

## Σημειωματάριο Τετάρτης 11 Οκτ. 2017

### Παραδείγματα χρήσης του `if` και χρήσης λιστών ¶

Το πρώτο πρόγραμμα που γράψαμε σήμερα παίρνει ως είσοδο τρεις πραγματικούς αριθμούς και τους τυπώνει με φθίνουσα σειρά.

Ο τρόπος που το πετυχαίνουμε αυτό είναι ότι πραγματοποιούμε μερικές "συγκρίσεις ζευγών". Μια σύγκριση ζεύγους συνίσταται στο να κοιτάξουμε δύο από τους αριθμούς μας (τους  $x$ ,  $y$ ,  $z$  δηλ. που θέλουμε στο τέλος να ικανοποιούν τις ανισότητες  $x \geq y \geq z$ ) και αν οι δύο αυτοί αριθμοί δεν είναι στη σωστή σειρά τότε εναλλάσσουμε τα περιεχόμενα των δύο αντίστοιχων μεταβλητών.

Π.χ. κοιτάμε στο πρώτο `if` του προγράμματος αν τα  $x$ ,  $y$  είναι στη σωστή σειρά, αν ισχύει δηλ.  $x \geq y$ . Αν δεν ισχύει το επιθυμητό, αν ισχύει δηλ.  $x < y$ , τότε εναλλάσσουμε τα περιεχόμενα των δύο μεταβλητών  $x$  και  $y$ . Αν αυτό το κάνουμε για τα ζεύγη  $x$ ,  $y$  και  $x$ ,  $z$  τότε ξέρουμε ότι μετά από τις δύο αυτές συγκρίσεις ζευγών η μεταβλητή  $x$  θα έχει το σωστό περιεχόμενο, θα περιέχει δηλ. τη μέγιστη τιμή από τις τρεις μεταβλητές. Αν, μετά από αυτό, κάνουμε ακόμη τη σύγκριση του ζεύγους  $y$ ,  $z$  τότε μετά το τέλος αυτής της πράξης και οι τρεις μεταβλητές θα έχουν τα σωστά τους περιεχόμενα.

Τα τρία `if` του προγράμματος κάνουν λοιπόν ακριβώς αυτές τις τρεις συγκρίσεις ζευγών.

Η βοηθητική μεταβλητή `t` χρησιμοποιείται κάθε φορά για να εναλλάξουμε τις τιμές δύο μεταβλητών. Π.χ. για να εναλλάξουμε τις τιμές των  $x$ ,  $y$  δίνουμε τις τρεις εκχωρήσεις:

```
t = y; y = x; x = t
```

Στην `python` είναι δυνατόν οι δύο μεταβλητές  $x$  και  $y$  να εναλλαγούν και πιο απλά από τη παραπάνω, με την εντολή

```
x, y = y, x
```

In [1]:

```
x = float(input("Δώσε αριθμό x: "))
y = float(input("Δώσε αριθμό y: "))
z = float(input("Δώσε αριθμό z: "))

# Κάνουμε τρεις ελέγχους ζευγών. Μετά από αυτές οι τρεις μεταβλητές
# x, y, z είναι σε φθίνουσα σειρά
if y>x:
    t = y; y = x; x = t
if z>x:
    t = z; z = x; x = t
if z>y:
    t = z; z = y; y = t

print("{} {} {}".format(x, y, z))
```

```
Δώσε αριθμό x: 3
Δώσε αριθμό y: 4
Δώσε αριθμό z: 1
4.0 3.0 1.0
```

Το επόμενο μας πρόγραμμα υπολογίζει το φόρο που αντιστοιχεί σε εισόδημα  $x$ .

Οι κανόνες είναι οι εξής: Αν το εισόδημα είναι  $x \geq 0$  και  $T(x)$  είναι ο φόρος που του αντιστοιχεί τότε η  $T(x)$  υπολογίζεται ως εξής:

Θεωρούμε το διάστημα  $[0, x]$  στην πραγματική ευθεία και τα διαχωριστικά σημεία  $x_0 = 5000$ ,  $x_1 = 10000$ ,  $x_2 = 20000$ . Το διάστημα  $[0, x]$  καλύπτει ένα κομμάτι σε κάθε ένα από τα διαστήματα  $(0, x_0]$ ,  $(x_0, x_1]$ ,  $(x_1, x_2]$ ,  $(x_2, +\infty)$ .

Το κομμάτι που καλύπτει στο πρώτο διάστημα πολλαπλασιάζεται με 0 (αφορολόγητο), το κομμάτι στο διάστημα  $(x_0, x_1]$  πολλαπλασιάζεται με 0.2 ("φόρος κλιμακίου 20%"), το κομμάτι στο διάστημα  $(x_1, x_2]$  πολλαπλασιάζεται με 0.3 και, τέλος, το κομμάτι στο διάστημα  $(x_2, +\infty)$  πολλαπλασιάζεται με 0.5. Αυτά αθροίζονται και έτσι προκύπτει ο συνολικός φόρος  $T(x)$ .

In [3]:

```
x = float(input("Ποιο είναι το εισόδημα; "))

if x <= 5e3: # Όλο το ποσό είναι αφορολόγητο
    T = 0

if 5e3 <= x and x <= 1e4: # Το ποσό καλύπτει και μέρος του διαστήματος (5000, 10000) αλλά όχι παραπέρα
    T = (x-5e3)*0.2

if 1e4 <= x and x <= 2e4: # Το ποσό καλύπτει και μέρος του διαστήματος (10000, 20000) αλλά όχι παραπέρα
    T = 5e3*0.2 + (x-1e4)*0.3

if x > 2e4: # Το ποσό είναι πέρα από 20000
    T = 5e3*0.2 + 1e4*0.3 + (x-2e4)*0.5

print("Ο φόρος είναι {}".format(T))
```

Ποιο είναι το εισόδημα; 25000  
Ο φόρος είναι 6500.0

Εδώ ξαναγράφουμε το πρόγραμμα με χρήση του `if ... elif`. Έτσι γίνεται απλούστερο και πιο κατανοητό.

In [4]:

```
x = float(input("Ποιο είναι το εισόδημα; "))

if x <= 5e3:
    T = 0
elif x <= 1e4:
    T = (x-5e3)*0.2
elif x <= 2e4:
    T = 5e3*0.2 + (x-1e4)*0.3
else:
    T = 5e3*0.2 + 1e4*0.3 + (x-2e4)*0.5

print("Ο φόρος είναι {}".format(T))
```

Ποιο είναι το εισόδημα; 15000  
Ο φόρος είναι 2500.0

Το επόμενο μας πρόγραμμα διαβάζει μια ημερομηνία από το χρήστη (ημέρα, μήνας, έτος) χωρισμένα από κόμμα και τυπώνει την ημερομηνία της επόμενης ημέρας. Φυσικά η δυσκολία είναι στις περιπτώσεις που αλλάζει ο μήνας ή και ο χρόνος.

Παρατηρείστε ότι στην πρώτη γραμμή του προγράμματος διαβάζουμε και τους τρεις αριθμούς ταυτόχρονα, ως μια τριάδα (tuple). Ο χρήστης πρέπει να τα δώσει στην ίδια γραμμή χωρισμένα με κόμμα.

In [17]:

```

day, month, year = eval(input("Δώστε την ημερομηνία στη μορφή ημέρα, μήνας, χρόν
ος: "))

# Ελέγχουμε αν έτος δίσεκτο. Ένα έτος είναι δίσεκτο αν
# (διαρείται από το 400) ή (διαρείται από το 4 αλλά όχι από το 100)
if (year % 400 == 0) or ( (year % 4 == 0) and (year % 100 != 0)):
    leap = True
else:
    leap = False # Στη μεταβλητή leap αποθηκεύουμε τον υπολογισμό μας. Είναι Tru
e αν και μόνο αν το έτος είναι δίσεκτο.

# Υπολογίζουμε την τελευταία μέρα του μήνα στη μεταβλητή lastday
if month in [4, 6, 9, 11]: # Οι μήνες με τριάντα μέρες είναι αυτοί που είναι μέσ
α στη λίστα.
    # Ελέγχουμε αν ο μήνας είναι μέσα σε αυτή τη λίστα
    lastday = 30
elif month == 2: # Αν ο μήνας είναι Φεβρουάριος
    if leap: # αν το έτος είναι δίσεκτο τότε η τελευταία μέρα είναι η 29η
        lastday = 29
    else: # αλλιώς είναι η 28η
        lastday = 28
else: # Αν ο μήνας δεν είναι 30άρης ούτε ο Φεβρουάριος τότε έχει 31 μέρες
    lastday = 31

if day < lastday: # Αν η μέρα δεν είναι η τελευταία του μήνα όλα είναι απλά
    nextday = day+1
    nextmonth = month
    nextyear = year
else: # Αν η μέρα είναι η τελευταία του μήνα
    nextday = 1 # η επόμενη μέρα είναι πάντα η πρώτη κάποιου μήνα
    if month==12: # αν ο μήνας είναι ο 12ος
        nextmonth = 1 # τότε ο νας της επόμενης μέρας είναι ο Ιανουάριος
        nextyear = year+1 # και η χρονιά της επόμενης μέρας είναι η επόμενη χρον
ιά
    else: # αν ο μήνας δεν είναι ο 12ος
        nextmonth = month+1 # ο μήνας της επόμενης μέρας είναι ο επόμενος αριθμη
τικά μήνας
        nextyear = year # και το έτος της επόμενης μέρας είναι το ίδιο

print("Η επόμενη ημερομηνία είναι η {}-{}-{}".format(nextday, nextmonth, nextyea
r))

```

Δώστε την ημερομηνία: 31,12,2017

Η επόμενη ημερομηνία είναι η 1-1-2018

Τώρα τροποποιούμε το πρόγραμμά μας ώστε να τυπώνουμε το μήνα με το όνομά του και όχι με τον αριθμό του.

Για να το πετύχουμε αυτό ορίζουμε μια λίστα name με όλα τα ονόματα μηνών στη σειρά (τη γενική του ονόματος). Αντί να τυπώνουμε τον αριθμό του μήνα nextmonth τυπώνουμε το όνομα που του αντιστοιχεί και το οποίο είναι το name[nextmonth-1]. Ο λόγος για το -1 είναι ότι η αρίθμηση των μηνών αρχίζει από το 1 ενώ η αρίθμηση των στοιχείων μιας λίστας (όπως και αυτή των γραμμάτων ενός string) αρχίζει από το 0.

In [22]:

```

day, month, year = eval(input("Δώστε την ημερομηνία: "))

# Ελέγχουμε αν έτος δίσεκτο
if (year % 400 == 0) or ( (year % 4 == 0) and (year % 100 != 0) ):
    leap = True
else:
    leap = False

# Υπολογίζουμε την τελευταία μέρα του μήνα
if month in [4, 6, 9, 11]:
    lastday = 30
elif month == 2:
    if leap:
        lastday = 29
    else:
        lastday = 28
else:
    lastday = 31

if day < lastday:
    nextday = day+1
    nextmonth = month
    nextyear = year
else:
    nextday = 1
    if month==12:
        nextmonth = 1
        nextyear = year+1
    else:
        nextmonth = month+1
        nextyear = year

name = ["Ιανουαρίου", "Φεβρουαρίου", "Μαρτίου", "Απριλίου", "Μαΐου", "Ιουνίου",
        "Ιουλίου", "Αυγούστου", "Σεπτεμβρίου", "Οκτωβρίου", "Νοεμβρίου", "Δεκεμβρίου"]
print("Η επόμενη ημερομηνία είναι η {}-{}-{}".format(nextday, name[nextmonth-1],
        nextyear))

```

Δώστε την ημερομηνία: 3,4,2020

Η επόμενη ημερομηνία είναι η 4-Απριλίου-2020

