

Ασκήσεις εργαστηρίου

Υπενθύμιση θεωρίας

Αναπαράσταση αριθμητικών τιμών

Εάν χρειάζεται να γράψουμε τιμές δυνάμεων του 10 π.χ. $7 \cdot 10^6$, αντί να γράψουμε $7 \cdot 10^{**6}$ μπορούμε να γράψουμε πιο σύντομα $7e6$. Παρόμοια για αρνητικές δυνάμεις γράφουμε π.χ. για το $7 \cdot 10^{-6}$, $7e-6$.

Είσοδος τιμών από το χρήστη

Εάν επιθυμούμε να εισάγει ο χρήστης δεδομένα χρησιμοποιούμε την εντολή `input`. Στην πιο απλή μορφή της γράφεται ως εξής:

`N = input()` όπου N είναι η τιμή που θα εισαχθεί

Επειδή όμως η Python αντιλαμβάνεται την είσοδο ως κείμενο, τότε αν εισάγουμε αριθμητική τιμή πρέπει να μετατρέψουμε την είσοδο π.χ. σε ακέραια ή πραγματική τιμή με τις συναρτήσεις `int` ή `float` αντίστοιχα:

`N = int(input())` ή `float(input())`.

Επίσης μπορούμε να εμφανίζουμε και ένα μήνυμα πριν την είσοδο των τιμών π.χ.

`N = float(input("Δώστε την τιμή:"))`

Ασκήσεις εργαστηρίου

Άσκηση 1

Γράψτε σε Python κώδικα που να υπολογίζει την τιμή του f ανάλογα με τις διάφορες τιμές του x ως εξής:

$$f = \begin{cases} x^3 + 5x + 3 & \text{για } x \leq 0 \\ \frac{5x + 1}{\sqrt{3x^2 + 2}} & \text{για } 0 < x \leq 100 \\ (3x + 1)^{-2x} & \text{για } x > 100 \end{cases}$$

Άσκηση 2

Γράψτε σε Python κώδικα που να υπολογίζει για ένα υλικό την πυκνότητα ρ και τη θερμική του αγωγιμότητα k για συγκεκριμένη θερμοκρασία T . Η πυκνότητα για διάφορες θερμοκρασίες δίνεται από τις σχέσεις:

$$-1.295 \cdot 10^{-6}T^4 + 7.459 \cdot 10^{-4}T^3 - 0.16537 \cdot T^2 + 15.12763 \cdot T - 2.45869, \quad 121 < T \leq 223$$

$$-5.310 \cdot 10^{-4}T^3 + 0.433 \cdot T^2 - 119.033 \cdot T + 11091.67, \quad 223 < T \leq 271$$

$$1.329 \cdot 10^{-7}T^4 - 1.935 \cdot 10^{-4}T^3 + 0.107 \cdot T^2 + 26.702 \cdot T + 2634.06, \quad 271 < T \leq 385$$

Ασκήσεις εργαστηρίου

Ενώ η θερμική αγωγιμότητα δίνεται από τις σχέσεις:

$$1.0704 \cdot 10^{-8}T^3 - 4.353 \cdot 10^{-6}T^2 - 6.44610^{-4} \cdot T + 0.307, 12 < T \leq 235$$

$$7.969 \cdot 10^{-6}T^2 - 0.004 \cdot T + 0.629, 235 < T \leq 262$$

$$-6.694 \cdot 10^{-9}T^3 + 7.368 \cdot 10^{-6}T^2 - 0.003 \cdot 10^{-2}T + 0.334, 262 < T \leq 385$$

Άσκηση 3

Γράψτε σε Python κώδικα ο οποίος με βάση τιμές που δίνει ο χρήστης υπολογίζει πόσες τιμές ήταν θετικές, πόσες αρνητικές και πόσες μηδέν. Η εισαγωγή δεδομένων θα τελειώνει αν ο χρήστης δίνει την τιμή -1000.

Άσκηση 4

Γράψτε σε Python κώδικα που θα ελέγχει αριθμούς από 1 έως N και θα υπολογίζει πόσοι αριθμοί είναι ζυγοί, πόσοι πολ/σια του 3 και πόσοι πολ/σια του 5.

Άσκηση 5

Γράψτε σε Python κώδικα που θα κάνει παραγοντοποίηση ενός αριθμού σε πρώτους παράγοντες.

Ασκήσεις εργαστηρίου

Άσκηση 6

Γράψτε σε Python κώδικα που θα ελέγχει αν ένας αριθμός είναι θετικός ακέραιος, και εάν είναι ζυγός θα τον διαιρεί με το 2 ενώ αν είναι περιττός θα πολλαπλασιάζει με το 3 και θα προσθέτει 1 έως ότου ο αριθμός γίνει 1. Δοκιμάστε το ίδιο εάν στην περίπτωση του περιττού αριθμού πολλαπλασιάζουμε με 5 και προσθέτουμε 1.

Άσκηση 7

Υπολογίστε προσεγγιστικά την τιμή της τετραγωνικής ρίζας ενός αριθμού n με βάση τη σχέση:

$$x_{n+1} = \frac{1}{2} \left(x_n + \frac{N}{x_n} \right)$$

Να συγκρίνετε τη λύση με την πράξη $N^{**}(1/2)$.

Άσκηση 8

Γράψτε κώδικα σε Python ο οποίος να ξεχωρίζει τα ψηφία ενός αριθμού N π.χ. 1560-1,5,6,0 και να υπολογίζει το άθροισμα και το γινόμενο τους.

Λυμένες ασκήσεις

Άσκηση 1

Για να υπολογιστεί η ζητούμενη τιμή χρειάζεται μια δομή επιλογής γιατί η σχέση υπολογισμού αλλάζει ανάλογα με τις τιμές του x . Επειδή οι επιλογές είναι τρεις και καλύπτουν όλες τις περιπτώσεις μπορούμε να χρησιμοποιήσουμε είτε `if, elif, elif` είτε `if, elif, else`.

Για τη δεύτερη περίπτωση χρειάζεται προσοχή γιατί το x πρέπει να είναι ταυτόχρονα μεγαλύτερο από το μηδέν και μικρότερο ή ίσο με το 100. Επίσης, το x πρέπει να οριστεί πριν τη δομή επιλογής για να μπορεί να γίνει στη συνέχεια ο έλεγχος.

Οπότε ο κώδικας γράφεται:

```
x = 50
```

```
if (x<=0):
```

```
    f = x**3+5*x+3
```

```
    print("Η τιμή του f είναι:",f)
```

```
elif (x>0) and (x<=100):
```

```
    f = (5*x+1)/((3*x**2+2)**(1/2))
```

```
    print("Η τιμή του f είναι:",f)
```

```
else:
```

```
    f = (3*x+1)**(-2*x)
```

```
    print("Η τιμή του f είναι:",f)
```

Λυμένες ασκήσεις

Άσκηση 2

Για να υπολογιστούν οι ζητούμενες τιμές των ρ και k χρειάζεται πάλι μια δομή επιλογής γιατί η σχέση υπολογισμού αλλάζει ανάλογα με τις τιμές του T . Οπότε η άσκηση μπορεί να επιλυθεί όπως η προηγούμενη με τη διαφορά ότι χρειάζεται μία επιπλέον επιλογή για την περίπτωση που η θερμοκρασία είναι κάτω από τους 121 ή πάνω από τους 385 βαθμούς. Επίσης χρειάζεται προσοχή στις πράξεις και να οριστεί το T πριν τη δομή επιλογής. Οπότε ο κώδικας γράφεται:

```
T = 180
```

```
if (T>121) and (T<=223):
```

```
    r = -1.295e-6*T**4+7.459e-4*T**3-0.16537*T**2+15.12763*T-2.45869
```

```
    print("Η τιμή της πυκνότητας είναι:",r)
```

```
elif (T>223) and (T<=271):
```

```
    r = -5.310e-4*T**3+0.433*T**2-119.033*T+11091.67
```

```
    print("Η τιμή της πυκνότητας είναι:",r)
```

```
elif (T>271) and (T<=385):
```

```
    r = 1.329e-7*T**4-1.935e-4*T**3+0.107*T**2+26.702*T-2634.06
```

```
    print("Η τιμή της πυκνότητας είναι:",r)
```

```
else:
```

```
    print("Η τιμή της θερμοκρασίας είναι εκτός των ορίων")
```

Λυμένες ασκήσεις

Άσκηση 3

Η εισαγωγή των τιμών και οι υπολογισμοί είναι επαναλαμβανόμενοι άρα θα χρησιμοποιηθεί κάποια δομή επανάληψης. Επειδή δεν προσδιορίζεται ο αριθμός των επαναλήψεων εκτός από ένα κριτήριο (η τιμή -1000) καλύτερο είναι να χρησιμοποιηθεί η δομή επανάληψης while.

Επειδή χρειάζεται να ελεγχθούν οι τιμές αν είναι θετικές, αρνητικές ή μηδέν χρειάζεται και μια δομή επιλογής μέσα στο while καθώς και μεταβλητές στις οποίες θα αποθηκεύεται ο αριθμός των θετικών, αρνητικών και μηδενικών τιμών. Τέλος επειδή η τελική τιμή είναι πάντα αρνητική (-1000) πρέπει να αφαιρεθεί 1 από τη μέτρηση των αρνητικών τιμών για να μην επηρεαστεί ο υπολογισμός.

Οπότε ο κώδικας γράφεται:

```
pos=0
neg=0
zero=0
n=0
while (n!=-1000):
    n = float(input())
    if (n>0):
        pos=pos+1
    elif (n<0):
```

Αρχικοποίηση μεταβλητών για μετρήσεις

Δίνεται μια αρχική τιμή στο n (που δεν θα επηρεάσει το πρόγραμμα) για να ξεκινήσει η επανάληψη

← Το n δίνεται από το χρήστη

← Στην Python μπορεί να γραφεί και ως: pos+=1

Λυμένες ασκήσεις

Άσκηση 3

```
        neg=neg+1
else:
        zero=zero+1
```

```
print("Οι θετικοί αριθμοί ήταν:",pos)
print("Οι αρνητικοί αριθμοί ήταν:",neg-1)
print("Τα μηδενικά ήταν:",zero)
```



Αφαιρείται 1 από την τιμή γιατί πάντα υπολογίζεται και το -1000 που είναι η τελευταία τιμή.

Άσκηση 4

Η άσκηση αυτή απαιτεί εισαγωγή στοιχείων και πολλαπλούς ελέγχους. Άρα χρειάζεται κάποια δομή επανάληψης, η οποία μπορεί να είναι η for γιατί το N είναι γνωστή τιμή. Για τους ελέγχους θα πρέπει να υπάρχει κάποια δομή επιλογής μέσα στο for στην οποία θα ελέγχεται ο κάθε αριθμός αν είναι ζυγός δηλαδή αν το υπόλοιπο της διαίρεσης με το 2 είναι μηδέν, αν είναι πολ/σιο του 3 (με εξαίρεση όσα είναι ζυγά), δηλαδή αν το υπόλοιπο της διαίρεσης με το 3 είναι μηδέν ή αν είναι πολ/σιο του 5 (με εξαίρεση όσα είναι και πολ/σια του 3), δηλαδή αν το υπόλοιπο της διαίρεσης με το 5 είναι μηδέν. Για να υπολογίσει πόσοι αριθμοί ανήκουν σε κάθε περίπτωση πρέπει να οριστούν από την αρχή μεταβλητές στις οποίες θα αποθηκεύονται οι ζητούμενοι αριθμοί.

Λυμένες ασκήσεις

Άσκηση 4

Οπότε ο κώδικας γράφεται:

```
even=0
mult3=0
mult5=0
N=18
for i in range(1,N+1):
    if (i%2==0):
        even=even+1
    elif (i%3==0):
        mult3=mult3+1
    elif (i%5==0):
        mult5=mult5+1
print("Ο αριθμός των ζυγών είναι:", even)
print("Ο αριθμός των πολ/σίων του 3 είναι:", mult3)
print("Ο αριθμός των πολ/σίων του 5 είναι:", mult5)
```

Αρχικοποίηση μεταβλητών για μετρήσεις

Ορισμός της τιμής του N

Δε χρειάζεται else γιατί δε μας ενδιαφέρουν οι υπόλοιπες περιπτώσεις