

ΛΥΜΕΝΕΣ ΑΣΚΗΣΕΙΣ ΑΕΠΠ
(ΕΠΙΛΟΓΗ – ΕΠΑΝΑΛΗΨΗ)

Για την εκτόξευση ενός πυραύλου χρησιμοποιείται ένα ρολόι το οποίο αναγράφει το χρόνο που απομένει για την εκτόξευση σε Ώρες : Λεπτά. Ο αλγόριθμος διαβάσει τις τιμές της ώρας και των λεπτών και υπολογίζει πόσα λεπτά απομένουν για την εκτόξευση του πυραύλου. Αν απομένουν λιγότερα από 90 λεπτά, εμφανίζει το μήνυμα «Εκτόξευση σε » και τα λεπτά που απομένουν

```
Αλγόριθμος Ασκ_1
Διάβασε hour, min
M <- hour * 60 + min
Αν M < 90 τότε
    Εμφάνισε "Εκτόξευση σε", M, "λεπτά"
Τέλος_Αν
Τέλος Ασκ_1
```

Ο υπάλληλος μιας εταιρίας πληρώνεται κάθε εβδομάδα προς 25 € την ώρα. Αν οι ώρες εβδομαδιαίας εργασίας είναι περισσότερες από 40, τότε κάθε ώρα πέραν των 40 πληρώνεται επιπλέον με 5 €. Ο αλγόριθμος διαβάσει το σύνολο των ωρών μιας εβδομάδας και υπολογίζει και εμφανίζει το σύνολο των εβδομαδιαίων αποδοχών του υπαλλήλου

```
Αλγόριθμος Ασκ_2
Διάβασε h
Αν h ≤ 40 τότε
    Αποδ <- h * 25
αλλιώς
    Αποδ <- 40 * 25 + (h - 40) * 30
Τέλος_αν
Εμφάνισε Αποδ
Τέλος Ασκ_2
```

Αλγόριθμος ο οποίος διαβάσει τρεις ακεραίους αριθμούς και αν η διαφορά του δεύτερου από τον πρώτο είναι μεγαλύτερη από την διαφορά τρίτου από το δεύτερο, τότε εναλλάσσει τις τιμές του πρώτου και του δεύτερου και τις εμφανίζει

```
Αλγόριθμος Ασκ_3
Διάβασε α, β, γ
Αν (α - β) > (β - γ) τότε
    X <- α
    α <- β
    β <- X
    Εμφάνισε α, β
Τέλος_αν
Τέλος Ασκ_3
```

Σε ένα συνδρομητικό κανάλι η μηνιαία συνδρομή είναι 30 € δρχ και η χρέωση ανά ταινία δίνεται από τον κάτωθι πίνακα. Ο αλγόριθμος διαβάσει το όνομα του συνδρομητή και τον αριθμό των ταινιών που είδε σε ένα μήνα. Υπολογίζει και εμφανίζει το ποσό πληρωμής για τον συνδρομητή ακολουθούμενο από το όνομά του. Θεωρείστε ότι η χρέωση των ταινιών είναι κλιμακωτή.

Αριθμός ταινιών	Χρέωση / ταινία (€)
1 - 10	1,4
11 - 30	1,1
31 και άνω	0,70

Αλγόριθμος Ασκ_4

```
Διάβασε name, tapes
Αν (tapes ≥ 1) και (tapes ≤ 10) τότε
    cost <- tapes * 1,4
αλλιώς_αν (tapes ≥ 11) και (tapes ≤ 30) τότε
    cost <- 10 * 1,4 + (tapes - 10) * 1,1
αλλιώς_αν (tapes ≥ 31) τότε
    cost <- 10 * 1,4 + 20 * 1,1 + (tapes - 30) * 0,7
Τέλος_αν
ΠΟΣΟ <- cost + 30
Εμφάνισε "Ποσό πληρωμής", ΠΟΣΟ, "για τον συνδρομητή", name
Τέλος Ασκ_4
```

Να γραφεί αλγόριθμος που βρίσκει και τυπώνει τις ρίζες μιας δευτεροβάθμιας εξίσωσης $ax^2 + bx + c = 0$ ως εξής:

(i) υπολογίζει την διακρίνουσα $\Delta = \beta^2 - 4\alpha\gamma$

(ii) Σε περίπτωση που $\alpha \neq 0$ ελέγχει την Δ κι αν $\Delta < 0$ εμφανίζει μήνυμα «δεν υπάρχουν πραγματικές ρίζες», αν $\Delta = 0$ τότε υπολογίζει και εμφανίζει μια διπλή ρίζα $-\beta/2\alpha$, ενώ αν $\Delta > 0$ υπολογίζει και εμφανίζει τις δύο ρίζες $(-\beta \pm \sqrt{\Delta})/2\alpha$

(iii) Σε περίπτωση που $\alpha = 0$ τότε ελέγχει τα β και γ και αν $\beta \neq 0$ τότε υπολογίζει και εμφανίζει τη ρίζα $-\gamma/\beta$, αν $\beta = \gamma = 0$ τότε εμφανίζει το μήνυμα «αόριστη» ενώ αν $\beta = 0$ και $\gamma \neq 0$ τότε εμφανίζει το μήνυμα «αδύνατη»

```
Αλγόριθμος Ασκ_5
Διάβασε α, β, γ
Δ <- β ^ 2 - 4 * α * γ
Αν α ≠ 0 τότε
    Αν Δ > 0 τότε
        Χ1 <- (-β + Δ ^ (1/2)) / (2 * α)
        Χ2 <- (-β - Δ ^ (1/2)) / (2 * α)
        Εμφάνισε Χ1, Χ2
    αλλιώς_αν Δ = 0 τότε
        Χ <- -β / (2 * α)
        Εμφάνισε Χ
    αλλιώς
        Εμφάνισε "Δεν υπάρχουν πραγματικές ρίζες"
Τέλος_αν
αλλιώς
    Αν β ≠ 0 τότε
        Χ <- -γ / β
        Εμφάνισε Χ
    αλλιώς_αν γ = 0 τότε
        Εμφάνισε "αόριστη"
    αλλιώς
        Εμφάνισε "αδύνατη"
Τέλος_αν
Τέλος_αν
Τέλος Ασκ_5
```

Ο τελικός βαθμός για ένα μάθημα ισούται με το άθροισμα του 70% της γραπτής βαθμολογίας και του 30% της προφορικής, αφού όμως πρώτα η προφορική προσαρμοστεί έτσι ώστε η διαφορά της από την γραπτή να μην ξεπερνά τις 3 μονάδες. Να γραφεί αλγόριθμος που διαβάσει τις δύο βαθμολογίες και υπολογίζει και εμφανίζει την τελική. Να θεωρηθεί ότι όλες οι βαθμολογίες βρίσκονται στην 20βάθμια κλίμακα

```
Αλγόριθμος Ασκ_6
Διάβασε πρ, γρ
Αν (γρ - πρ > 3) τότε
    πρ <- γρ - 3
αλλιώς_αν (πρ - γρ > 3) τότε
    πρ <- γρ + 3
Τέλος_αν
ΤΕΛ <- 0,3 * πρ + 0,7 * γρ
Εμφάνισε "Τελική βαθμολογία", ΤΕΛ
Τέλος Ασκ_6
```

Σε μια πολυκατοικία πωλούνται διαμερίσματα με την παρακάτω πολιτική. Στο ισόγειο η τιμή είναι 1000 € ανά τμ, ενώ για κάθε όροφο η τιμή ανά τμ αυξάνει κατά 30 € (π.χ. στον 1ο όροφο είναι 1030 €, στον 2ο όροφο είναι 1060 € κλπ). Η ελάχιστη τιμή (αξία) που μπορεί να έχει ένα διαμέρισμα είναι 30,000 €. Ο φόρος μεταβίβασης είναι 5% και τα μεσιτικά έξοδα είναι 500 €. Ο αλγόριθμος:

Α) διαβάσει σε ποιο όροφο και πόσα τετραγωνικά μέτρα είναι ένα διαμέρισμα

Β) υπολογίζει και εκτυπώνει την αξία του διαμερίσματος

Γ) υπολογίζει και εκτυπώνει το συνολικό κόστος κτήσης του διαμερίσματος

```
Αλγόριθμος Ασκ_7
Διάβασε ΤΜ, ΟΡ
Αξία <- (1000 + ΟΡ * 30) * ΤΜ
Αν (Αξία < 30000) τότε
    Αξία <- 30000
Τέλος_Αν
Εκτύπωσε "Η αξία του διαμερίσματος είναι", Αξία
Κόστος <- Αξία + 0,05 * Αξία + 500
Εκτύπωσε "Κόστος κτήσης είναι", Κόστος
Τέλος Ασκ_7
```

Να γραφεί αλγόριθμος που διαβάσει τη χρονολογία που συσκευάστηκε ένα προϊόν π.χ 28-06-2005 και την χρονολογία που έχουμε σήμερα και τυπώνει μήνυμα σχετικό με το αν μπορεί να καταναλωθεί αυτό το προϊόν ή είναι ληγμένο.

Το προϊόν θεωρείται ληγμένο αν έχουν περάσει τουλάχιστον 1000 μέρες από τη συσκευασία του. Να θεωρηθεί ότι όλοι οι μήνες έχουν 30 μέρες και ότι κανένα έτος δεν είναι δίσεκτο

```
Αλγόριθμος Ασκ_8
Εμφάνισε "Δώσε χρονολογία συσκευασίας"
Διάβασε day,month,year
Εμφάνισε "Δώσε σημερινή χρονολογία"
Διάβασε d,m,y
MERES_1 <-- year*12*30 + month*30 + day
MERES_2 <-- y*12*30 + m*30 + d
Αν MERES_2 – MERES_1 > 1000 τότε
    Εμφάνισε "Κονσέρβα ληγμένη"
Αλλιώς
    Εμφάνισε "Κονσέρβα προς κατανάλωση"
Τέλος_αν
Τέλος Ασκ_8
```

Να γράψετε πρόγραμμα σε ΓΛΩΣΣΑ που να διαβάσει από το πληκτρολόγιο τιμή για ένα έτος, και αποφασίζει αν το έτος που διάβασε είναι δίσεκτο ή όχι. Δίσεκτο είναι ένα έτος που είναι πολλαπλάσιο του 400 ή είναι πολλαπλάσιο του 4 και όχι του 100

```
ΠΡΟΓΡΑΜΜΑ Ασκ_9
ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ: etos
ΑΡΧΗ
    ΓΡΑΨΕ "Δώσε το έτος"
    ΔΙΑΒΑΣΕ etos
    ΑΝ ((etos mod 4 = 0) ΚΑΙ (etos mod 100 <> 0)) Ή (etos mod 400 = 0) ΤΟΤΕ
        ΓΡΑΨΕ "Έτος δίσεκτο"
    ΑΛΛΙΩΣ
        ΓΡΑΨΕ "Έτος όχι δίσεκτο"
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```

Να γράψετε πρόγραμμα σε ΓΛΩΣΣΑ το οποίο διαβάσει μια ημερομηνία υπό τη μορφή Μέρα / Μήνας / Έτος (3 ακέραιοι αριθμοί) και την ελέγχει αν αυτή είναι σωστή εμφανίζοντας ανάλογο μήνυμα. Να θεωρηθεί ότι :

(i) οι μήνες Ιανουάριος, Μάρτιος, Μάιος, Ιούλιος, Αύγουστος, Οκτώβριος και Δεκέμβριος έχουν 31 ημέρες

(ii) οι μήνες Απρίλιος, Ιούνιος, Σεπτέμβριος και Νοέμβριος έχουν 30 ημέρες

(iii) ο μήνας Φεβρουάριος έχει 28 ημέρες εκτός αν το έτος είναι δίσεκτο οπότε και έχει 29 ημέρες

```
ΠΡΟΓΡΑΜΜΑ Ασκ_10
ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ: d, m, y, meres_minia
    ΛΟΓΙΚΕΣ: OK
ΑΡΧΗ
    ΔΙΑΒΑΣΕ d, m, y
    OK <-- Αληθής
    ΑΝ (d < 1) Ή (d > 31) Ή (m < 1) Ή (m > 12) Ή (y < 0) ΤΟΤΕ
        OK <-- Ψευδής
        ΓΡΑΨΕ "Λάθος, τιμή εκτός ορίων"
    ΤΕΛΟΣ_ΑΝ
    ΑΝ (m = 1) Ή (m = 3) Ή (m = 5) Ή (m = 7) Ή (m = 8) Ή (m = 10) Ή (m = 12) ΤΟΤΕ
        meres_minia <-- 31
    ΑΛΛΙΩΣ_ΑΝ (m = 4) Ή (m = 6) Ή (m = 9) Ή (m = 11) ΤΟΤΕ
        meres_minia <-- 30
    ΑΛΛΙΩΣ_ΑΝ (m = 2) ΤΟΤΕ
        ΑΝ (y mod 400 = 0) Ή ((y mod 4 = 0) ΚΑΙ (y mod 100 <> 0)) ΤΟΤΕ
            meres_minia <-- 29
        ΑΛΛΙΩΣ
            meres_minia <-- 28
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΑΝ
ΑΝ meres_minia < d ΤΟΤΕ
    OK <-- Ψευδής
    ΓΡΑΨΕ "Λάθος, ο μήνας αυτός δεν έχει τόσες μέρες"
ΤΕΛΟΣ_ΑΝ
ΑΝ OK = Αληθής ΤΟΤΕ
```

ΓΡΑΨΕ "Ημερομηνία σωστή"
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Αλγόριθμος που διαβάσει ένα ακέραιο αριθμό και διασφαλίζει ότι είναι θετικός. Αν ο αριθμός είναι περιττός τότε εμφανίζει όλους τους θετικούς περιττούς αριθμούς που προηγούνται από αυτόν, ενώ αν είναι άρτιος εμφανίζει όλους τους θετικούς άρτιους αριθμούς που προηγούνται από αυτόν.

```
Αλγόριθμος Ασκ_1
Αρχή_Επανάληψης ! Διασφάλιση τιμής
  Διάβασε num
  Μέχρις_ότου (num > 0)
  Αν (num mod 2 = 1) τότε
    Για i από 1 μέχρι num-2 με_βήμα 2
      Εμφάνισε i
    Τέλος_επανάληψης
  αλλιώς
    Για i από 2 μέχρι num-2 με_βήμα 2
      Εμφάνισε i
    Τέλος_επανάληψης
Τέλος_αν
Τέλος Ασκ_1
```

Να γραφεί αλγόριθμος ο οποίος διαβάσει συνεχώς αριθμούς και σταματάει όταν δοθεί η τιμή 555. Κατόπιν εμφανίζει πόσοι από αυτούς ήταν άρτιοι, πόσοι περιττοί, πόσοι αρνητικοί, πόσοι θετικοί και πόσοι μηδέν. Ο τερματικός αριθμός 555 (τιμή φρουρός) δεν λαμβάνεται υπόψη.

```
Αλγόριθμος Ασκ_2
αρτ <-- 0
περ <-- 0
θετ <-- 0
αρν <-- 0
μηδ <-- 0
Διάβασε X
Όσο (X ≠ 555) επανάλαβε
  Αν (X mod 2 = 0) τότε
    αρτ <-- αρτ + 1
  αλλιώς
    περ <-- περ + 1
  Τέλος_αν
  Αν X > 0 τότε
    θετ <-- θετ + 1
  αλλιώς_αν X < 0 τότε
    αρν <-- αρν + 1
  αλλιώς
    μηδ <-- μηδ + 1
  Τέλος_αν
  Διάβασε X
Τέλος_επανάληψης
Εμφάνισε αρτ, περ, θετ, αρν, μηδ
Τέλος Ασκ_2
```

Να γραφεί αλγόριθμος ο οποίος εμφανίζει όλους τους τριψήφιους αριθμούς από το 100 μέχρι το 999 που το άθροισμα των ψηφίων τους ισούται με 6 π.χ. οι αριθμοί 600, 213, 501 κλπ.

```
Αλγόριθμος Ασκ_3
Για i από 100 μέχρι 999
  A1 <-- i div 100
  A2 <-- i mod 100 div 10
  A3 <-- i mod 100 mod 10 !αλλιώς A3 <-- i mod 10
  S <-- A1 + A2 + A3
  Αν S = 6 τότε Εμφάνισε i
Τέλος_επανάληψης
Τέλος Ασκ_3
```

Να γραφεί αλγόριθμος ο οποίος θα διαβάσει τα ονόματα κάποιων μαθητών καθώς και τους βαθμούς τους σε τρία μαθήματα. Κατόπιν υπολογίζει το μέσο όρο βαθμολογίας τους και τον εμφανίζει ακολουθούμενο από το όνομά τους. Ο αλγόριθμος τερματίζει την επαναληπτική διαδικασία όταν για όνομα μαθητή διαβάσει τον κενό χαρακτήρα

```

Αλγόριθμος Ασκ_4
Διάβασε name
Όσο (name ≠ " ") επανάλαβε
    Διάβασε β1, β2, β3
    M <- (β1 + β2 + β3) / 3
    Εμφάνισε M, name
    Διάβασε name
Τέλος_επανάληψης
Τέλος Ασκ_4

```

Εστω η συνάρτηση $f(x) = 1 / (x-3)$. Να γραφεί αλγόριθμος ο οποίος εμφανίζει την τιμή της συνάρτησης για κάθε ακέραια τιμή του x στο διάστημα $[-5,5]$. Για την περίπτωση που μηδενίζεται ο παρανομαστής να εμφανίζει το μήνυμα «Διαίρεση με το 0»

```

Αλγόριθμος Ασκ_5
Για x από -5 μέχρι 5
    Αν x ≠ 3 τότε
        F <- 1 / (x-3)
        Εμφάνισε F
    αλλιώς
        Εμφάνισε "Διαίρεση με το 0"
Τέλος_αν
Τέλος_επανάληψης
Τέλος Ασκ_5

```

Ένα φορτηγό μεταφορών καλύπτει καθημερινά έναν αριθμό km. Για τα πρώτα 30 km που καλύπτει, το κόστος του είναι 35€. Ενώ για κάθε επιπλέον km η αξία φθάνει στα 0,50 €/km. Ο αλγόριθμος
 Α) διαβάσει τον αριθμό των ημερών που δούλεψε το φορτηγό καθώς και τα km που καθημερινά κάλυψε
 Β) υπολογίζει και εμφανίζει το ημερήσιο κόστος μεταφοράς βάσει km
 Γ) υπολογίζει και εμφανίζει το συνολικό κόστος όλων των ημερών

```

Αλγόριθμος Ασκ_6
Διάβασε days
sum <- 0
Για i από 1 μέχρι days
    Διάβασε km
    Αν km ≤ 30 τότε
        cost <- 35
    αλλιώς
        cost <- 35 + (km-30) * 0,50
    Τέλος_αν
    Εμφάνισε "Ημερήσιο κόστος", cost
    sum <- sum + cost
Τέλος_επανάληψης
Εμφάνισε "Συνολικό κόστος", sum
Τέλος Ασκ_6

```

Να γραφεί αλγόριθμος που διαβάσει ένα άγνωστο πλήθος αριθμών μέχρι να διαβαστούν 10 αρνητικοί. Επίσης υπολογίζει και τυπώνει το άθροισμα των αριθμών που ήταν θετικοί

```

Αλγόριθμος Ασκ_7
Σ <- 0
neg <- 0
Όσο neg < 10 επανάλαβε
    Εμφάνισε "Δώσε ένα αριθμό"
    Διάβασε X
    Αν X > 0 τότε
        Σ <- Σ + X
    αλλιώς_αν X < 0 τότε
        neg <- neg + 1
    Τέλος_Αν
Τέλος_επανάληψης
Εκτύπωσε "Άθροισμα θετικών:", Σ
Τέλος Ασκ_7

```

Να γραφεί αλγόριθμος που δέχεται κάποιον αριθμό και υπολογίζει και εμφανίζει το πλήθος των ψηφίων του π.χ αν δεχτεί τον αριθμό 5567 θα εμφανίζει 4

```

Αλγόριθμος Ασκ_8
Αρχή_επανάληψης

```

```

Διάβασε num
Μέχρις_ότου (num > 0)
pl <-- 0
Όσο (num ≠ 0) επανάλαβε
    num <-- num div 10
    pl <-- pl + 1
Τέλος_επανάληψης
Εμφάνισε "Πλήθος ψηφίων αριθμού", pl
Τέλος Ασκ_8

```

Να γραφεί αλγόριθμος που υπολογίζει το ημίτονο και το συνημίτονο ενός αριθμού με βάση τις παρακάτω σχέσεις

$$\eta\mu x = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5 - \frac{1}{7!}x^7 + \dots + \frac{1}{29!}x^{29}$$

$$\sigma\upsilon\nu x = 1 - \frac{1}{2!}x^2 + \frac{1}{4!}x^4 - \frac{1}{6!}x^6 + \dots + \frac{1}{20!}x^{20}$$

```

Αλγόριθμος Ασκ_9
Διάβασε x
proximo <-- 1
ημX <-- 0
Για i από 1 μέχρι 29 με_βήμα 2
    paragodiko <-- 1
    Για k από 1 μέχρι i
        paragodiko <-- paragodiko * k
    Τέλος_επανάληψης
    ημX <-- ημX + proximo * x^i / paragodiko
    proximo <-- proximo * (-1)
Τέλος_επανάληψης
συνX <-- 1
proximo <-- -1
Για i από 2 μέχρι 20 με_βήμα 2
    paragodiko <-- 1
    Για k από 1 μέχρι i
        paragodiko <-- paragodiko * k
    Τέλος_επανάληψης
    συνX <-- συνX + proximo * x^i / paragodiko
    proximo <-- proximo * (-1)
Τέλος_επανάληψης
Αποτελέσματα //ημX, συνX//
Τέλος Ασκ_9

```

Ένας μαθητής ζήτησε από τους γονείς του να του αγοράσουν ένα υπολογιστικό σύστημα αξίας 1450 €. Οι γονείς του δήλωσαν ότι μπορούν να του διαθέσουν σταδιακά το ποσό, δίνοντας του κάθε εβδομάδα ποσό διπλάσιο από την προηγούμενη, αρχίζοντας την πρώτη εβδομάδα με ποσό 15 €. Να γραφεί αλγόριθμος που
 Α) να υπολογίζει μετά από πόσες εβδομάδες ο μαθητής θα μπορέσει να αγοράσει το υπολογιστικό σύστημα
 Β) να υπολογίζει και να ελέγχει πιθανό περίσσειμα χρημάτων

```

Αλγόριθμος Ασκ_1
ποσό <-- 0
χαρτζηλίκι <-- 15
week <-- 0
Όσο (ποσό < 1450) επανάλαβε
    ποσό <-- ποσό + χαρτζηλίκι
    χαρτζηλίκι <-- χαρτζηλίκι * 2
    week <-- week + 1
Τέλος_επανάληψης
Εμφάνισε "Αγορά σε ", week, "εβδομάδες"
Αν ποσό > cost τότε
    Εμφάνισε "Περίσσειμα: ", ποσό-1450
Τέλος_αν
Τέλος Ασκ_1

```

Σύμφωνα με την παράδοση η δημιουργία του σκακιού οφείλεται στην προτροπή ενός βασιλιά να ανταμείψει όποιον από τους συμβούλους του θα έφτιαχνε ένα πρωτότυπο παιχνίδι, διαφορετικό κάθε φορά στην εξέλιξή του. Ένας λοιπόν από τους συμβούλους του σκέφτηκε να δημιουργήσει το σκάκι. Ο βασιλιάς ενθουσιάστηκε και ρώτησε για την αμοιβή του συμβούλου. Αυτός ζήτησε να πάρει το ρύζι που θα μαζευόταν ως εξής: Στο 1ο

τετράγωνο του σκακιού να τοποθετηθεί 1 κόκκος ρυζιού, στο 2ο τετράγωνο 2 κόκκοι ρυζιού, στο 3ο τετράγωνο 4 κόκκοι, στο 4ο τετράγωνο 8 κόκκοι κλπ. Το άθροισμα των κόκκων θα ήταν η αμοιβή του.

Να γραφεί αλγόριθμος που θα υπολογίζει πόσοι τόνοι ρυζιού θα ήταν η ποσότητα αυτή, αν θεωρηθεί ότι 1 Kg ρυζιού έχει 10000 κόκκους

```
Αλγόριθμος Ασκ_2
κόκκοι <-- 1
P <-- 1
Για i από 2 μέχρι 64
    P <-- P * 2
    κόκκοι <-- κόκκοι + P
Τέλος_επανάληψης
τόνοι <-- (κόκκοι/10000)/1000
Εμφάνισε "Σύνολο τόνων ρυζιού", τόνοι
Τέλος Ασκ_2
```

Να γραφεί αλγόριθμος που διαβάζει έναν ακέραιο αριθμό διασφαλίζοντας ότι είναι μεγαλύτερος του 4 και εμφανίζει αν αυτός είναι πρώτος ή σύνθετος (πρώτος είναι ο αριθμός που διαιρείται μόνο με τον εαυτό του και τη μονάδα)

```
Αλγόριθμος Ασκ_3
Αρχή_επανάληψης
    Διάβασε N
Μέχρις_ότου N > 4
    protos <-- Αληθής
    Για i από 2 μέχρι (N div 2)
        Αν N mod i = 0 τότε
            protos <-- Ψευδής
    Τέλος_Αν
Τέλος_επανάληψης
Αν protos = Αληθής τότε
    Εμφάνισε "Ο αριθμός", N, "είναι πρώτος"
αλλιώς
    Εμφάνισε "Ο αριθμός", N, "είναι σύνθετος"
Τέλος_Αν
Τέλος Ασκ_3
```

Τέλειος θεωρείται ένας ακέραιος αριθμός του οποίου οι παράγοντες έχουν σαν άθροισμα τον ίδιο τον αριθμό. Στους παράγοντες συμπεριλαμβάνεται το 1 αλλά όχι ο ίδιος ο αριθμός (γνήσιοι διαιρέτες), π.χ. τέλειοι αριθμοί είναι ο 6 ($1 + 2 + 3 = 6$) και ο 28 ($1 + 2 + 4 + 7 + 14 = 28$). Να βρεθούν και να εκτυπωθούν οι 10 πρώτοι τέλειοι θετικοί ακέραιοι

```
Αλγόριθμος Ασκ_4
plithos <-- 0
number <-- 1
Όσο (plithos < 10) επανάλαβε
    sum <-- 0
    Για i από 1 μέχρι (number div 2)
        Αν number mod i = 0 τότε
            sum <-- sum + i
    Τέλος_αν
    Τέλος_επανάληψης
    Αν sum = number τότε
        Εμφάνισε "Ο αριθμός", number, "είναι τέλειος"
    plithos <-- plithos + 1
    Τέλος_αν
    number <-- number + 1
Τέλος_επανάληψης
Τέλος Ασκ_4
```

Ένα Video Club προσφέρει δύο διαφορετικούς τρόπους ενοικίασης κασετών: Ο 1ος προβλέπει εγγραφή 30€ και ενοικίαση 1,20€ για κάθε κασέτα ενώ ο 2ος δεν προβλέπει εγγραφή αλλά ενοικιάζει κάθε κασέτα προς 1,90€ .

Έστω ότι κάποιος πελάτης ενοικιάζει X κασέτες. Ο αλγόριθμος

A) υπολογίζει και εκτυπώνει ποιος από τους δύο τρόπους ενοικίασης συμφέρει τον πελάτη

B) αν τον συμφέρει ο δεύτερος τρόπος, υπολογίζει τον αριθμό των κασετών που μπορεί να νοικιάσει ακόμα, ώστε ο δεύτερος τρόπος να παραμένει η πιο συμφέρουσα λύση

```
Αλγόριθμος Ασκ_5
Διάβασε X
cost1 <-- 30 + 1,20 * X
```

```

cost2 ← 1,90 * Χ
Αν cost1 < cost2 τότε
    Εκτύπωσε "Συμφέρει ο πρώτος"
αλλιώς
    Εκτύπωσε "Συμφέρει ο δεύτερος"
    διαφορά ← cost1 – cost2
    kasetes ← 0
    Όσο (διαφορά ≥ 1,90) επανάλαβε
        διαφορά ← διαφορά – 1,90
        kasetes ← kasetes + 1
    Τέλος_επανάληψης
    Εκτύπωσε "Μπορεί να νοικιάσει ακόμα", kasetes
Τέλος_Αν
Τέλος Ασκ_5

```

Να γραφεί αλγόριθμος που διαβάσει ένα κωδικό πρόσβασης (password) σε ένα σύστημα καθώς και ένα κωδικό αριθμό (code) που δίνει ο χρήστης ώστε να έχει πρόσβαση στο σύστημα. Αν δοθεί τρεις φορές λάθος κωδικός, τότε θα αποκλείει την είσοδο με ανάλογο μήνυμα, διαφορετικά θα εμφανίζει μήνυμα επιτυχούς πρόσβασης στο σύστημα.

```

Αλγόριθμος Ασκ_6
Διάβασε password
done ← ψευδής
i ← 1
Όσο (i ≤ 3) και (done = ψευδής) επανάλαβε
    Διάβασε code
    Αν password = code τότε
        done ← αληθής
    αλλιώς
        i ← i + 1
    Τέλος_αν
Τέλος_επανάληψης
Αν done = αληθής τότε
    Εμφάνισε "Επιτυχής πρόσβαση"
αλλιώς
    Εμφάνισε "Ανεπιτυχής προσπάθεια εισόδου"
Τέλος_Αν
Τέλος Ασκ_6

```

Μια εταιρία υπηρεσιών Internet παρέχει δωρεάν λογαριασμό e-mail με αποθηκευτικό χώρο 500 MB. Να γραφεί αλγόριθμος ο οποίος θα διαβάσει το μέγεθος κάθε μηνύματος σε MB που εισέρχεται στο mailbox του χρήστη. Αν το συνολικό μέγεθος των μηνυμάτων είναι μεταξύ 420 MB και 480 MB θα εκτυπώνει το μήνυμα «Προσοχή, λίγος χώρος» και θα συνεχίζει να δέχεται μηνύματα. Αν το συνολικό μέγεθος των μηνυμάτων ξεπεράσει το όριο των 500 MB τότε θα εκτυπώνει το πλήθος των μηνυμάτων που έχουν αποθηκευτεί καθώς και το μήνυμα «Υπέρβαση ορίων» ακολουθούμενο από το πλήθος των MB που θα διαγραφούν ώστε να είναι στα 500 MB το μέγεθος του mailbox

```

Αλγόριθμος Ασκ_7
Σ ← 0
pl ← 0
Όσο Σ ≤ 500 επανάλαβε
    Διάβασε mail
    Σ ← Σ + mail
    pl ← pl + 1
    Αν (Σ ≥ 420) και (Σ ≤ 480) τότε
        Εκτύπωσε "Προσοχή,σας μένει λίγος χώρος"
    αλλιώς_αν (Σ > 500) τότε
        pl ← pl - 1
        Εκτύπωσε "Αποθηκευόμενα μηνύματα",pl
        Εκτύπωσε "Υπέρβαση ορίων"
        Εκτύπωσε "Θα διαγραφούν", Σ-500, "MB"
    Τέλος_αν
Τέλος_επανάληψης
Τέλος Ασκ_7

```

Ένας καταναλωτής έχοντας στη τσέπη του 5000 € ξεκινάει να αγοράσει διάφορα προϊόντα και ταυτόχρονα κρατάει το συνολικό ποσό στο οποίο έχει φτάσει κάθε φορά που αγοράζει ένα είδος. Οι τιμές των προϊόντων που αγοράζει είναι σε δραχμές και είναι γνωστό ότι 1 € ισούται με 340,75 δρχ.

Να γραφεί αλγόριθμος για τον υπολογισμό του συνολικού ποσού που θα πληρώσει ο καταναλωτής για τα προϊόντα που αγόρασε και να τον ειδοποιεί να σταματήσει τις αγορές όταν το προϊόν που έχει επιλέξει να αγοράσει είναι πιο ακριβό από τα χρήματα που του έχουν απομείνει

```
Αλγόριθμος Ασκ_8
ΑΡΧΙΚΟ <-- 5000 * 340,75
αγορές <-- 0
Διάβασε ΕΙΔΟΣ
Όσο (αγορές + ΕΙΔΟΣ ≤ ΑΡΧΙΚΟ) επανάλαβε
  αγορές <-- αγορές + ΕΙΔΟΣ
  Διάβασε ΕΙΔΟΣ
Τέλος_επανάληψης
Εμφάνισε "Σταμάτησε τις αγορές"
Εμφάνισε "Συνολικό ποσό αγορών",αγορές
Τέλος Ασκ_8
```

Ένας υπάλληλος διανομής διαφημιστικών εντύπων θέλει να διανείμει 800 διαφημιστικά έντυπα στα γραμματοκιβώτια κατοικιών. Αν το πλήθος των διαμερισμάτων μιας κατοικίας που επισκέπτεται είναι μέχρι και 6, τότε αφήνει στο γραμματοκιβώτιο ένα διαφημιστικό έντυπο για κάθε διαμέρισμα. Αν το πλήθος των διαμερισμάτων είναι περισσότερα από 6, τότε αφήνει συνολικά 8 έντυπα. Να γραφεί αλγόριθμος που

A) για κάθε κατοικία που επισκέπτεται ο υπάλληλος διαβάζει τον αριθμό των διαμερισμάτων της και

B) εμφανίζει τον αριθμό των κατοικιών που επισκέφτηκε μέχρι να μοιράσει όλα τα φυλλάδια

```
Αλγόριθμος Ασκ_9
κατ <-- 0
Φυλλάδια <-- 800
Όσο Φυλλάδια > 0 επανάλαβε
  Διάβασε διαμ
  Αν διαμ ≤ 6 τότε
    F <-- διαμ
  αλλιώς
    F <-- 8
  Τέλος_Αν
  Αν Φυλλάδια ≥ F τότε
    Φυλλάδια <-- Φυλλάδια - F
  αλλιώς
    Φυλλάδια <-- 0
  Τέλος_Αν
κατ <-- κατ + 1
Τέλος_επανάληψης
Εμφάνισε "Αριθμός κατοικιών που επισκέφτηκε", κατ
Τέλος Ασκ_9
```

Σε ένα ραδιοφωνικό σταθμό γίνεται ένας διαγωνισμός για να μοιραστούν δώρα συνολικής αξίας 2000 €. Κάθε ακροατής που διαγωνίζεται απάντά τηλεφωνικά σε 4 ερωτήσεις που η καθεμία μπορεί να πιάσει βαθμό από 0 μέχρι και 12. Ο διαγωνιζόμενος κερδίζει αν έχει συνολικό βαθμό και από τις 4 ερωτήσεις τουλάχιστον 40. Η αξία του δώρου που θα πάρει είναι ίση με το τριπλάσιο της συνολικής του βαθμολογίας σε €. Να γραφεί αλγόριθμος που

A) διαβάζει το όνομα κάθε διαγωνιζόμενου και την βαθμολογία του σε καθεμιά από τις 4 ερωτήσεις

B) ελέγχει για κάθε διαγωνιζόμενο αν δικαιούται δώρο μέχρι να μοιραστεί όλο το αρχικό ποσό του σταθμού

Γ) εμφανίζει πόσοι διαγωνίστηκαν, πόσοι κέρδισαν (μαζί με τα ονόματά τους) και πόσοι κέρδισαν δώρα αξίας 120 €

```
Αλγόριθμος Ασκ_1
poso <-- 2000
diag <-- 0
win <-- 0
win120 <-- 0
Όσο (poso > 0) επανάλαβε
  Διάβασε name
  Σ <-- 0
  Για i από 1 μέχρι 4
    Διάβασε answer
    Σ <-- Σ + answer
  Τέλος_επανάληψης
  Αν (Σ ≥ 40) τότε
    doro <-- 2 * Σ
    Εμφάνισε "Ο παίκτης", name, "κέρδισε", doro
    win <-- win + 1
  Αν doro = 120 τότε
```

```

win120 <-- win120 + 1
Τέλος_Av
Av (poso ≥ doro) τότε
    poso <-- poso – doro
αλλιώς
    poso <-- 0 !αν το δώρο είναι περισσότερο από το ποσό
Τέλος_Av !που απομένει, ο σταθμός δίνει ότι έχει και μηδενίζεται
Τέλος_av
diag <-- diag + 1
Τέλος_Επανάληψης
Εμφάνισε “Διαγωνίστηκαν”, diag, “ακροατές”
Εμφάνισε “Κέρδισαν”, win, “ακροατές”
Εμφάνισε “Δώρα αξίας 120 € κέρδισαν”, win120
Τέλος Ασκ_1

```

Ένα μικρό πλοίο μπορεί να δεχτεί συνολικό φορτίο 9500 kgf. Κατά τη φόρτωσή του πρέπει το συνολικό φορτίο που τοποθετείται στην αριστερή πλευρά να είναι περίπου ίσο με το συνολικό φορτίο που τοποθετείται στη δεξιά πλευρά. Κάθε κιβώτιο ζυγίζεται πριν γίνει η φόρτωσή του. Αν το συνολικό φορτίο της αριστερής πλευράς είναι μικρότερο από αυτό της δεξιάς τότε το κιβώτιο τοποθετείται στην αριστερή πλευρά, διαφορετικά τοποθετείται στην δεξιά πλευρά. Να γραφεί αλγόριθμος που

- A) διαβάσει το βάρος κάθε κιβωτίου
 Β) εμφανίζει το πλήθος των κιβωτίων κάθε πλευράς
 Γ) εμφανίζει το συνολικό βάρος κάθε πλευράς

```

Αλγόριθμος Ασκ_2
left <-- 0
right <-- 0
left_kiv <-- 0
right_kiv <-- 0
sum <-- 0
Διάβασε varos
Όσο (sum + varos < 9500) επανάλαβε
    Av (left < right) τότε
        left <-- left + varos
        left_kiv <-- left_kiv + 1
    αλλιώς
        right <-- right + varos
        right_kiv <-- right_kiv + 1
    Τέλος_av
sum <-- sum + varos
Διάβασε varos
Τέλος_Επανάληψης
Εμφάνισε “Κιβώτια δεξιάς πλευράς”, right_kiv, “με βάρος”, right
Εμφάνισε “Κιβώτια αριστερής πλευράς”, left_kiv, “με βάρος”, left
Τέλος Ασκ_2

```

Σε μια πτήση αεροπορικής εταιρίας χρησιμοποιείται αεροπλάνο με 50 θέσεις ισοκαταμεμημένες δεξιά και αριστερά. Το αεροπλάνο μπορεί να πετάξει με ασφάλεια μόνο όταν η διαφορά βάρους αριστερής και δεξιάς πλευράς είναι λιγότερο από 250 κιλά. Να γραφεί αλγόριθμος που επιτρέπει στον υπάλληλο ταξιδιωτικού πρακτορείου να κάνει για κάθε πελάτη που προσέρχεται στο γκισέ εισητηρίων τα εξής:

- A) Να πληκτρολογεί το βάρος του πελάτη
 Β) Να πληκτρολογεί 'Α' αν ο πελάτης θέλει να κλείσει αριστερή θέση ή 'Δ' αν θέλει να κλείσει δεξιά
 Γ) Να ελέγχει αν είναι εφικτή η επιθυμία του πελάτη (βάσει περιορισμών ασφαλείας και χωρητικότητας αεροπλάνου) και αν είναι να τυπώνει μήνυμα «OK» διαφορετικά να τυπώνει το μήνυμα «Δεν μπορείται να κάνετε αυτή την επιλογή θέσης» και να τυπώνεται το τμήμα του αεροπλάνου που θα κάτσει τελικά ο πελάτης
Σημείωση: Θεωρήστε ότι κανένας επιβάτης δεν είναι 250 κιλά ή περισσότερο. Επίσης η επιλογή της θέσης είναι μόνο 'Α' ή 'Δ'.

Τέλος η διαδικασία κρατάει μέχρι να συμπληρωθούν οι θέσεις της πτήσης

```

Αλγόριθμος Ασκ_3
right <-- 0
sits_R <-- 0
left <-- 0
sits_L <-- 0
πελάτες <-- 0
Όσο (πελάτες < 50) επανάλαβε
    Εμφάνισε “δώσε το βάρος του πελάτη”
    Διάβασε B
    Εμφάνισε “δώσε επιλογή θέσης πελάτη”
    Διάβασε θέση

```

```

Αν (θέση = "Α") και (left+B-right ≤ 250) και (sits_L < 25) τότε
    Εμφάνισε "OK"
    left <- left + B
    sits_L <- sits_L + 1
    πελάτες <- πελάτες + 1
αλλιώς_αν (θέση = "Δ") και (right+B-left ≤ 250) και (sits_R < 25) τότε
    Εμφάνισε "OK"
    right <- right + B
    sits_R <- sits_R + 1
    πελάτες <- πελάτες + 1
αλλιώς
    Εμφάνισε "δεν μπορείτε να κάνετε αυτή την επιλογή θέσης"
Αν (θέση = "Α") και (sits_R < 25) τότε
    Εμφάνισε "θα κάτσετε δεξιά"
    right <- right + B
    sits_R <- sits_R + 1
    πελάτες <- πελάτες + 1
αλλιώς_αν (θέση = "Δ") και (sits_L < 25) τότε
    Εμφάνισε "θα κατσετε αριστερά"
    left <- left + B
    sits_L <- sits_L + 1
    πελάτες <- πελάτες + 1
Τέλος_αν
Τέλος_αν
Τέλος_επανάληψης
Τέλος Ασκ_3

```

Να γραφεί αλγόριθμος που θα μπορεί να χρησιμοποιηθεί στα ΑΤΜ των τραπεζών και θα εκτελεί την παρακάτω λειτουργία :

Α) Ο πελάτης πληκτρολογεί το ποσό των χρημάτων που θέλει να κάνει ανάληψη
 Β) Το ΑΤΜ επειδή διαθέτει μόνο χαρτονομίσματα των 20 € και 50 € εξετάζει αν το ποσό που ζήτησε ο πελάτης είναι εφικτό να δοθεί.

Γ) Αν είναι εφικτό, τυπώνει στην οθόνη του ΑΤΜ τον αριθμό των χαρτονομισμάτων 20 € και των χαρτονομισμάτων 50 € που θα εξαχθούν, διαφορετικά τυπώνει το μήνυμα «Διορθώστε το ποσό» και ακολουθείται πάλι η παραπάνω διαδικασία μέχρι να δοθεί ποσό που μπορεί να εκταμιευθεί από το ΑΤΜ

Σημείωση: Ο εκάστοτε συνδυασμός χαρτονομισμάτων που δίνει κάθε φορά το ΑΤΜ είναι ο ελάχιστος δυνατός. Για παράδειγμα

Ποσό Ανάληψης 100 € → 2 * 50 € και όχι 5 * 20 €

Ποσό Ανάληψης 160 € → 2 * 50 € και 3 * 20 € και όχι 8 * 20 €

```

Αλγόριθμος Ασκ_4
Εμφάνισε "Δώστε το ποσό ανάληψης"
Διάβασε poso
Όσο (poso mod 10 ≠ 0) ή (poso = 10) ή (poso = 30) επανάλαβε
    Εμφάνισε "ΔΙΟΡΘΩΣΤΕ ΤΟ ΠΟΣΟ"
    Διάβασε poso
Τέλος_επανάληψης
euro_50 <- poso div 50
Αν (poso mod 50 = 10) τότε
    euro_50 <- euro_50 - 1
    euro_20 <- 3
αλλιώς_αν (poso mod 50 = 30) τότε
    euro_50 <- euro_50 - 1
    euro_20 <- 4
αλλιώς
    euro_20 <- poso mod 50 div 20
Τέλος_αν
Εμφάνισε "Σύνολο χαρτονομισμάτων 50€ ",euro_50
Εμφάνισε "Σύνολο χαρτονομισμάτων 20€ ",euro_20
Τέλος Ασκ_4

```

Οι αριθμοί Fibonacci ορίζονται ως εξής : $F_0 = 0$, $F_1 = 1$ και $F_k = F_{k-1} + F_{k-2}$. Κάθε αριθμός είναι ίσος με το άθροισμα των δύο προηγούμενων του. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ που για ένα δεδομένο θετικό αριθμό N που διαβάζεται, υπολογίζει τον N-οστό όρο της ακολουθίας (F_N) . Αν ο αριθμός που διαβαστεί είναι αρνητικός, εμφανίζει μήνυμα λάθους

```

ΠΡΟΓΡΑΜΜΑ Ασκ_5
ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ : N, FN, X1, X2, i

```

```

ΑΡΧΗ
ΓΡΑΨΕ 'Δώστε τον αριθμό'
ΔΙΑΒΑΣΕ N
ΑΝ (N<0) ΤΟΤΕ
    ΓΡΑΨΕ 'Λάθος, δώσατε αρνητικό αριθμό'
ΑΛΛΙΩΣ_ΑΝ (N = 0) ΤΟΤΕ
    FN <-- 0
ΑΛΛΙΩΣ_ΑΝ (N = 1) ΤΟΤΕ
    FN <-- 1
ΑΛΛΙΩΣ
    X1 <-- 0
    X2 <-- 1
    ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ N
        FN <-- X1 + X2
        X1 <-- X2
        X2 <-- FN
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΑΝ
ΓΡΑΨΕ 'Ο N-οστος όρος είναι', FN
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ασκ_5

```

Γράψτε πρόγραμμα σε ΓΛΩΣΣΑ που διαβάσει το πολύ 10 ακέραιους αριθμούς και υπολογίζει το άθροισμά τους, το πλήθος τους και τον μέσο όρο τους. Σε περίπτωση που διαβάσει τον αριθμό 5 σταματάει την εκτέλεσή του και εμφανίζει τα τρέχοντα αποτελέσματα.

```

ΠΡΟΓΡΑΜΜΑ Ασκ_6
ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ : i, sum, X
    ΠΡΑΓΜΑΤΙΚΕΣ : ΜΟ
ΑΡΧΗ
sum <-- 0
i <-- 0
ΔΙΑΒΑΣΕ X
ΟΣΟ (X <> 5) ΚΑΙ (i < 10) ΕΠΑΝΑΛΑΒΕ
    sum <-- sum + X
    i <-- i + 1
    ΑΝ i <> 10 ΤΟΤΕ
        ΔΙΑΒΑΣΕ X
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ i <> 0 ΤΟΤΕ
    ΜΟ <-- sum / i
    ΓΡΑΨΕ sum, i
    ΓΡΑΨΕ 'Μέσος Όρος = ', ΜΟ
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ασκ_6

```

Σε μια τάξη υπάρχουν 20 μαθητές που έχουν βαθμολογηθεί σε 8 μαθήματα. Να γράψετε πρόγραμμα που να διαβάσει τα ονόματα και τους βαθμούς των μαθητών και να υπολογίζει τον μέσο όρο κάθε μαθητή. Στη συνέχεια, να εμφανίζει τον καλύτερο μαθητή βάσει των μέσων όρων.

Παρατήρηση: Οι βαθμοί είναι ακέραιοι στην εικοσαβάθμια κλίμακα

```

ΠΡΟΓΡΑΜΜΑ Ασκ_7
ΜΕΤΑΒΛΗΤΕΣ
    ΧΑΡΑΚΤΗΡΕΣ: ON, best_mathitis
    ΑΚΕΡΑΙΕΣ : ΒΑΘ, i, j, Σ
    ΠΡΑΓΜΑΤΙΚΕΣ : ΜΟ, best
ΑΡΧΗ
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20
    ΔΙΑΒΑΣΕ ON
    Σ <-- 0
    best <-- 0
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 8
        ΓΡΑΨΕ 'Δώσε βαθμό του', i, 'μαθητή στο', j, 'μάθημα'
        ΔΙΑΒΑΣΕ ΒΑΘ
        Σ <-- Σ + ΒΑΘ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΜΟ <-- Σ / 8
ΑΝ ΜΟ > best ΤΟΤΕ
    best <-- ΜΟ

```

```

best_mathitis <-- ON
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΙΧΕΙΡΗΣΗΣ
ΓΡΑΨΕ best_mathitis
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ασκ_7

```

Να γράψετε πρόγραμμα σε ΓΛΩΣΣΑ που να διαβάσει για 25 μαθητές το όνομά τους και τη βαθμολογία τους σε 12 μαθήματα. Υπολογίζει τον μέσο όρο για κάθε μαθητή και εμφανίζει το όνομά του και τον μέσο όρο ακολουθούμενο από ένα χαρακτηρισμό («Άριστος» 17-20, «Καλός» 14-17, «Μέτριος» 10-14, «Απορρίπτεται» 0-10). Στα παραπάνω θα πρέπει να υπολογίσουμε συν μία μονάδα (+1) στον μέσο όρο επειδή οι μαθητές είχαν δημιουργήσει πολύ καλές εργασίες. Επίσης θα πρέπει να εμφανίζει το πλήθος των μαθητών που απορρίφθηκαν.

```

ΠΡΟΓΡΑΜΜΑ Ασκ_8
ΜΕΤΑΒΛΗΤΕΣ
    ΑΚΕΡΑΙΕΣ : i, j, sum, Βαθ, pl
    ΠΡΑΓΜΑΤΙΚΕΣ : ΜΟ
    ΧΑΡΑΚΤΗΡΕΣ : ΟΝΟΜΑ
ΑΡΧΗ
    pl <-- 0
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 25
        ΓΡΑΨΕ 'Δώσε όνομα μαθητή', i
        ΔΙΑΒΑΣΕ ΟΝΟΜΑ
        sum <-- 0
        ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 12
            ΓΡΑΨΕ 'Δώσε βαθμό μαθητή', ΟΝΟΜΑ, 'στο μάθημα', j
            ΔΙΑΒΑΣΕ ΟΝΟΜΑ, Βαθ
            sum <-- sum + Βαθ
        ΤΕΛΟΣ_ΕΠΙΧΕΙΡΗΣΗΣ
        ΜΟ <-- sum / 12 + 1
        ΑΝ (ΜΟ > 20) ΤΟΤΕ
            ΜΟ <-- 20
        ΤΕΛΟΣ_ΑΝ
        ΑΝ (ΜΟ >= 17) ΤΟΤΕ
            ΓΡΑΨΕ ΟΝΟΜΑ, ΜΟ, 'Άριστος'
        ΑΛΛΙΩΣ_ΑΝ (ΜΟ >= 14) ΤΟΤΕ
            ΓΡΑΨΕ ΟΝΟΜΑ, ΜΟ, 'Καλός'
        ΑΛΛΙΩΣ_ΑΝ (ΜΟ >= 10) ΤΟΤΕ
            ΓΡΑΨΕ ΟΝΟΜΑ, ΜΟ, 'Μέτριος'
        ΑΛΛΙΩΣ_ΑΝ (ΜΟ >= 0) ΤΟΤΕ
            ΓΡΑΨΕ ΟΝΟΜΑ, ΜΟ, 'Απορρίπτεται'
        pl <-- pl + 1
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΙΧΕΙΡΗΣΗΣ
ΓΡΑΨΕ 'Πλήθος απορριφθέντων', pl
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ασκ_8

```

Έστω ένα δοχείο αποθήκευσης νερού με ύψος 35 cm. Ο κάτωθι πίνακας δείχνει τον τρόπο με τον οποίο μπορούμε να υπολογίσουμε τον όγκο του νερού μέσα στο δοχείο, συναρτήσει του ύψους του (π.χ. αν $Υ = 15$ cm τότε $ΟΓΚΟΣ = 10 * 100 + 5 * 20π$). Το πρόγραμμα

- A) Διαβάζει το ύψος του νερού μέσα στο δοχείο διασφαλίζοντας και ότι είναι σωστό
 Β) Υπολογίζει τον όγκο του νερού
 Γ) Εκτυπώνει το μήνυμα «ΟΓΚΟΣ ΝΕΡΟΥ = » και τον αντίστοιχο όγκο

Υψος Νερού (cm)	Συντελεστής Υπολογισμού Όγκου
$Υ \leq 10$	100
$10 < Υ \leq 20$	$20π$
$Υ > 20$	50

```

ΠΡΟΓΡΑΜΜΑ Ασκ_9
ΣΤΑΘΕΡΕΣ
    p = 3.14
ΜΕΤΑΒΛΗΤΕΣ
    ΠΡΑΓΜΑΤΙΚΕΣ : Υ, V
ΑΡΧΗ
    ΑΡΧΗ_ΕΠΙΧΕΙΡΗΣΗΣ
        ΔΙΑΒΑΣΕ Υ
    ΜΕΧΡΙΣ_ΟΤΟΥ Υ >= 0 ΚΑΙ Υ <= 35
        ΑΝ Υ <= 10 ΤΟΤΕ

```

```

V <- 100 * γ
ΑΛΛΙΩΣ_ΑΝ γ <= 20 ΤΟΤΕ
    V <- 10 * 100 + (γ-10) * 20 * p
ΑΛΛΙΩΣ
    V <- 10 * 100 + 10 * 20 * p + (γ-20) * 50
ΤΕΛΟΣ_ΑΝ
ΓΡΑΨΕ 'ΟΓΚΟΣ ΝΕΡΟΥ = ', V
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ασκ_9

```

Σε ένα αγώνα ρίψης ακοντίου, διεξάγεται ο προκριματικός γύρος με τη συμμετοχή 14 αθλητών. Στην τελική φάση προκρίνονται όσοι αθλητές επιτύχουν επίδοση άνω των 80 μέτρων. Να γραφεί αλγόριθμος που να διαβάσει την επίδοση κάθε αθλητή, να υπολογίζει και να εμφανίζει πόσοι αθλητές πέρασαν το όριο.

```

Αλγόριθμος Ακοντισμός
Π ← 0
Για i από 1 μέχρι 14
    Διάβασε B
    Αν B > 80 τότε
        Π ← Π + 1
Τέλος_αν
Τέλος_επανάληψης
Εμφάνισε Π
Τέλος Ακοντισμός

```

Να γραφεί αλγόριθμος που να διαβάσει το πολύ 10 αριθμούς και να υπολογίζει :

A) το άθροισμα τους

B) το πλήθος τους

Γ) το μέσο όρο τους

Στην περίπτωση που διαβάσει την τιμή 5, σταματάει την εκτέλεση και εμφανίζει τα τρέχοντα αποτελέσματα.

```

Αλγόριθμος Αριθμοί
Διάβασε x
i ← 0
Αθροισμα ← 0
Όσο i <= 10 και x <> 5 επανέλαβε
    Αθροισμα ← Αθροισμα + x
    i ← i + 1
    Διάβασε x
Τέλος_επανάληψης
Αν i = 0 τότε
    Εμφάνισε ' Δώσατε ως πρώτο αριθμό το 5 '
Αλλιώς
    ΜΟ ← Αθροισμα / i
    Εμφάνισε Αθροισμα, i, ΜΟ
Τέλος_αν
Τέλος Αριθμοί

```

Δίνεται τμήμα αλγορίθμου:

```

X ← 13
Όσο X <= 20 επανέλαβε
    εμφάνισε X
    X ← X + 2
Τέλος_επανάληψης
εμφάνισε X

```

α) Για ποια τιμή του X τερματίζεται ο αλγόριθμος;

β) Κατά την εκτέλεση του τμήματος αλγορίθμου ποιες είναι οι τιμές του X που θα εμφανιστούν;

γ) Να γραφεί το παραπάνω τμήμα αλγορίθμου με τη χρήση των εντολών Αρχή_επανάληψης μέχρις_ότου και Για από μέχρι , έτσι ώστε να εμφανίζονται οι ίδιες ακριβώς τιμές.

α) Ο αλγόριθμος τερματίζεται για X=21

β) Οι τιμές που θα εμφανιστούν είναι 13, 15, 17, 19, 21

- γ) i) $X \leftarrow 13$
 Αρχή_επανάληψης
 εμφάνισε X
 $X \leftarrow X+2$
 Μέχρις_ότου $X > 20$
 εμφάνισε X
- ii) Για X από 13 μέχρι 21 με_βήμα 2
 εμφάνισε X
 Τέλος_επανάληψης

Μια δημόσια υπηρεσία διενεργεί διαγωνισμό πρόσληψης και θέλει να εξάγει στατιστικά στοιχεία. Να γραφεί αλγόριθμος ή πρόγραμμα σε Γλώσσα το οποίο :

- α) διαβάζει το ονοματεπώνυμο κάθε υποψηφίου και σταματά μόλις δοθεί ως ονοματεπώνυμο το κενό
 β) διαβάζει την οικογενειακή κατάσταση κάθε υπαλλήλου ('Ε' = Έγγαμος, 'Α' = Άγαμος) και ελέγχει την ορθή εισαγωγή
 γ) διαβάζει το επίπεδο εκπαίδευσης κάθε υπαλλήλου (1 = Υποχρεωτική Εκπαίδευση, 2 = Δευτεροβάθμια Εκπαίδευση 3 = Πανεπιστημιακή Εκπαίδευση) και ελέγχει την ορθή εισαγωγή
 δ) υπολογίζει και εμφανίζει το πλήθος των υπαλλήλων που είναι έγγαμοι
 ε) στο πλήθος των εγγάμων υπαλλήλων να υπολογίζει και να εμφανίζει το ποσοστό των υπαλλήλων που έχουν υποχρεωτική και το ποσοστό των υπαλλήλων που έχουν πανεπιστημιακή εκπαίδευση

```

Αλγόριθμος ΑΣΕΠ
Πλήθος_εγγάμων ← 0
Πλήθος_Υπ ← 0
Πλήθος_Παν ← 0
Διάβασε Ονομα
Όσο Ονομα <> " " επανάλαβε
  Αρχή_επανάληψης
  Διάβασε Οικ_κατάσταση
  Μέχρις_ότου Οικ_κατάσταση = "Ε" ή Οικ_κατάσταση = "Α"
  Αρχή_επανάληψης
  Διάβασε Επίπεδο
  Μέχρις_ότου Επίπεδο = 1 ή Επίπεδο = 2 ή Επίπεδο = 3
  Αν Οικ_κατάσταση = "Ε" τότε
    Πλήθος_εγγάμων ← Πλήθος_εγγάμων + 1
  Αν Επίπεδο = "1" τότε
    Πλήθος_Υπ ← Πλήθος_Υπ + 1
  Τέλος_αν
  Αν Επίπεδο = "3" τότε
    Πλήθος_Παν ← Πλήθος_Παν + 1
  Τέλος_αν
Τέλος_επανάληψης
Διάβασε Ονομα
Τέλος_επανάληψης
Ποσοστό_Υπ ← Πλήθος_Υπ * 100 / Πλήθος_εγγάμων
Ποσοστό_Παν ← Πλήθος_Παν * 100 / Πλήθος_εγγάμων
Εμφάνισε "Το ποσοστό των εγγάμων υπαλλήλων υποχρεωτικής εκπ/σης είναι", Ποσοστό_Υπ
Εμφάνισε "Το ποσοστό των εγγάμων υπαλλήλων πανεπιστημιακής εκπ/σης είναι", Ποσοστό_Υπ
Τέλος ΑΣΕΠ

```

Ένα super market κάνει προσφορές στους κατόχους κάρτας μέλους ανάλογα με τους πόντους που συγκεντρώνουν στις αγορές τους. Για αγορά αξίας 3€ αντιστοιχεί ένας πόντος και μόλις συμπληρωθούν 200 πόντοι τότε γίνεται έκπτωση 6€ στο ποσό αγορών και μηδενίζεται ο αριθμός των πόντων στην κάρτα. Να γραφεί αλγόριθμος ο οποίος :

- α) διαβάζει το ποσό αγορών σε ευρώ για έναν πελάτη
 β) διαβάζει την πληροφορία αν είναι κάτοχος κάρτας μέλους ή όχι
 γ) σε περίπτωση που ο πελάτης είναι κάτοχος κάρτας μέλους τότε
- 1) να διαβάζει τους πόντους που είναι καταγεγραμμένοι στην κάρτα πριν την συγκεκριμένη αγορά
 - 2) να υπολογίζει τους πόντους που αναλογούν από τη συγκεκριμένη αγορά
 - 3) να υπολογίζει το σύνολο των πόντων στη συγκεκριμένη αγορά
 - 4) να υπολογίζει και να εμφανίζει το ποσό που θα πληρώσει καθώς και το σύνολο των πόντων που υπάρχουν στην κάρτα μετά την αγορά
- δ) Σε περίπτωση που ο πελάτης δεν έχει κάρτα να εμφανίζει το ποσό των αγορών που θα πληρώσει

Σημείωση : Να θεωρήσετε ότι οι τιμές της μεταβλητής για την κάρτα μέλους είναι αλφαριθμητική και παίρνει αποκλειστικά τις τιμές «Ναι» ή «Όχι».

```

Αλγόριθμος Super_Market
Διάβασε Ποσό
Διάβασε Κάτοχος_κάρτας
Αν Κάτοχος_κάρτας = "ΝΑΙ" τότε
    Διάβασε Υπάρχοντες_Πόντοι
    Νέοι_Πόντοι ← Ποσό div 3
    Υπάρχοντες_Πόντοι ← Υπάρχοντες_Πόντοι + Νέοι_Πόντοι
    Αν Υπάρχοντες_Πόντοι >= 200 τότε
        Υπάρχοντες_Πόντοι ← Υπάρχοντες_Πόντοι -200
        Ποσό ← Ποσό -6
    Τέλος_αν
    Εμφάνισε Υπάρχοντες_Πόντοι, Ποσό
Αλλιώς
    Εμφάνισε Ποσό
Τέλος_αν
Τέλος Super_Market

```

Σε ένα βιντεοκλάμπ, οι βιντεοκασέτες χρεώνονται ως εξής:

- 1,5 € το τριήμερο.
- 0,60 € για κάθε ημέρα καθυστέρησης.

Να κατασκευαστεί αλγόριθμος ο οποίος να διαβάζει τις ημέρες της ενουκίασης (H) και να υπολογίζει και εμφανίζει τη χρέωση (X).

```

Αλγόριθμος Βίντεοκλαμπ
Διάβασε H
Αν (H <= 3) τότε
    X ← 1,5
αλλιώς
    X ← 1,5 + 0,6*(H - 3)
Τέλος_αν
Εμφάνισε "Η χρέωση θα είναι ", X, " ευρώ"
Τέλος Βίντεοκλάμπ

```

Να γραφεί αλγόριθμος ο οποίος διαβάζει έναν αριθμό σε δραχμές, τον μετατρέπει σε ευρώ και κατόπιν τον εμφανίζει. Η διαδικασία αυτή θα πρέπει να επαναλαμβάνεται ξανά και ξανά, μέχρι ο υπολογιστής να διαβάσει την τιμή 0. Στο τέλος, θα πρέπει επίσης να εμφανίζει και το πλήθος των αριθμών που διάβασε και μετέτρεψε. (Δίνεται ότι: 1 ευρώ = 340,75 δραχμές)

```

Αλγόριθμος Μετατροπές
N ← 0
Εμφάνισε "Δώσε τιμή σε δραχμές (0 για τέλος)"
Διάβασε ΔΡΧ
Όσο (ΔΡΧ <> 0) επανάλαβε
    ΕΥΡΩ ← ΔΡΧ / 340.75
    Εμφάνισε ΕΥΡΩ
    N ← N + 1
    Εμφάνισε "Δώσε τιμή σε δραχμές (0 για τέλος)"
    Διάβασε ΔΡΧ
Τέλος_επανάληψης
Εμφάνισε N
Τέλος Μετατροπές

```


Στο δήμο Πεταλούδων σε κάθε δημοτικό διαμέρισμα λειτουργεί ένας αυτόματος καταγραφέας θερμοκρασίας ο οποίος παίρνει τιμές ανά μία ώρα. Οι τιμές αυτές επεξεργάζονται από το γραφείο Περιβάλλοντος του Δήμου. Από τα αναγκαία κατά την επεξεργασία είναι ο υπολογισμός μέσων τιμών για διάφορες χρονικές περιόδους. Να γίνει αλγόριθμος που θα διαβάζει τις θερμοκρασίες και θα δίνει το μέσο όρο τους. (Το τέλος της ακολουθίας των αριθμών θα δίνεται με μία τιμή της επιλογής σας, την οποία και θα αιτιολογήσετε.)

```

Αλγόριθμος Θερμοκρασίες
Πλήθος ← 0
Αθροισμα ← 0
Εμφάνισε "Δώσε θερμοκρασία (999 για τέλος):"
Διάβασε Θερμοκρασία
Όσο Θερμοκρασία <> 999 επανάλαβε
    Πλήθος ← Πλήθος + 1
    Αθροισμα ← Αθροισμα + Θερμοκρασία
    Εμφάνισε "Δώσε θερμοκρασία (999 για τέλος):"
    Διάβασε Θερμοκρασία
Τέλος_επανάληψης
Αν Πλήθος = 0 τότε
    εμφάνισε "Δε δόθηκε καμία θερμοκρασία"
αλλιώς
    ΜΟ ← Αθροισμα / Πλήθος
    εμφάνισε "Ο μέσος όρος τους είναι: ", ΜΟ
Τέλος_αν
Τέλος Θερμοκρασίες

```

Να γραφεί αλγόριθμος που να υπολογίζει όλες τις ακέραιες λύσεις της εξίσωσης ax^3+bx^2+cx+d στο διάστημα $[-100, 100]$. Ο αλγόριθμος θα πρέπει να είναι «έξυπνος» και στην περίπτωση που βρει και τις τρεις λύσεις, θα πρέπει να σταματάει την αναζήτηση.

```

Αλγόριθμος Τριτοβάθμια
Λύσεις ← 0
Χ ← -100
Εμφάνισε "Δώσε μου τους συντελεστές α, β, γ και δ:"
Διάβασε α, β, γ, δ
Όσο (Λύσεις < 3) και (Χ <= 100) επανάλαβε
    Υ ← α*Χ^3 + β*Χ^2 + γ*Χ + δ
    Αν (Υ = 0) τότε
        Εμφάνισε "Λύση της εξίσωσης: ", Χ
        Λύσεις ← Λύσεις + 1
    Τέλος_αν
    Χ ← Χ + 1
Τέλος_επανάληψης
Εμφάνισε "Πλήθος λύσεων που βρέθηκαν: ", Λύσεις
Τέλος Τριτοβάθμια

```

Ο κύκλος σπουδών μιας σχολής πληροφορικής αποτελείται από 50 μαθήματα και την εκπόνηση μιας διπλωματικής εργασίας. Ο βαθμός πτυχίου ενός αποφοίτου προκύπτει από το μέσο όρο των μαθημάτων επί τον συντελεστή 0,8 συν το βαθμό της διπλωματικής εργασίας επί τον συντελεστή 0,2. Με βάση αυτό το βαθμό, αναγράφεται στο πτυχίο ένας απ' τους ακόλουθους χαρακτηρισμούς:

- "ΑΡΙΣΤΑ", αν $9 \leq \text{βαθμός} \leq 10$
- "ΛΙΑΝ ΚΑΛΩΣ", αν $7 \leq \text{βαθμός} < 9$
- "ΚΑΛΩΣ", αν $5 \leq \text{βαθμός} < 7$.

Να γράψετε αλγόριθμο ο οποίος:

- Διαβάζει τους βαθμούς ενός αποφοίτου στα 50 αυτά μαθήματα.
- Διαβάζει το βαθμό που πήρε στη διπλωματική του εργασία.

- Υπολογίζει κι εμφανίζει το μέσο όρο μαθημάτων.
- Υπολογίζει κι εμφανίζει το βαθμό πτυχίου.
- Εμφανίζει μήνυμα με τον κατάλληλο χαρακτηρισμό.

(Σημείωση: Να θεωρήσετε ότι όλοι οι βαθμοί ανήκουν στο διάστημα [5, 10] και να μην ασχοληθείτε με την εγκυρότητά τους.)

```

Αλγόριθμος Βαθμός_πτυχίου
Αθροισμα ← 0
Για i από 1 μέχρι 50
    Εμφάνισε "Δώσε βαθμό μαθήματος:"
    Διάβασε Βαθμός
    Αθροισμα ← Αθροισμα + Βαθμός
Τέλος_επανάληψης
Εμφάνισε "Δώσε βαθμό διπλωματικής:"
Διάβασε Διπλωματική
ΜΟ ← Αθροισμα / 50
ΒΠ ← 0.8*ΜΟ + 0.2*Διπλωματική
Εμφάνισε "Μέσος όρος μαθημάτων:", ΜΟ
Εμφάνισε "Βαθμός πτυχίου:", ΒΠ
Αν ΒΠ >= 9 τότε
    εμφάνισε "ΑΡΙΣΤΑ"
αλλιώς_αν (ΒΠ >= 7) τότε
    εμφάνισε "ΛΙΑΝ ΚΑΛΩΣ"
αλλιώς
    εμφάνισε "ΚΑΛΩΣ"
Τέλος_αν
Τέλος Βαθμός_πτυχίου

```