αυτός είναι ο ρόλος του x:02d μέσα στα άγγιστρα. Το x είναι εδώ το όνομα της "τρύ
    πας" με το οποίο αναφερόμαστε
    # σε αυτή στο format. Το 02d σημαίνει ότι τυπώνουμε ακέραιο, σε 2 το πολύ θέσεις, μ
    ε "leading zeros".
    for d in range(5): # Για κάθε μια από τις 5 μέρες
        sp="    " # αυτό θα τυπωθεί στη θέση της μέρας στη γραμμή αυτή. Αν δε βρούμε μάθ
        ημα εκεί τότε θα τυπωθούν κενά
        for h in H: # Για κάθε μάθημα στον πίνακα H
            if [day[d], i] in h[1]: # Αν η μέρα και ώρα που κοιτάμε είναι στις ώρες του
            μαθήματος
                sp="** " # τότε ξέρουμε ότι διδάσκεται μάθημα αυτή την ώρα και τυπώνουμ

```

```

ε αστεράκια
    break # δε χρειάζεται να ψάξουμε αν γίνονται άλλα μαθήματα αφού ήδη έχο
υμε βρει ένα.
    line += sp # προσθέτουμε το string που βρήκαμε στη γραμμή line που θα τυπώσουμε
    print(line) # τυπώνουμε τη γραμμή που φτιάξαμε (που αντιστοιχεί σε μια από τις 10 ώ
ρες της μέρας)

print()

# Για κάθε φοιτητή θέλουμε τώρα να τυπώσουμε το ωρολόγιό του πρόγραμμα

for st in students: # Για κάθε φοιτητή st
    stcourses = [] # βρίσκουμε πρώτα από τη λίστα L τη λίστα με τα μαθήματά του
    for l in L:
        if l[0]==st:
            stcourses=l[1]
            break

    print("-----Ωρολόγιο πρόγραμμα του/ης {name}".format(name=st))
    print()
    s = "        Δε Τρ Τε Πε Πα" # Πρώτη γραμμή, επικεφαλίδα, λίγο πιο μέσα τώρα
    print(s) # τυπώνουμε την επικεφαλίδα
    for i in range(10): # Για κάθε ώρα της μέρας
        line = "{x:02d} - {y:02d}".format(x=i+9, y=i+10)+" " # τώρα τυπώνουμε αρχή και
τέλος της ώρας
        for d in range(5): # για κάθε μέρα της εβδομάδας
            sp=" " # ψάχνουμε όπως και πριν να βρούμε τι θα τυπώσουμε στη θέση αυτή,
κενά ή αστεράκια
            for h in H:
                if h[0] not in stcourses:
                    continue
                if [day[d], i] in h[1]:
                    sp="** "
                    break
            line += sp
    print(line)

```

Τα μαθήματα είναι:

['ΑπΙ', 'ΠρογρΙ', 'ΑπΙΙΙ', 'ΓρΑλ', 'ΑναλΙ', 'ΦυσΙ']

Οι φοιτητές είναι:

['Γιάννης', 'Μαρία', 'Ελένη', 'Μανώλης', 'Παντελής', 'Μιχάλης']

Η λίστα των μαθημάτων με τους φοιτητές τους είναι:

```
[[['ΑπΙ', ['Γιάννης', 'Μαρία', 'Ελένη']],
  ['ΠρογρΙ', ['Γιάννης', 'Μαρία', 'Ελένη', 'Παντελής']],
  ['ΑπΙΙΙ', ['Γιάννης', 'Παντελής']],
  ['ΓρΑλ', ['Μαρία', 'Ελένη']],
  ['ΑναλΙ', ['Ελένη', 'Παντελής', 'Μιχάλης']],
  ['ΦυσΙ', ['Παντελής']]]
```

Τα ζεύγη των φοιτητών με τα κοινά τους μαθήματα είναι:

```
[[['Γιάννης', 'Μαρία'], ['ΑπΙ', 'ΠρογρΙ']],
  [['Γιάννης', 'Ελένη'], ['ΑπΙ', 'ΠρογρΙ']],
  [['Γιάννης', 'Μανώλης'], []],
  [['Γιάννης', 'Παντελής'], ['ΠρογρΙ', 'ΑπΙΙΙ']],
  [['Γιάννης', 'Μιχάλης'], []],
  [['Μαρία', 'Ελένη'], ['ΑπΙ', 'ΠρογρΙ', 'ΓρΑλ']],
  [['Μαρία', 'Μανώλης'], []],
  [['Μαρία', 'Παντελής'], ['ΠρογρΙ']],
  [['Μαρία', 'Μιχάλης'], []],
  [['Ελένη', 'Μανώλης'], []],
  [['Ελένη', 'Παντελής'], ['ΠρογρΙ', 'ΑναλΙ']],
  [['Ελένη', 'Μιχάλης'], ['ΑναλΙ']],
  [['Μανώλης', 'Παντελής'], []],
  [['Μανώλης', 'Μιχάλης'], []],
  [['Παντελής', 'Μιχάλης'], ['ΑναλΙ']]]
```

Συνολικό ωρολόγιο πρόγραμμα

	Δε	Τρ	Τε	Πε	Πα
09	**	**	**	**	
10		**	**	**	**
11	**				
12	**			**	

13 ** ** ** ** ** ** ** **
14 ** ** **
15
16
17
18

-----Ωρολόγιο πρόγραμμα του/ης Γιάννης

	Δε	Τρ	Τε	Πε	Πα
09 - 10		**	**	**	**
10 - 11		**	**	**	**
11 - 12	**				
12 - 13	**				
13 - 14			**		
14 - 15			**		
15 - 16					
16 - 17					
17 - 18					
18 - 19					

-----Ωρολόγιο πρόγραμμα του/ης Μαρία

	Δε	Τρ	Τε	Πε	Πα
09 - 10		**		**	
10 - 11		**		**	
11 - 12	**				
12 - 13	**				**
13 - 14	**		**		**
14 - 15			**		
15 - 16					
16 - 17					
17 - 18					
18 - 19					

-----Ωρολόγιο πρόγραμμα του/ης Ελένη

	Δε	Τρ	Τε	Πε	Πα
09 - 10		**		**	
10 - 11		**		**	
11 - 12	**				
12 - 13	**				**
13 - 14	**		**		**
14 - 15			**		**

15 - 16
16 - 17
17 - 18
18 - 19

-----Ωρολόγιο πρόγραμμα του/ης Μανώλης

Δε Τρ Τε Πε Πα

09 - 10
10 - 11
11 - 12
12 - 13
13 - 14
14 - 15
15 - 16
16 - 17
17 - 18
18 - 19

-----Ωρολόγιο πρόγραμμα του/ης Παντελής

Δε Τρ Τε Πε Πα

09 - 10 ** **
10 - 11 ** **
11 - 12 **
12 - 13 **
13 - 14 ** ** **
14 - 15 ** ** **
15 - 16
16 - 17
17 - 18
18 - 19

-----Ωρολόγιο πρόγραμμα του/ης Μιχάλης

Δε Τρ Τε Πε Πα

09 - 10
10 - 11
11 - 12
12 - 13
13 - 14 ** **
14 - 15 ** **
15 - 16
16 - 17

17 - 18

18 - 19