

Αλγόριθμος Άθροισμα_Άρτιων
!Στη μεταβλητή Σ θα αποθηκευθεί το άθροισμα.
Σ <-- 0
Για μ από 1 μέχρι 100
!Διακρίνουμε αν ένας αριθμός είναι άρτιος όταν το υπόλοιπο της διαίρεσης του με το 2 δίνει
!αποτέλεσμα 0.
Αν $\mu \bmod 2 = 0$ τότε
 Σ <-- Σ + μ
Τέλος_αν
Τέλος_επανάληψης
Εμφάνισε Σ
Τέλος_Άθροισμα_Άρτιων

1

Αλγόριθμος ΜΟ_50
!Για να βρούμε τον ΜΟ χρειαζόμαστε άθροισμα και πλήθος. Αρχικοποίηση αυτών των μεταβλητών.
Σ <-- 0
Πλήθος <-- 0
Για μ από 1 μέχρι 50
!Διαβάζουμε 50 αριθμούς.
Διάβασε α
!Κάθε φορά που βρίσκουμε αριθμό ο οποίος ικανοποιεί την παρακάτω συνθήκη αυξάνουμε το άθροισμα
!και το πλήθος των αριθμών που βρέθηκαν κατά μια μονάδα.
Αν $(\alpha \bmod 10 < 5)$ τότε
 Σ <-- Σ + α
 Πλήθος <-- Πλήθος + 1
Τέλος_αν
Τέλος_επανάληψης
!Ελέγχεται αν το πλήθος είναι διαφορετικό του μηδενός
Αν Πλήθος <> 0 τότε
!Υπολογίζουμε τον ΜΟ και τον εμφανίζουμε
ΜΟ <-- Σ / Πλήθος
Εμφάνισε "Ο μέσος όρος είναι ", ΜΟ
Αλλιώς
Εμφάνισε "Δεν δόθηκαν αριθμοί που το τελευταίο τους ψηφίο είναι μικρότερο του 5"
Τέλος_αν
Τέλος_ΜΟ_50

2

Αλγόριθμος ΜΟ_αριθμων
!Αρχικά ζητάμε από το χρήστη να δώσει το πλήθος των αριθμών που θα διαβαστούν.
Εμφάνισε 'Πόσους αριθμούς θα διαβάσω;'
Διάβασε Ν
!Αρχικοποίηση της μεταβλητής που θα μετράει το άθροισμα των αριθμών εισόδου.
Σ <-- 0
Για i από 1 μέχρι Ν
 Εμφάνισε 'Δώσε αριθμό: '
 Διάβασε x
 Σ <-- Σ + x
Τέλος_επανάληψης
!Ελέγχεται αν το πλήθος των επαναλήψεων είναι μεγαλύτερο του μηδενός
Αν Ν > 0 τότε
 ΜΟ <-- Σ / Ν
 Εμφάνισε 'Ο μέσος όρος είναι ', ΜΟ
Αλλιώς
Εμφάνισε "Δεν δόθηκε σωστό πλήθος επαναλήψεων"
Τέλος_αν
Τέλος_ΜΟ_αριθμών

3

Αλγόριθμος ΜΟ_Περιττών
!Αρχικοποιούμε τις μεταβλητές που θα μετράνε το άθροισμα και το πλήθος των περιττών αριθμών.
Σ <-- 0
Πλήθος <-- 0
!Διαβάζουμε τον πρώτο αριθμό εκτός βρόχου.
Διάβασε x
!Ο βρόχος θα πρέπει να σταματάει όταν δοθεί ως είσοδος αρνητικός αριθμός.
!Αυτό σημαίνει πως αν ο (κάθε) αριθμός είναι ≥ 0 ο βρόχος συνεχίζεται.
Όσο x ≥ 0 επανάλαβε
 !Ελέγχουμε αν ο αριθμός είναι περιττός.
 Αν $x \bmod 2 <> 0$ τότε
 Σ <-- Σ + x
 Πλήθος <-- Πλήθος + 1
 Τέλος_αν
 !Διαβάζουμε τον επόμενο αριθμό ακριβώς πριν κλείσει ο βρόχος.
 !Αν δεν το κάνουμε αυτό, ο βρόχος θα εκτελείται για πάντα (ατέρμων βρόχος).
 Διάβασε x
Τέλος_επανάληψης
!Ελέγχεται αν το πλήθος είναι διαφορετικό του μηδενός
Αν Πλήθος <> 0 τότε
 !Υπολογίζουμε τον ΜΟ και τον εμφανίζουμε
 ΜΟ <-- Σ / Πλήθος
 Εμφάνισε 'Ο Μ.Ο. είναι ', ΜΟ
Αλλιώς
Εμφάνισε "Δεν δόθηκαν Περιττοί"
Τέλος_αν
Τέλος_ΜΟ_Περιττών

4

Αλγόριθμος ΜΟ_άρτιων_και_περιττών

!Επειδή σε αυτή την άσκηση πρέπει να βρούμε δυο διαφορετικούς μέσους όρους,
!θα χρειαστούμε δυο διαφορετικές μεταβλητές για να μετράμε τα αθροίσματα άρτιων
!και περιττών και δυο μεταβλητές για τα πλήθη τους.

Σ1 <-- 0

Σ2 <-- 0

Πλήθος_άρτιων <-- 0

Πλήθος_περιττών <-- 0

Όσο Σ1<=2000 ή Σ2<=5000 επανέλαβε

!Διαβάζουμε αριθμούς.

Διάβασε x

!Ελέγχεται ο κάθε αριθμός αν είναι άρτιος ή περιττός.

Αν $x \bmod 2 = 0$ τότε

Σ1 <-- Σ1 + x

Πλήθος_άρτιων <-- Πλήθος_άρτιων + 1

Αλλιώς

Σ2 <-- Σ2 + x

Πλήθος_περιττών <-- Πλήθος_περιττών + 1

Τέλος_αν

Τέλος_επανάληψης

ΜΟ1 <-- Σ1 / Πλήθος_άρτιων

ΜΟ2 <-- Σ2 / Πλήθος_περιττών

Εμφάνισε ΜΟ1, ΜΟ2

Τέλος ΜΟ_άρτιων_και_περιττών

5

Αλγόριθμος Έλεγχος_διαστήματος

Διάβασε x

Όσο $x < -100$ ή $x \geq 100$ επανάλαβε

Εμφάνισε 'Λάθος τιμή. Ξαναδώσε.'

Διάβασε x

Τέλος_επανάληψης

Τέλος Έλεγχος_διαστήματος

6

Αλγόριθμος Έλεγχος_A_K

!Το x είναι μεταβλητή χαρακτήρα.

Διάβασε x

Όσο $x \neq 'A'$ και $x \neq 'K'$ επανάλαβε

Εμφάνισε 'Λάθος. Δώσε νέα τιμή.'

Διάβασε x

Τέλος_επανάληψης

Τέλος Έλεγχος_A_K

7

Αλγόριθμος Καλημέρα_Καληνύχτα

!Μέσα στην αρχή επανάληψης γίνεται ο έλεγχος εγκυρότητας τιμής.

Αρχή_επανάληψης

Διάβασε x

Μέχρις_ότου $x = 1$ ή $x = 2$

Αν $x = 1$ τότε

Εμφάνισε 'Καλημέρα'

Αλλιώς

Εμφάνισε 'Καληνύχτα'

Τέλος_αν

Τέλος Καλημέρα_Καληνύχτα

8

Αλγόριθμος Αθροισμα_άρτιων_περιττών

!Αρχειοποίηση των αθροισμάτων.

Σ1 <-- 0

Σ2 <-- 0

Όσο Σ1<2000 ή Σ2<2000 επανέλαβε

!Αμέσως σε μια νέα δομή επανάληψης Μέχρις_ότου θα γίνει ο έλεγχος εγκυρότητας τιμής.

!Για τον έλεγχο του αριθμού εάν αυτός είναι ακέραιος θα χρησιμοποιηθεί η συνάρτηση A_M(...).

Αρχή_επανάληψης

Διάβασε x

Μέχρις_ότου $A_M(x) - x = 0$

!Ελέγχεται αν ο αριθμός είναι άρτιος ή περιττός.

Αν $x \bmod 2 = 0$ τότε

Σ1 <-- Σ1 + x

Αλλιώς

Σ2 <-- Σ2 + x

Τέλος_αν

Τέλος_επανάληψης

Εμφάνισε 'Αθροισμα άρτιων: ', Σ1

Εμφάνισε 'Αθροισμα περιττών: ', Σ2

Τέλος Αθροισμα_άρτιων_περιττώ

9

Αλγόριθμος Μέγιστος_Ελάχιστος_1000

!Σε ασκήσεις όπου ζητείται να βρεθεί ο μέγιστος ή ο ελάχιστος αριθμός (ή και οι δυο), πάντα

!ως πρώτη τιμή στις μεταβλητές του μέγιστου και ελάχιστου αποθηκεύεται ο πρώτος αριθμός που διαβάζεται.

!Στη συνέχεια με χρήση επαναληπτικής δομής διαβάζονται όλοι οι υπόλοιποι αριθμοί εισόδου και γίνονται οι

!απαραίτητες συγκρίσεις.

!Διαβάζεται ο πρώτος αριθμός (από τους 1000).

Διάβασε a

10

```

!Στα max και min εκκωρείται ο αριθμός που διαβάστηκε.
max <-- α
min <-- α
!Γίνεται βρόχος για τους υπόλοιπους 999 αριθμούς.
Για μ από 1 μέχρι 999 !Θα ήταν σωστό να πούμε και για μ από 2 μέχρι 1000
  Διάβασε α
  Αν α > max τότε
    max <-- α
  Τέλος_αν
  Αν α < min τότε
    min <-- α
  Τέλος_αν
Τέλος_επανάληψης
Εμφάνισε max, min
Τέλος Μέγιστος_Ελάχιστος_1000

```

Προσοχή: Στη συγκεκριμένη άσκηση εφόσον γίνεται έλεγχος εγκυρότητας το α θα έχει τιμή από 1 μέχρι 100. Αυτό σημαίνει πως θα μπορούσαμε αυθαίρετα να δίνουμε στο max μια πολύ μικρή τιμή (προκειμένου να αλλάξει από τον πρώτο κιάλας έλεγχο) και στο min μια πολύ μεγάλη τιμή (προκειμένου και εδώ να αλλάξει από τον πρώτο κιάλας έλεγχο). Δηλαδή να δώσουμε στο max<--0 και στο min<--101. Η άσκηση θα λυνόταν με τον παρακάτω τρόπο:

```

Αλγόριθμος Μέγιστος_Ελάχιστος_500
!Στα max & min δίνεται η αρχική τιμή.
max <-- 0
min <-- 101
!Γίνεται βρόχος για τις υπόλοιπες 499 τιμές που πρέπει να διαβαστούν.
Για μ από 1 μέχρι 500
  !Σε κάθε επανάληψη θα γίνεται έλεγχος εγκυρότητας τιμής.
  Αρχή_επανάληψης
    Διάβασε α
    Μέχρις_ότου α >= 1 και α <= 100
    !Έλεγχος για max ή min.
    Αν α > max τότε
      max <-- α
    Τέλος_αν
    Αν α < min τότε
      min <-- α
    Τέλος_αν
  Τέλος_επανάληψης
Εμφάνισε max, min
Τέλος Μέγιστος_Ελάχιστος_500

```

11

```

Αλγόριθμος Άθροισμα_άρτιων_και_περιττών
!Επειδή δεν είναι γνωστό των πλήθος των επαναλήψεων, σε αυτή την άσκηση
!δεν μπορούμε στα max να εκκωρήσουμε ως αρχική τιμή τον πρώτο αριθμό που
!θα διαβαστεί. Για αυτό ως αρχικές τιμές δίνουμε έναν αρνητικό αριθμό, ο οποίος
!έτσι και αλλιώς θα δώσει τη θέση του στον πρώτο αριθμό (άρτιο ή περιττό αντίστοιχα)
!που θα διαβαστεί.
μεγ_άρτιος <-- -1
μεγ_περιττός <-- -1
!Αρχικοποίηση και των μεταβλητών των αθροισμάτων.
Σ_άρτιων <-- 0
Σ_περιττών <-- 0
Όσο Σ_άρτιων < 2000 ή Σ_περιττών < 2000 επανάλαβε
  !Εδώ μπαίνει ο έλεγχος εγκυρότητας τιμής.
  Αρχή_επανάληψης
    Διάβασε α
    Μέχρις_ότου α > 0
    !Ελέγχεται αν ο αριθμός είναι άρτιος ή περιττός και αντίστοιχα
    !στη συνέχεια γίνεται η σύγκριση για το μέγιστο.
    Αν α mod 2 = 0 τότε
      Αν α > μεγ_άρτιος τότε
        μεγ_άρτιος <-- α
      Τέλος_αν
      Σ_άρτιων<-- Σ_άρτιων + α
    Αλλιώς
      Αν α > μεγ_περιττός τότε
        μεγ_περιττός <-- α
      Τέλος_αν
      Σ_περιττών<-- Σ_περιττών + α
    Τέλος_αν
  Τέλος_επανάληψης
Εμφάνισε μεγ_άρτιος, μεγ_περιττός
Τέλος Άθροισμα_άρτιων_και_περιττών

```

12

Αλγόριθμος Ακέραιοι_σε_ανοικτό_διάστημα

!Αρχικά θα ζητηθεί από το χρήστη να δώσει δυο αριθμούς (α και β) σύμφωνα με την άσκηση.

!Στη συνέχεια θα γίνει έλεγχος για τον μεγαλύτερο από αυτούς. Επειδή το διάστημα είναι

!ανοικτό ο μεγαλύτερος συν μια μονάδα θα αποτελέσει την τιμή εκκίνησης του βρόχου,

!ενώ ο δεύτερος μείον μια μονάδα θα αποτελέσει την τελευταία τιμή του βρόχου.

!Ο μεγαλύτερος και ο μικρότερος αντίστοιχα θα αποθηκευθούν σε δυο άλλες μεταβλητές

!για να χρησιμοποιηθούν στο βρόχο.

Εμφάνισε 'Δώσε δυο ακέραιους αριθμούς: '

Διάβασε α, β

Αν $a < b$ τότε

αρχική_τιμή $\leftarrow a + 1$

τελική_τιμή $\leftarrow b - 1$

Αλλιώς !Δηλαδή $a > b$

αρχική_τιμή $\leftarrow b + 1$

τελική_τιμή $\leftarrow a - 1$

Τέλος_αν

Για μ από αρχική_τιμή μέχρι τελική_τιμή

Εμφάνισε μ

Τέλος_επανάληψης

Τέλος Ακέραιοι_σε_ανοικτό_διάστημα

13

Αλγόριθμος Ανάποδοι_αριθμοί

!Για να διαβάζεται ένας αριθμός κανονικά και ανάποδα με τον ίδιο τρόπο, θα πρέπει το πρώτο ψηφίο

!με το τελευταίο και το δεύτερο ψηφίο με το τρίτο να είναι ίδια. Για να ελεγχθεί αν τα ψηφία είναι ίδια, θα πρέπει

!να γίνει πρώτα διαχωρισμός τους. Αν μετά τον έλεγχο ισχύουν τα παραπάνω, ο αριθμός θα εμφανίζεται.

Για μ από 1000 μέχρι 9999

!Γίνεται ο διαχωρισμός των ψηφίων.

$\Psi_1 \leftarrow \mu \text{ div } 1000$

$\Upsilon_1 \leftarrow \mu \text{ mod } 1000$

$\Psi_2 \leftarrow \Upsilon_1 \text{ div } 100$

$\Upsilon_2 \leftarrow \Upsilon_1 \text{ mod } 100$

$\Psi_3 \leftarrow \Upsilon_2 \text{ div } 10$

$\Psi_4 \leftarrow \Upsilon_2 \text{ mod } 10$

!Ελέγχεται αν $\Psi_1 = \Psi_4$ και $\Psi_2 = \Psi_3$.

Αν $\Psi_1 = \Psi_4$ και $\Psi_2 = \Psi_3$ τότε

Εμφάνισε μ

Τέλος_αν

Τέλος_επανάληψης

Τέλος Ανάποδοι_αριθμοί

14

Αλγόριθμος Πρώτοι_αριθμοί

!Πρώτος λέγεται ένας αριθμός που διαιρείται μόνο με τον εαυτό του και τη μονάδα.

!Σε αυτή την άσκηση θα χρειαστούν δυο βρόχοι. Ο πρώτος θα χρησιμοποιηθεί για

!να διαβαστούν όλοι οι αριθμοί από το 2 μέχρι το 1000. Ο δεύτερος (και εμφωλευμένος)

!βρόχος θα χρησιμοποιηθεί για να ελεγχθεί αν γίνεται τέλεια διαίρεση του κάθε αριθμού

!από τον πρώτο βρόχο με όλους τους αριθμούς από το 1 μέχρι τον εαυτό του. Σε κάθε

!τέλεια διαίρεση που θα προκύπτει θα αυξάνεται ο αριθμός των διαιρετών κατά μια μονάδα.

!Αν μετά το τέλος του δεύτερου βρόχου οι διαιρέτες είναι 2, τότε ο αριθμός είναι πρώτος και

!θα εμφανίζεται.

Για μ από 2 μέχρι 1000

!Σε κάθε επανάληψη θα πρέπει να γίνεται αρχικοποίηση αυτής της μεταβλητής.

διαρ $\leftarrow 0$

Για ν από 1 μέχρι μ

!Ελέγχεται αν ένας αριθμός διαιρείται τέλεια με κάποιον άλλο χρησιμοποιώντας το υπόλοιπο.

Αν $\mu \text{ mod } \nu = 0$ τότε

διαρ $\leftarrow \text{διαρ} + 1$

Τέλος_αν

Τέλος_επανάληψης

!Ελέγχεται το πλήθος των διαιρετών.

Αν διαρ = 2 τότε

Εμφάνισε μ

Τέλος_αν

Τέλος_επανάληψης

Τέλος Πρώτοι_αριθμοί

15

Αλγόριθμος Θερμοκρασίες

!Ο αλγόριθμος θα πρέπει να διαβάζει μια θερμοκρασία για κάθε μέρα του Ιανουαρίου,

!δηλαδή συνολικά 31 θερμοκρασίες (βρόχος με 31 επαναλήψεις).

!Αρχικοποίηση των μεταβλητών που θα χρειαστούν.

$\Sigma_{\text{θερμ}} \leftarrow 0$

Για μ από 1 μέχρι 31

Εμφάνισε 'Δώσε θερμοκρασία: '

Διάβασε θερμ

$\Sigma_{\text{θερμ}} \leftarrow \Sigma_{\text{θερμ}} + \text{θερμ}$

Τέλος_επανάληψης

$MO_{\text{θερμ}} \leftarrow \Sigma_{\text{θερμ}} / 31$

Εμφάνισε 'Ο μέσος όρος των θερμοκρασιών για το μήνα Ιανουάριο είναι ', $MO_{\text{θερμ}}$

Τέλος Θερμοκρασίες

16

Αλγόριθμος Γενική_Βαθμολογία

!Εφόσον θα πρέπει ο αλγόριθμος να ζητάει τους βαθμούς (γραπτούς και προφορικούς) 30 μαθητών,
!θα υλοποιηθεί μια δομή επανάληψης η οποία θα εκτελεστεί 30 φορές.

!Για να υπολογιστεί και να εμφανιστεί το ποσοστό των μαθητών που δεν προβιβάζονται, θα πρέπει
!πρώτα να βρεθεί πόσοι μαθητές βγάζουν μέσο όρο βαθμολογίας κάτω από τη βάση που δίνεται στην
!άσκηση. Για αυτό θα χρειαστεί να μετρηθεί το πλήθος τους (κάθε φορά που θα υπολογίζεται γενικός βαθμός
!κάτω από τη βάση το πλήθος θα αυξάνεται κατά μια μονάδα).

Πλ <-- 0

Για μ από 1 μέχρι 30

!Αρχικά διαβάζεται η βαθμολογία σε προφορικά και γραπτά για κάθε μαθητή.

Εμφάνισε 'Δώσε προφορικό βαθμό: '

Διάβασε B_προφ

Εμφάνισε 'Δώσε γραπτό βαθμό: '

Διάβασε B_γρ

!Υπολογίζεται ο γενικός βαθμός.

Γεν_βαθμός <-- 0.7 * B_γρ + 0.3 * B_προφ

!Ελέγχεται αν ο γενικός βαθμός είναι κάτω από 10. Αν ισχύει αυτό, θα αυξήσουμε το πλήθος
!μας κατά ένα.

Αν Γεν_βαθμός < 10 τότε

Πλ <-- Πλ + 1

Τέλος_αν

Τέλος_επανάληψης

!Το πλήθος των μαθητών που δεν προβιβάζονται και υπολογίστηκε παραπάνω μετατρέπεται σε μορφή ποσοστού.

Ποσοστό <-- (Πλ * 100) / 30

Εμφάνισε 'Το ποσοστό των μαθητών που δεν προβιβάζονται είναι ', Ποσοστό

Τέλος Γενική_Βαθμολογία

17

Αλγόριθμος Εξεταστική_Διαδικασία

!Για τη λύση αυτής της άσκησης θα χρειαστούν 3 μεταβλητές που θα μετράνε το πλήθος των

!διαγωνιζομένων του κάθε διαστήματος βαθμολογίας που ζητάει η άσκηση. Επίσης

!θα χρειαστούν και 3 διαφορετικές μεταβλητές αθροισμάτων για τις ηλικίες, ώστε να

!υπολογιστεί στο τέλος ο μέσος όρος τους για καθένα από τα παραπάνω διαστήματα.

Πλ1 <-- 0

Πλ2 <-- 0

Πλ3 <-- 0

Σ1 <-- 0

Σ2 <-- 0

Σ3 <-- 0

!Γίνεται βρόχος με 100 επαναλήψεις (όσοι και οι διαγωνιζόμενοι).

Για μ από 1 μέχρι 100

!Αρχικά γίνεται ο έλεγχος εγκυρότητας της βαθμολογίας.

Εμφάνισε 'Δώσε βαθμολογία διαγωνιζόμενου: '

Διάβασε βαθμ

Όσο βαθμ < 1 ή βαθμ > 200 επανάλαβε

Εμφάνισε 'Λάθος βαθμολογία. Ξαναδώσε: '

Διάβασε βαθμ

Τέλος_επανάληψης

!Στη συνέχεια διαβάζεται και η ηλικία του διαγωνιζόμενου.

!Παρατήρηση: Διαβάζουμε πρώτα τη βαθμολογία και κάνουμε τον έλεγχο εγκυρότητας της

!και μετά διαβάζουμε την ηλικία, ώστε αν δίνεται λάθος βαθμολογία να μην υποχρεώνεται

!ο χρήστης να εισάγει ξανά και την ηλικία. Με αυτό τον τρόπο η ηλικία για κάθε διαγωνιζόμενο

!θα δίνεται μια φορά μόνο και αφού δοθεί σωστή βαθμολογία.

Εμφάνισε 'Δώσε ηλικία διαγωνιζόμενου: '

Διάβασε ηλικία

!Ελέγχεται σε ποιο διάστημα ανήκει η βαθμολογία.

Αν βαθμ > 0 και βαθμ <= 100 τότε

!Αυξάνεται το πλήθος των διαγωνιζομένων με βαθμολογία σε αυτό το διάστημα.

Πλ1 <-- Πλ1 + 1

Σ1 <-- Σ1 + ηλικία

Αλλιώς_αν βαθμ <= 150 τότε

!Αντίστοιχα με πριν.

Πλ2 <-- Πλ2 + 1

Σ2 <-- Σ2 + ηλικία

Αλλιώς

!Αντίστοιχα με πριν.

Πλ3 <-- Πλ3 + 1

Σ3 <-- Σ3 + ηλικία

Τέλος_αν

Τέλος_επανάληψης

!Υπολογίζεται ο μέσος όρος των ηλικιών για κάθε διάστημα.

ΜΟ1 <-- Σ1 / Πλ1

ΜΟ2 <-- Σ2 / Πλ2

ΜΟ3 <-- Σ3 / Πλ3

Εμφάνισε Πλ1, ΜΟ1, Πλ2, ΜΟ2, Πλ3, ΜΟ3

Τέλος Εξεταστική_διαδικασία

18

Αλγόριθμος Μάθημα_Πληροφορικής

!Σε αυτή την άσκηση, επειδή είναι γνωστό το εύρος τιμών της βαθμολογίας των μαθητών,

!ως αρχική τιμή στη μέγιστη και ελάχιστη βαθμολογία που ζητείται μπορούμε να δώσουμε

!αντίστοιχα μια τιμή κάτω από το 0 για το μέγιστο και μια τιμή πάνω από 1000 για το ελάχιστο.

μεγ_βαθμ <-- -1

ελ_βαθμ <-- 1001

!Γίνεται βρόχος με 25 επαναλήψεις.

Για μ από 1 μέχρι 25

19

!Κατά τον έλεγχο εγκυρότητας τιμής, αν δεν θέλουμε να εμφανίσουμε κάποιο μήνυμα
!λάθους κάθε φορά που θα δίνεται βαθμολογία εκτός ορίων, μπορούμε να χρησιμοποιήσουμε
!τη δομή Μέχρις_ότου. Διαφορετικά βολεύει περισσότερο να χρησιμοποιηθεί η Όσο... επανάλαβε.
!Με τον παρακάτω τρόπο κάθε φορά που θα δίνεται λάθος βαθμολογία θα εμφανίζεται το ίδιο
!μήνυμα στο χρήστη και όχι μήνυμα λάθους.

```
Αρχή_επανάληψης
    Εμφάνισε 'Δώσε βαθμολογία: '
    Διάβασε βαθμ
Μέχρις_ότου βαθμ >= 0 και βαθμ <= 1000
!Έλεγχος για τη μέγιστη και ελάχιστη βαθμολογία.
Αν βαθμ > μεγ_βαθμ τότε
    μεγ_βαθμ <-- βαθμ
Τέλος_αν
Αν βαθμ < ελ_βαθμ τότε
    ελ_βαθμ <-- βαθμ
Τέλος_αν
Τέλος_επανάληψης
Εμφάνισε μεγ_βαθμ, ελ_βαθμ
Τέλος Μάθημα_Πληροφορικής
```

20

!Βρόχος με 100 επαναλήψεις (όσοι και οι μαθητές).
Για μ από 1 μέχρι 100
!Αρχικοποίηση της λογικής μεταβλητής.
Λάθος_βαθμολογία <-- Ψευδής
!Έλεγχος εγκυρότητας βαθμολογίας. Θα γίνει με χρήση της Όσο... επανάλαβε
!ώστε κάθε φορά που θα δίνεται λάθος βαθμός να εμφανίζεται μήνυμα λάθους.
Εμφάνισε 'Δώσε βαθμό απολυτηρίου: '
Διάβασε βαθμ
Όσο βαθμ < 0 ή βαθμ > 20 επανάλαβε
 Εμφάνισε 'Έδωσες λάθος βαθμό. Δώσε πάλι: '
 Διάβασε βαθμ
 !Αυξάνεται το πλήθος των βαθμών που δόθηκαν λάθος.
 Πλ_λάθος <-- Πλ_λάθος + 1
 !Η μεταβλητή που "δείχνει" αν καταχωρήθηκε λάθος βαθμός γίνεται αληθής.
 Λάθος_βαθμολογία <-- Αληθής
Τέλος_επανάληψης
!Ελέγχεται η τιμή της Λάθος_βαθμολογία. Αν είναι αληθής τότε ο χρήστης έδωσε για
!τον τρέχοντα μαθητή ήδη μια βαθμολογία λάθος, οπότε αυξάνεται και ο αντίστοιχος
!μετρητής.
Αν Λάθος_βαθμολογία = Αληθής τότε
 Πλ_μαθ <-- Πλ_μαθ + 1
Τέλος_αν
!Αυξάνεται το συνολικό άθροισμα των βαθμών
Σ <-- Σ + βαθμ
Τέλος_επανάληψης
ΜΟ <-- Σ / 100
Εμφάνισε 'Ο Μ.Ο. των βαθμών είναι ', ΜΟ
Εμφάνισε 'Το πλήθος των λάθος βαθμών που δόθηκαν είναι ', Πλ_λάθος
Εμφάνισε 'Το πλήθος των μαθητών για τους οποίους έγινε τουλάχιστον μια λάθος καταχώρηση είναι ', Πλ_μαθ
Τέλος Βαθμοί_Απολυτηρίων

21

Αλγόριθμος Βαθμοί_μαθητή
!Αρχικοποίηση της μεταβλητής που θα υπολογίζει το πλήθος των μαθημάτων στα οποία ο
!μαθητής έχει πάρει 20.
Πλ_20 <-- 0
!Διαβάζεται ο πρώτος βαθμός εκτός βρόχου, γιατί μπορεί να είναι εκτός ορίων βαθμολογικής κλίμακας,
!οπότε δεν θα πρέπει να μπει ο αλγόριθμος μέσα στο βρόχο.
Διάβασε βαθμ
Όσο βαθμ >= 0 και βαθμ <= 20 επανάλαβε
 !Ελέγχεται αν ο βαθμός είναι 20 ώστε να αυξηθεί το πλήθος.
 Αν βαθμ = 20 τότε
 Πλ_20 <-- Πλ_20 + 1
 Τέλος_αν
 !Διαβάζεται ο επόμενος βαθμός.
 Διάβασε βαθμ
Τέλος_επανάληψης
Εμφάνισε 'Το πλήθος των μαθημάτων στα οποία ο μαθητής πήρε 20 είναι ', Πλ_20
Τέλος Βαθμοί_μαθητή

22

Αλγόριθμος Βαθμοί_μαθητών
!Αρχικοποίηση της μεταβλητής που θα μετράει το πλήθος των μαθητών που έχουν βαθμό πάνω από 18.
Πλ_18 <-- 0
!Αρχικοποίηση της μεταβλητής που θα μετράει το σύνολο των στοιχείων των μαθητών που καταχωρήθηκαν.
Πλ_μαθ <-- 0
!Το πρώτο ονοματεπώνυμο διαβάζεται εκτός δομής επανάληψης γιατί μπορεί να είναι το κενό, οπότε και
!ο βρόχος δεν θα πρέπει να εκτελεστεί.
Εμφάνισε 'Δώσε ονοματεπώνυμο: '
Διάβασε ον
Όσο ον <> '' επανάλαβε
 !Γίνεται έλεγχος εγκυρότητας βαθμολογίας 2 φορές. Μια φορά για τον προφορικό και μια για τον
 !γραπτό βαθμό.
 Εμφάνισε 'Δώσε προφορικό βαθμό: '
 Διάβασε βαθμ_προφ
 Όσο βαθμ_προφ < 0 ή βαθμ_προφ > 20 επανάλαβε

```
Εμφάνισε "Έδωσες λάθος προφορικό βαθμό. Δώσε πάλι: "  
  Διάβασε βαθμ_προφ  
Τέλος_επανάληψης  
Εμφάνισε 'Δώσε γραπτό βαθμό: '  
  Διάβασε βαθμ_γρ  
  Όσο βαθμ_γρ < 0 ή βαθμ_γρ > 20 επανάλαβε  
    Εμφάνισε "Έδωσες λάθος γραπτό βαθμό. Δώσε πάλι: "  
      Διάβασε βαθμ_γρ  
    Τέλος_επανάληψης  
    Τελικός_βαθμ <-- 0.3 * βαθμ_προφ + 0.7 * βαθμ_γρ  
    Εμφάνισε "Το ονοματεπώνυμο του μαθητή είναι ', ον  
    Εμφάνισε "Ο τελικός βαθμός του είναι ', Τελικός_βαθμ  
    Αν Τελικός_βαθμ > 18 τότε  
      Πλ_18 <-- Πλ_18 + 1  
    Τέλος_αν  
    !Αυξάνεται το πλήθος των των μαθητών για τους οποίους καταχωρήθηκαν στοιχεία.  
    Πλ_μαθ <-- Πλ_μαθ + 1  
    !Διαβάζεται και πάλι το ονοματεπώνυμο.  
    Εμφάνισε 'Δώσε ονοματεπώνυμο: '  
    Διάβασε ον  
Τέλος_επανάληψης  
!Ελέγχεται πρώτα αν το πλήθος των μαθητών για τους οποίους δόθηκαν στοιχεία είναι διαφορετικό από μηδέν.  
!Αυτό χρειάζεται γιατί δεν πρέπει να γίνει διαίρεση με μηδέν.  
Αν Πλ_μαθ <> 0 τότε  
  Ποσοστό <-- (Πλ_18 * 100) / Πλ_μαθ  
  Εμφάνισε "Το ποσοστό των μαθητών που έχουν βαθμό μεγαλύτερο από 18 είναι ', Ποσοστό, %"  
Αλλιώς  
  Εμφάνισε 'Δεν δόθηκαν στοιχεία μαθητών'  
Τέλος_αν  
Τέλος Βαθμοί_μαθητών
```