

Διάβασμα τιμών (Μέρος 1^ο)

Όταν μια άσκηση μας ζητάει να διαβάσουμε κάποια τιμή εννοεί ότι πρέπει να κάνουμε εισαγωγή τιμών από το πληκτρολόγιο. Αυτό γίνεται με τις εντολές `input` και `raw_input`. Χρησιμοποιούμε την `raw_input` για να διαβάσουμε αλφαριθμητικά δεδομένα (`str`) και την `input` για όλα τα υπόλοιπα. Είναι αρκετά χρήσιμο να χρησιμοποιούμε μαζί με την `input` και τις ενσωματωμένες συναρτήσεις μετατροπής τύπων δεδομένων, ανάλογα με τον τύπο των δεδομένων που εισάγουμε (`int` για ακέραιους, `float` για πραγματικούς).

```
Μεταβλητή = raw_input("Δώσε κείμενο:")  
Μεταβλητή = int(input("Δώσε ακέραιο:"))  
Μεταβλητή = float(input("Δώσε πραγματικό:"))
```

Αρχικοποίηση

Από τη στιγμή που μια μεταβλητή χρησιμοποιείται ΔΕΞΙΑ από το `=` πρέπει να έχει πάρει τιμή κάποια στιγμή νωρίτερα. Αν δεν παίρνει τιμή από `input` ή δεν μας δίνεται από την άσκηση, θα πρέπει να δίνουμε εμείς μια αρχική τιμή (συνήθως 0 για τα αριθμητικά δεδομένα, `" "` για τα αλφαριθμητικά, `[]` για τις λίστες κτλ). Η διαδικασία αυτή λέγεται αρχικοποίηση.

Διαίρεση

Όταν πρέπει να κάνουμε οποιαδήποτε διαίρεση σε κάποιο πρόγραμμά μας πρέπει να κάνουμε ΠΑΝΤΑ 2 ενέργειες.

1. Να φροντίζουμε ο ένας από τους 2 αριθμούς να είναι `float`. Αν δεν είναι θα πρέπει να τον μετατρέπουμε με τη συνάρτηση `float()` ή προσθέτοντας το μηδέν ως δεκαδικό ψηφίο σε κάποιον από τους 2 αριθμούς.
2. Να φροντίζουμε ώστε ο παρονομαστής να μην είναι μηδέν. Αν δεν είμαστε σίγουροι θα πρέπει να ελέγχουμε την αντίστοιχη μεταβλητή με την εντολή `if`.

Αναπαράσταση ποσοστών

Στον προγραμματισμό τα ποσοστά να αναπαρίστανται ως μαθηματικές πράξεις, οπότε το 20% αναγράφεται ως 20/100. Επειδή όμως η πράξη αυτή είναι διαίρεση, ο σωστός τρόπος για να αναγράφουμε τα ποσοστά είναι είτε με τη δεκαδική τους μορφή πχ 0.2 είτε ως διαίρεση με δεκαδικό πχ `float(20)/100` ή 20.0/100.

Προσοχή: 70% = 0.7 , 7% = 0.07 , 64% = 0.64 , 6.4% = 0.064 , 23% = 0.23

Υπολογισμός ποσοστού

Για να βρούμε το ποσοστό εφαρμόζουμε την πράξη:

Ποσοστό = Μέρος του συνόλου / Σύνολο * 100

Αν ένα από τα 2 μέρη δεν είναι δεκαδικός, λόγω διαίρεσης ο τύπος πρέπει να γίνει:

```
Ποσοστό = ( float(Μέρος του συνόλου) ) / Σύνολο * 100
```

Υπολογισμός αύξησης

Για να βρούμε πόσο θα αυξηθεί μια τιμή βάσει ενός ποσοστού, ακολουθούμε μια διαδικασία με δυο βήματα.

1. Βρίσκουμε πόσο αυξάνεται η αρχική τιμή που έχουμε, δηλαδή την αύξηση.
2. Βρίσκουμε την τελική τιμή, όπως διαμορφώνεται μετά την αύξηση.

Η αύξηση υπολογίζεται από τον τύπο:

$$\text{Αύξηση} = \text{Αρχική τιμή} * \text{Ποσοστό}$$

Η τελική τιμή που προκύπτει είναι:

$$\text{Τελική τιμή} = \text{Αρχική τιμή} + \text{Αύξηση}$$

Τα 2 βήματα μπορούν να γίνουν και ένα αν δεν μας ζητείται η αύξηση σαν ξεχωριστό ερώτημα:

$$\text{Τελική τιμή} = \text{Αρχική τιμή} + (\text{Αρχική τιμή} * \text{Ποσοστό})$$

Προσοχή στον τρόπο αναπαράστασης του ποσοστού (δεκαδικό).

Υπολογισμός μείωσης

Ισχύουν τα ίδια βήματα με την αύξηση, με τη διαφορά ότι η μείωση αφαιρείται από την αρχική τιμή.

Η μείωση υπολογίζεται από τον τύπο:

$$\text{Μείωση} = \text{Αρχική τιμή} * \text{Ποσοστό}$$

Η τελική τιμή που προκύπτει είναι:

$$\text{Τελική τιμή} = \text{Αρχική τιμή} - \text{Μείωση}$$

Τα 2 βήματα μπορούν να γίνουν και ένα αν δεν μας ζητείται η μείωση σαν ξεχωριστό ερώτημα:

$$\text{Τελική τιμή} = \text{Αρχική τιμή} - (\text{Αρχική τιμή} * \text{Μείωση})$$

Υπολογισμός ποσοστιαίας μεταβολής

Στην ποσοστιαία μεταβολή βρίσκουμε κατά ποιο ποσοστό μεταβλήθηκαν 2 τιμές. ($\text{Τελική τιμή} - \text{Αρχική τιμή}$) / $\text{Αρχική τιμή} * 100$

Αν ένα από τα 2 μέρη δεν είναι δεκαδικός, λόγω διαίρεσης ο τύπος πρέπει να γίνει:

$$(\text{float}(\text{Τελική τιμή}) - \text{Αρχική τιμή}) / \text{Αρχική τιμή} * 100$$

Προσοχή στην παρένθεση που πρέπει να περιλαμβάνει την αφαίρεση.

Υπολογισμός τόκων

Οι τόκοι είναι μια αύξηση οπότε ισχύει ο ίδιος τύπος με την αύξηση. Η διαφορά είναι ότι οι τόκοι υπολογίζονται για μια σειρά ετών και η αύξηση κάθε φορά γίνεται πάνω

στο ήδη αυξημένο ποσοστό. Επομένως για να υπολογίσω τόκους χρειάζομαι επανάληψη με τον αλγόριθμό:

Συνθήκη επανάληψης:

τόκος = αρχικό ποσό * επιτόκιο

τελικό ποσό = αρχικό ποσό + τόκος

Εμφάνιση τελικού ποσού

Αν οι τόκοι δεν ζητούνται ως ερώτημα, μπορούμε να υπολογίσουμε το τελικό ποσό σε μια εντολή όπως και στην αύξηση.

Προσοχή: Επιτόκιο λέγεται το ποσοστό αύξησης και είναι ποσοστό. Επομένως ισχύουν τα ίδια για την αναπαράσταση ποσοστών (δεκαδικό).

Υπολογισμός αθροίσματος (Αθροιστής)

Ο υπολογισμός ενός αθροίσματος τιμών γίνεται μέσα σε μια δομή επανάληψης. Αν ζητείται το άθροισμα από «κάτι», τότε το άθροισμα υπολογίζεται από τον αλγόριθμο:

Αθροισμα = 0

Συνθήκη επανάληψης:

Αθροισμα = Αθροισμα + κάτι

Εμφάνιση αθροίσματος

Προσοχή: Η μεταβλητή του αθροίσματος αρχικοποιείται ΠΡΙΝ την επανάληψη. ΜΕΣΑ στην επανάληψη υπολογίζεται το άθροισμα και η εμφάνιση γίνεται ΜΕΤΑ την επανάληψη (εκτός αν ζητείται διαφορετικά).

Υπολογισμός πλήθους (Μετρητής)

Όταν μας ζητείται να υπολογίσουμε το πλήθος από κάτι, τότε χρησιμοποιούμε μετρητή γιατί στην ουσία πρέπει να μετρήσουμε πόσα είναι αυτά τα κάτι. Ο υπολογισμός γίνεται μέσα σε δομή επανάληψης και ο αλγόριθμος είναι:

Πλήθος = 0

Συνθήκη επανάληψης:

Πλήθος = Πλήθος + 1

Προσοχή: Η μεταβλητή του πλήθους αρχικοποιείται ΠΡΙΝ την επανάληψη. ΜΕΣΑ στην επανάληψη υπολογίζεται το πλήθος και η εμφάνιση μπορεί να γίνεται είτε ΠΡΙΝ είτε ΜΕΤΑ την επανάληψη, ανάλογα με το ζητούμενο.

Υπολογισμός μέσου όρου

Ο υπολογισμός του μέσου όρου απαιτεί το άθροισμα κάποιων τιμών και το πλήθος των τιμών αυτών. Συνεπώς απαιτεί να βρεθούν πρώτα το άθροισμα και το πλήθος.

Προσοχή: Η μεταβλητή του αθροίσματος και του πλήθους, αρχικοποιούνται ΠΡΙΝ την επανάληψη. ΜΕΣΑ στην επανάληψη υπολογίζονται το άθροισμα και το πλήθος που απαιτούνται. Ο υπολογισμός του Μέσου όρου γίνεται ΜΕΤΑ την επανάληψη ΠΑΝΤΑ και επειδή πρόκειται για διαίρεση πρέπει να φροντίσουμε για 2 πράγματα: Έλεγχος ότι το πλήθος δεν είναι μηδέν και μετατροπή ενός από τους 2 αριθμούς σε float.

```

Αθροισμα = 0
Πλήθος = 0
Συνθήκη επανάληψης:
    Αθροισμα = Αθροισμα + κάτι
    Πλήθος = Πλήθος + 1
Αν Πλήθος != 0:
    Μέσος όρος = float(Αθροισμα) / Πλήθος
    Εμφάνισε Μέσος όρος
Αλλιώς:
    Εμφάνισε «Δεν δώσατε τιμές»
    
```

Διάβασμα τιμών με έλεγχο ορθότητας ή εγκυρότητας (Μέρος 2ο)

Για να ελέγξουμε αν οι τιμές που εισάγονται είναι έγκυρες χρειαζόμαστε μια επαναληπτική δομή while αμέσως μετά το διάβασμα των τιμών. Η συνθήκη που ελέγχεται είναι η αντίθετη από τις τιμές που δεχόμαστε ως σωστές.

```

Μεταβλητή = input("Δώσε τιμή:")
while not (έλεγχος μεταβλητής με σωστή τιμή):
    Μεταβλητή = input("Δώσε τιμή:")
    
```

ή

```

Μεταβλητή = input("Δώσε τιμή:")
while έλεγχος μεταβλητής με λάθος τιμή:
    Μεταβλητή = input("Δώσε τιμή:")
    
```

Προσοχή: Διαβάζουμε την τιμή μια φορά ΠΡΙΝ την επανάληψη και στη συνέχεια διαβάζουμε άλλη μια φορά ΜΕΣΑ, ακριβώς πριν βγούμε από την επανάληψη. Αυτό συμβαίνει μόνο για τη μεταβλητή που συμμετέχει στη συνθήκη. Οι υπόλοιπες τιμές του προγράμματος μπορούν να διαβαστούν είτε μέσα είτε έξω από την επανάληψη ανάλογα με το τι ζητείται.

Παραδείγματα:

Να διαβάζονται θετικοί αριθμοί μόνο.

```

arithmos = input("Δώσε αριθμό:")
while not (arithmos >= 0):
    arithmos = input("Δώσε αριθμό:")
    
```

Εναλλακτικά η συνθήκη: while arithmos < 0:

Να διαβαστούν οι βαθμοί ενός μαθητή με έλεγχο εγκυρότητας από 0 έως 20.

```

vathmos = input("Δώσε βαθμό:")
while not (vathmos >= 0 and vathmos <= 20):
    vathmos = input("Δώσε βαθμό:")
    
```

Εναλλακτικά: while vathmos < 0 or vathmos > 20:

Να διαβάζονται όλοι οι χαρακτήρες εκτός από το Ν και το Ο.

```

xarakt = raw_input("Δώσε χαρακτήρα:")
while not (xarakt == "N" or xarakt == "O"):
    xarakt = raw_input("Δώσε χαρακτήρα:")
    
```

Εναλλακτικά: while xarakt != "N" and xarakt != "O":

Βασικές οδηγίες στη Δομή Επανάληψης while

1. Η μεταβλητή/μεταβλητές που συμμετέχουν στη συνθήκη της while θα πρέπει ΠΑΝΤΑ να έχουν πάρει τιμή ΠΡΙΝ την επανάληψη.
2. Τουλάχιστον μια από τις μεταβλητές που συμμετέχουν στη συνθήκη της while θα πρέπει ΠΑΝΤΑ να αλλάζουν τιμή ΜΕΣΑ στην επανάληψη.
3. Όταν διαβάζουμε τιμές από τις οποίες εξαρτάται η συνθήκη, τότε (σύμφωνα και με τα παραπάνω) διαβάζουμε μια φορά ΕΞΩ από την επανάληψη και άλλη μια φορά ΜΕΣΑ στην επανάληψη, ως τελευταία εντολή πριν βγούμε.
4. Αν έχουμε και έλεγχο εγκυρότητας, τότε το διάβασμα περιλαμβάνει τον έλεγχο εγκυρότητας και ΠΡΙΝ αλλά και ΜΕΣΑ στην επανάληψη.

Μέγιστο/Ελάχιστο

Για να βρω τη μεγαλύτερη ή την μικρότερη από ένα σύνολο τιμών (συνήθως μέσα σε μια επανάληψη) ακολουθώ τα παρακάτω βήματα.

1. Ορίζω την πρώτη τιμή που εισάγεται ως μέγιστη/ελάχιστη (max/min) ΕΞΩ από την επανάληψη.
2. Εκτελώ τον αλγόριθμο εύρεσης μεγίστου/ελαχίστου.

Παρατήρηση: Αν δεν μπορεί να δοθεί ως αρχική τιμή των min και max η πρώτη τιμή που εισάγεται, τότε αρχικοποιούμε εμείς, δηλαδή δίνουμε εμείς τιμές στο min και max. Το max παίρνει πάντα μία πολύ μικρή τιμή και το min παίρνει πάντα μια πολύ μεγάλη τιμή.

```
Μεταβλητή = input("Δώσε τιμή:")
max = Μεταβλητή
min = -Μεταβλητή
while Συνθήκη εξόδου:
    if Μεταβλητή > max:
        max = Μεταβλητή
    if Μεταβλητή < min:
        min = Μεταβλητή
    Μεταβλητή = input("Δώσε τιμή:")
Εμφάνισε min, max
```

Προσοχή: Οι μεταβλητές min, max αρχικοποιούνται ΠΡΙΝ την επανάληψη. ΜΕΣΑ στην επανάληψη υπολογίζεται το μέγιστο και ελάχιστο και η εμφάνιση γίνεται ΜΕΤΑ την επανάληψη.

Μετατροπή for σε while

- ΟΛΕΣ οι for μπορούν να μετατραπούν σε while.

Για τις μετατροπές ακολουθούμε τους παρακάτω αλγόριθμους:

```
for μεταβλητή in range(αρχή, τέλος, βήμα):
    εντολές
```



```
μεταβλητή = αρχή
while μεταβλητή < τέλος:
    εντολές
    μεταβλητή = μεταβλητή + βήμα
```

Προσοχή: Αν έχω αρνητικό βήμα, η συνθήκη μετατρέπεται σε μεταβλητή > τέλος και η τελευταία εντολή σε μεταβλητή = μεταβλητή - βήμα.

Παράδειγμα

```
for i in range (1,10):
    print i

i=1
while i<10:
    print i
    i=i+1
```

Μετατροπή while σε for

- ΔΕΝ μπορούν όλες οι while να μετατραπούν σε for.

```
μεταβλητή = αρχή
while μεταβλητή < τέλος:
    εντολές
    μεταβλητή = μεταβλητή + βήμα
```



```
for μεταβλητή in range(αρχή, τέλος, βήμα):
    εντολές
```

Παράδειγμα

```
i=2
while i<72:
    print i
    i=i+4

for i in range (2,72,4):
    print i
```