DIDAKTIKH

03 ΕΠΙΛΟΓΗ

>_ Δρ. Στυλιανός Καραγιάννης, Ιόνιο Πανεπιστήμιο,
skaragiannis@ionio.gr

Παράδειγμα 1, Τετράδιο Εργασιών, Μετατροπή από βαθμούς Φαρενάιτ σε βαθμούς Κελσίου

Κεφάλαιο 2, Παραδείγματα, Τετράδιο Εργασιών (Μαθητή)

Η μετατροπή μίας θερμοκρασιακής τιμής από βαθμούς Φαρενάιτ σε βαθμούς Κελσίου γίνεται με βάση τον τύπο:

$$C = \frac{5(F-32)}{9}$$

όπου οι μεταβλητές C και F συμβολίζουν τις αντίστοιχες τιμές. Η μετατροπή αυτή γίνεται εύκολα με τον επόμενο αλγόριθμο που έχει ακολουθιακή δομή.
Βασικές Έννοιες Αλγορίθμων Κεφάλαιο 2

Αλγόριθμος Θερμοκρασία

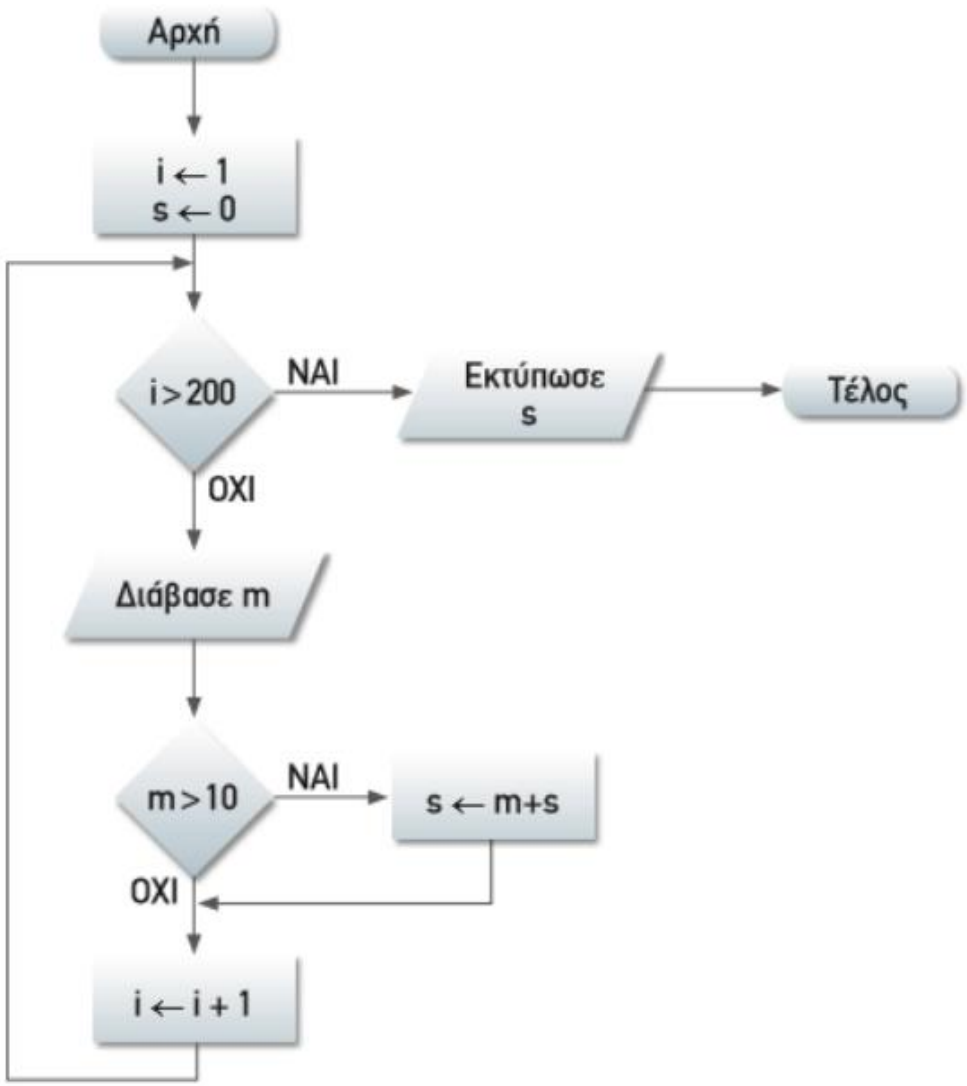
Διάβασε farenheit

celsius ← (farenheit-32) * 5 / 9

Εκτύπωσε celsius

Τέλος Θερμοκρασία

Δίνεται το παρακάτω διάγραμμα ροής:



Να δώσετε την εκφώνηση του προβλήματος που εκφράζεται με το συγκεκριμένο διάγραμμα ροής.

Λύση:

Να γραφεί αλγόριθμος που θα διαβάζει 200 τιμές, θα υπολογίζει και θα εκτυπώνει το άθροισμα των τιμών (από όσες διαβάσθηκαν) που είναι μεγαλύτερες από το 10.

Παράδειγμα 2, Τετράδιο Εργασιών, Υπολογισμός γεωμετρικών μεγεθών

Κεφάλαιο 2, Παραδείγματα, Τετράδιο Εργασιών (Μαθητή)

Έστω ότι δεδομένου του μήκους της ακτίνας θέλουμε να υπολογίσουμε το εμβαδόν του αντίστοιχου κύκλου, το εμβαδόν του τετραγώνου που είναι περιγεγραμμένο στο δεδομένο κύκλο και το μήκος της διαγωνίου του τετραγώνου αυτού. Ο επόμενος αλγόριθμος επιλύει το γεωμετρικό αυτό πρόβλημα, όπου τα ονόματα των μεταβλητών είναι προφανή. Τέλος, διευκρινίζεται ότι ο ακόλουθος αλγόριθμος καλεί έναν αλγόριθμο ονομαζόμενο Ρίζα, που επιστρέφει την τετραγωνική ρίζα ενός θετικού αριθμού.

Αλγόριθμος Γεωμετρικός
Διάβασε aktina
emvadon ← 3.14 * aktina * aktina
plevra ← 2 * aktina
tetragwno ← plevra * plevra
diagwnios ← Ρίζα(2 * tetragwno)
Εκτύπωσε emvadon, tetragwno, diagwnios
Τέλος Γεωμετρικός

ΔΤ1.

```
Αλγόριθμος εκκρεμές
Διάβασε L, g
T ← 2 * 3.14 * Ρίζα(L/g)
Εκτύπωσε T
Τέλος εκκρεμές
```

ΔΤ2 .

```
Αλγόριθμος συνάλλαγμα
Euro ← 330
lira ← 550
dollar ← 280
marko ← 100
synolo ← 1025*lira+2234*dollar+3459*marko
Εκτύπωσε synolo
Τέλος συνάλλαγμα
```

Σχόλιο: η διατήρηση διαφορετικών μεταβλητών για κάθε νόμισμα είναι ιδιαίτερα χρήσιμη γιατί σε περίπτωση ενημέρωσης νέων τιμών συναλλάγματος, αλλάζουν μόνο οι μεταβλητές και όχι ο τύπος για τον υπολογισμό του συνόλου.

ΔΤ3.

1.

```
Αλγόριθμος Μέσος_Ορος
ATHR ← 0
Για i από 1 μέχρι 100
    Διάβασε HLIKIA
    ATHR ← ATHR+ HLIKIA
Τέλος_επανάληψης
MO ← ATHR/100
Αποτελέσματα // MO //
Τέλος Μέσος_Ορος
```

2. Η άσκηση υλοποιείται με δεδομένες τις βαθμολογίες 5 ομάδων.

```
Αλγόριθμος Ομάδες
ATHR ← 0
Για i από 1 μέχρι 5
    Διάβασε VATHMOS
    Αν VATHMOS>100 τότε ATHR ← ATHR+VATHMOS
Τέλος_επανάληψης
Αποτελέσματα // ATHR //
Τέλος Ομάδες
```

Οι υπάλληλοι μίας εταιρείας συμφώνησαν για το μήνα Δεκέμβριο να κρατηθούν από το μισθό τους δύο ποσά, ένα για την ενίσχυση του παιδικού χωριού SOS και ένα για την ενίσχυση των σκοπών της UNICEF. Ο υπολογισμός του ποσού των εισφορών εξαρτάται από τον αρχικό μισθό του κάθε υπαλλήλου και υπολογίζεται με βάση τα παρακάτω όρια μισθών:

Μισθός	Εισφορά 1	Εισφορά 2
Έως 500€	5%	4%
501 – 800	7,5%	6%
801 – 1100	9,5%	8%
μεγαλύτερος από 1100	12%	11%

Να γραφεί αλγόριθμος που να δέχεται ως είσοδο το μισθό του και στη συνέχεια να υπολογίζει το ποσό των δύο εισφορών και το καθαρό ποσό που θα πάρει ο υπάλληλος.

```
Αλγόριθμος εισφορές
Δεδομένα // MISTHOS//
Αν MISTHOS<150000 τότε
    EISF1 ← 0.05*MISTHOS
    EISF2 ← 0.04*MISTHOS
Αλλιώς_αν (MISTHOS>150000 και MISTHOS<250000 τότε
    EISF1 ← 0.075*MISTHOS
    EISF2 ← 0.06*MISTHOS
Αλλιώς_αν (MISTHOS>250000 και MISTHOS<400000 τότε
    EISF1 ← 0.095*MISTHOS
    EISF2 ← 0.08*MISTHOS
Αλλιώς_αν MISTHOS>400000 τότε
    EISF1 ← 0.12*MISTHOS
    EISF2 ← 0.11*MISTHOS
Τέλος_αν
Εκτύπωσε ESIF1, EISF2, MISTHOS-(EISF1+EISHF2)
```

Σε ένα φυτώριο υπάρχουν 3 είδη δένδρων που θα δοθούν για δενδροφύτευση. Το 1ο είδος δένδρου θα δοθεί στην περιοχή της Μακεδονίας, το 2ο στην περιοχή της Θράκης και το 3ο είδος στην περιοχή της Πελοποννήσου. Να σχεδιασθεί το διάγραμμα ροής και να γραφεί ένας αλγόριθμος που θα διαβάζει τον αριθμό του είδους του δένδρου και θα εκτυπώνει την περιοχή στην οποία θα γίνει η δενδροφύτευση.

Λύση:

Αλγόριθμος Φυτώριο

Δεδομένα //Ε //

Αν Ε=1 **τότε**

Εκτύπωσε “ Μακεδονία”

αλλιώς_αν Ε=2 **τότε**

Εκτύπωσε “Θράκη”

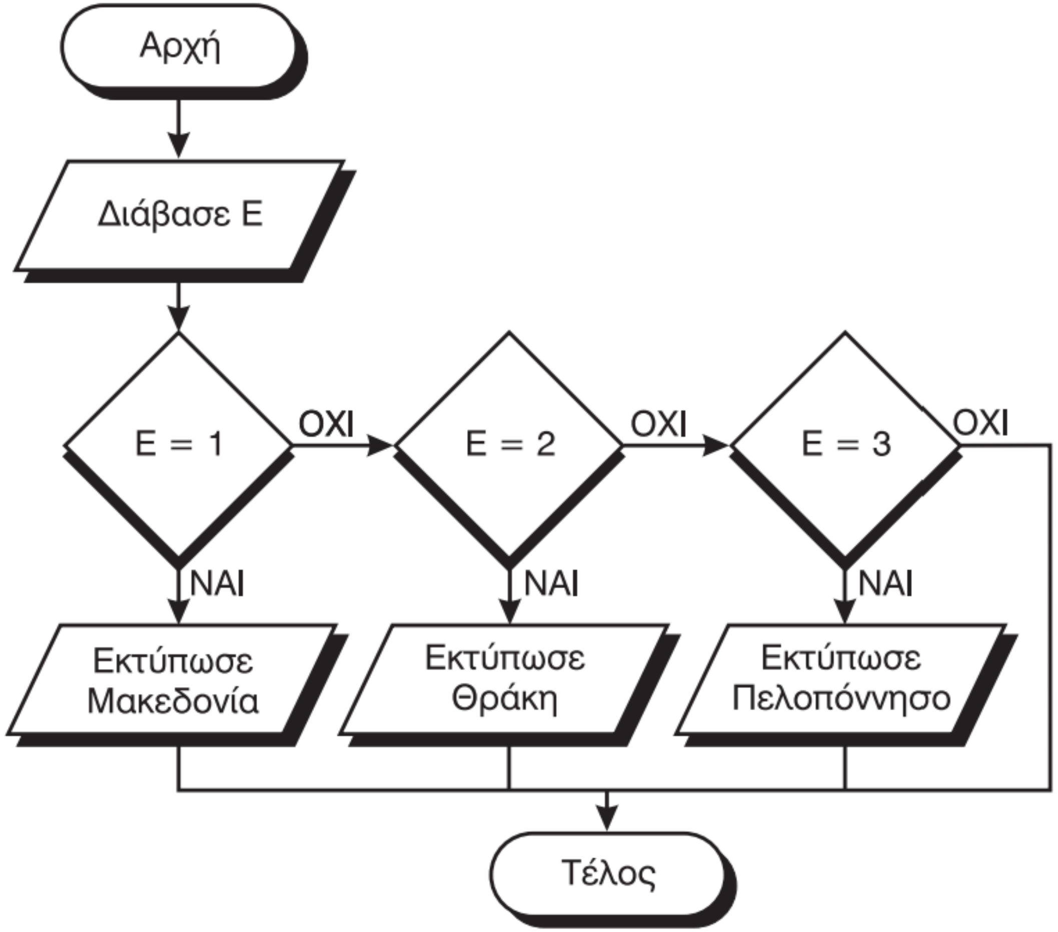
αλλιώς_αν Ε=3 **τότε**

Εκτύπωσε “Πελοπόννησος”

Τέλος_αν

Τέλος Φυτώριο

Διάγραμμα ροής



Πρόβλημα 4. Να γραφεί αλγόριθμος που να εμφανίζει τους αριθμούς από 1 έως 100.

Η αρχική τιμή του μετρητή ορίζεται εύκολα με την εντολή i ← 1. Το πρόβλημα εστιάζεται στο πότε και πώς θα γίνει αντιληπτό ότι φθάσαμε στην τελική τιμή 100. Μια λύση θα ήταν η χρήση της εντολής Αν i=100 τότε ..., αλλά η χρήση της μας οδηγεί αναπόφευκτα στη χρήση και της εντολής "πήγαινε" (goto), πράγμα που πρέπει να αποφευχθεί. Ωστόσο η εντολή **Οσο** *<συνθήκη>* **επανάλαβε** λύνει το πρόβλημα αυτόματα και έτσι καταλήγουμε στον επόμενο αλγόριθμο.

```
Αλγόριθμος Μέτρημα
i ← 1
Οσο i<= 100 επανάλαβε
    Γράψε i
    i ← i+1
Τέλος_επανάληψης
Τέλος Μέτρημα
```

Το τμήμα του αλγόριθμου που επαναλαμβάνεται, δηλ. από την εντολή Οσο μέχρι το Τέλος_επανάληψης αποκαλείται **βρόχος** (Προσοχή, όχι βρόγχος. Βρόχος=θηλιά, αγγλ. loop, γαλ. boucle, ενώ βρόγχος=πνευμόνι). Προφανώς ο βρόχος εκτελείται όσο η συνθήκη είναι αληθής.

Πρόβλημα 4. Να γραφεί αλγόριθμος που να εμφανίζει τους αριθμούς από 1 έως 100.

2η προσέγγιση. Εντολή Για...από...μέχρι

Η προηγούμενη προσέγγιση της επαναληπτικότητας είναι η γενικότερη. Ο βρόχος εκτελείται όσο είναι αληθής η συνθήκη που έχει τεθεί. Ωστόσο αν ο αριθμός των φορών επανάληψης του βρόχου είναι γνωστός εκ των προτέρων, είναι προτιμότερο να χρησιμοποιείται η εντολή Για...από...μέχρι. Ετσι ο προηγούμενος αλγόριθμος γίνεται:

```
Αλγόριθμος Μέτρημα
Για i από 1 μέχρι 100
   Γράψε i
Τέλος_επανάληψης
Τέλος Μέτρημα
```

Οπως είναι φανερό στην εντολή αυτή εμπεριέχονται όλα τα στοιχεία που αφορούν το μετρητή, δηλ. αρχική τιμή, τελική τιμή και βήμα μεταβολής. Ο βρόχος τερματίζει αυτόματα όταν εκτελεστεί για την τελική τιμή.

Πρόβλημα 5. Να υπολογιστεί το άθροισμα 1+2+3+...+100.

Μια λύση που ενδεχόμενα μπορεί να προτείνει κάποιος μαθητής, είναι να χρησιμοποιηθεί ο τύπος του αθροίσματος αριθμητικής προόδου. Ωστόσο μπορεί να μην τον θυμόμαστε ή εν γένει να μην υπάρχει πάντα κάποιος τύπος. Εχοντας λύσει προηγούμενα την επαναληπτικότητα του αλγορίθμου, αρκεί στο βρόχο να προστεθούν κάποια νέα στοιχεία (εντολές) που να επιλύουν την επιπλέον απαίτησης της άθροισης.

Προς τούτο χρησιμοποιούμε μία μεταβλητή, έστω S στην οποία σε κάθε επαναληπτικό βήμα αθροίζουμε την τιμή της μεταβλητής i. Η σχετική εντολή εκχώρησης είναι:

$$S \leftarrow S + i$$

Η εντολή αυτή δρα ως εξής: *"η νέα τιμή του S είναι η παλιά συν την τιμή της μεταβλητής i"*. Συχνά η μεταβλητή S αποκαλείται **αθροιστής** λόγω του ρόλου που παίζει. Ενα σημείο που αξίζει προσοχής είναι ότι ο αθροιστής πρέπει πάντα να εκκινεί με κάποια αρχική τιμή (συνήθως μηδέν, αλλά όχι πάντα).

Ετσι ο αλγόριθμος γίνεται:

```
Αλγόριθμος Αθροισμα
S ← 0
Για i από 1 μέχρι 100
    S ← S + i
Τέλος_επανάληψης
Αποτελέσματα // S //
Τέλος Αθροισμα
```

Σχετικά με τον αλγόριθμο αυτό πρέπει να εξηγηθεί λεπτομερώς ο διαφορετικός ρόλος των μεταβλητών i και S. Προτείνεται ο καθηγητής να δημιουργήσει στον πίνακα δύο παραλληλόγραμμα, ένα για το S και ένα για το i, να τοποθετηθεί σε κάθε ένα η αρχική τιμή και να εκτελεστεί ο αλγόριθμος “με το χέρι” γράφοντας σε κάθε επανάληψη τις διαδοχικές τιμές που λαμβάνουν οι μεταβλητές αυτές, δηλ. 1, 2, 3, 4 ... για το i και 0, 1, 3, 6, 10 ... για το S.

Πρόβλημα 6. Να βρεθεί το Ν παραγοντικό.

Το πρόβλημα αυτό μπορεί να δοθεί ως άσκηση αμέσως μετά από το προηγούμενο. Υπενθυμίζεται ότι Ν!=1.2.3...(Ν-1).Ν και προτείνεται να χρησιμοποιηθεί η μεταβλητή Ρ για την υποδοχή του γινομένου. Οι μαθητές κατά πάσα πιθανότητα θα δημιουργήσουν αλγόριθμο με αρχική τιμή του Ρ=0. Συνιστάται να τους προτρέψουμε να εκτελέσουν τον αλγόριθμο στο χέρι, προκειμένου να εντοπίσουν μόνοι τους το λάθος.

```
Αλγόριθμος Ν_Παραγοντικό
Δεδομένα // Ν //
Ρ ← 1
Για i από 1 μέχρι Ν
   Ρ ← Ρ * i
Τέλος_επανάληψης
Αποτελέσματα // Ρ //
Τέλος Ν_Παραγοντικό
```

Με την ευκαιρία του προβλήματος αυτού καλό είναι να γίνει ένα σχόλιο σχετικά με τη μέγιστη τιμή του Ν. Είναι γνωστό ότι το Ν! αυξάνεται πολύ γρήγορα. Και ενώ στο επίπεδο του αλγορίθμου το θέμα αυτό δεν μας απασχολεί, θα μας απασχολήσει σίγουρα στην υλοποίηση.

ΔΤ2. Τι τύπου μεταβλητές πρέπει να χρησιμοποιήσετε για τα παρακάτω στοιχεία του μαθητολόγιου του σχολείου μας; Γράψετε το αντίστοιχο τμήμα δηλώσεων. 1.Το όνομα ενός μαθητή. 2. Ο αριθμός μαθητολογίου του μαθητή. 3. Τη βαθμολογία του μαθητή. 4. Το τηλέφωνο ενός μαθητή. 5. Τη διεύθυνση ενός μαθητή. 6. Το φύλο ενός μαθητή (πώς μπορεί να οριστεί με χρήση λογικής μεταβλητής;)

1. Χαρακτήρες
2. Ακέραια
3. Πραγματική
4. Χαρακτήρες
5. Χαρακτήρες
6. Χαρακτήρες (αληθής = άνδρας και ψευδής = γυναίκα)

ΔΕ2. Η απόσταση μεταξύ δύο σημείων (x1, y1) και (x2, y2) ενός Καρτεσιανού συστήματος συντεταγμένων υπολογίζεται από τον τύπο: $d=\sqrt{(\chi_1-\chi_2)^2+(y_1-y_2)^2}$

Γράψτε πρόγραμμα το οποίο να υπολογίζει και να εκτυπώνει την απόσταση δύο σημείων των οποίων οι συντεταγμένες δίνονται από το χρήστη

```
ΠΡΟΓΡΑΜΜΑ Καρτεσιανό_Σύστημα
ΜΕΤΑΒΛΗΤΕΣ
  ΠΡΑΓΜΑΤΙΚΕΣ: d, x1, y1, x2, y2
ΑΡΧΗ
  ΔΙΑΒΑΣΕ x1, y1, x2, y2
  d <- ΤΡ ((x1 - x2) ^ 2 + (y1 - y2) ^ 2)
  ΓΡΑΨΕ 'Η απόσταση είναι', d
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```

ΔΣ1. Η μετατροπή της θερμοκρασίας από βαθμούς Celsius σε Fahrenheit δίνεται από τον τύπο: F= 9/5*C+32 Να γραφτεί πρόγραμμα το οποίο να διαβάζει τη θερμοκρασία σε βαθμούς Celsius και να την υπολογίζει και να την τυπώνει σε βαθμούς Fahrenheit $F = \frac{9}{5}C + 32$

```
ΠΡΟΓΡΑΜΜΑ Μετατροπή
ΜΕΤΑΒΛΗΤΕΣ
  ΠΡΑΓΜΑΤΙΚΕΣ: C, F
ΑΡΧΗ
  ΔΙΑΒΑΣΕ C
  F <- (9 / 5) * C + 32
  ΓΡΑΨΕ 'Η θερμοκρασία σε F είναι', F
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```

ΔΤ1. Αν η μεταβλητή Α έχει την τιμή 10, η μεταβλητή Β έχει την τιμή 5 και η μεταβλητή Γ έχει την τιμή 3, ποιες από τις παρακάτω εκφράσεις είναι αληθείς και ποιες ψευδείς.

Α. Όχι (Α > Β)	Α. Ψευδής
Β. Α > Β ΚΑΙ Α<Γ Η Γ=<Β Γ. Α> < Γ Η Γ <=Β	Β. Αληθής
Γ. Α > Β ΚΑΙ (Α < Γ Η Γ = Β)	Γ. Αληθής
Δ. Α = Β Η (Γ - Β) < 0	Δ. Αληθής
Ε. (Α > Β ΚΑΙ Γ< Β) Η (Β <> Γ ΚΑΙ Α < Γ)	Ε. Αληθής

Πόσες φορές θα εκτελεστεί ο βρόχος;

Ποια η λειτουργία των εντολών;

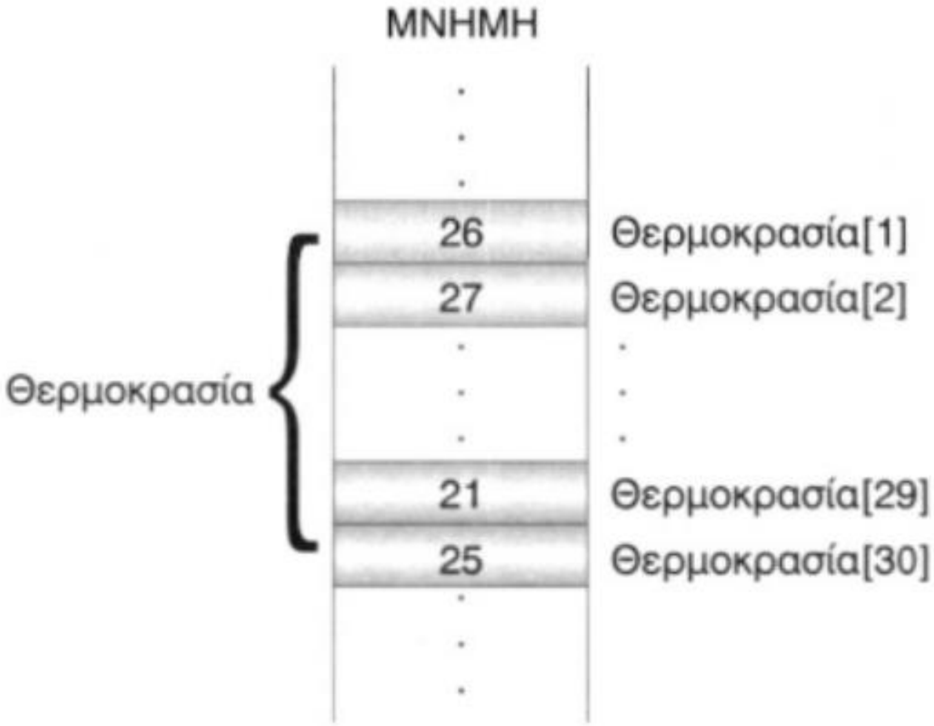
Γράψτε τις παραπάνω εντολές χρησιμοποιώντας την εντολή επανάληψης «ΟΣΟ» και την εντολή επανάληψης «ΜΕΧΡΙΣ_ΟΤΟΥ». Ποιον από τους τρεις τρόπους προτιμάς και γιατί;

Ο αλγόριθμος υπολογίζει το άθροισμα των κύβων των αριθμών που είναι μικρότεροι ή ίσοι του 100 και είναι πολλαπλάσια του 5. Ο βρόχος θα εκτελεστεί λοιπόν 21 φορές. Η δομή επανάληψης ΓΙΑ είναι η πιο κατάλληλη αφού το πλήθος των επαναλήψεων είναι γνωστό

I <- 0	I <- 0
K <- 0	K <- 0
ΟΣΟ I <= 100 **ΕΠΑΝΑΛΑΒΕ**	**ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ**
A <- I ^ 3	A <- I ^ 3
K <- K + A	K <- K + A
ΓΡΑΨΕ I, A	**ΓΡΑΨΕ** I, A
I <- I + 5	I <- I + 5
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ	**ΜΕΧΡΙΣ_ΟΤΟΥ** I > 100
ΓΡΑΨΕ I, A	**ΓΡΑΨΕ** I, A

ΔΣ3. Να γραφτεί πρόγραμμα το οποίο διαβάζει το όνομα ενός μαθητή, τους βαθμούς του σε τρία μαθήματα και υπολογίζει και τυπώνει το μέσο όρο. Το πρόγραμμα να σταματάει, όταν για όνομα δοθεί το κενό

```
ΠΡΟΓΡΑΜΜΑ Υπολογισμός_ΜΟ
ΜΕΤΑΒΛΗΤΕΣ
 ΑΚΕΡΑΙΕΣ: βαθμός, άθροισμα
 ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ
 ΧΑΡΑΚΤΗΡΕΣ : όνομα
ΑΡΧΗ
 ΔΙΑΒΑΣΕ όνομα
 ΟΣΟ (όνομα <> ' ') ΕΠΑΝΑΛΑΒΕ
  άθροισμα <– 0
  ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 3
   ΔΙΑΒΑΣΕ βαθμός
   άθροισμα <– άθροισμα + βαθμός
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  ΜΟ <– άθροισμα / 3
  ΓΡΑΨΕ 'Ο μέσος όρος είναι: ', ΜΟ
  ΔΙΑΒΑΣΕ όνομα
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```



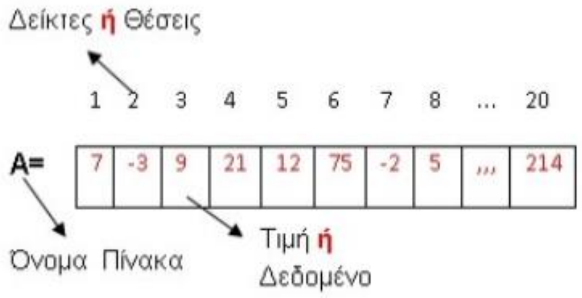
Σχ. 9.1. Ο πίνακας θερμοκρασία

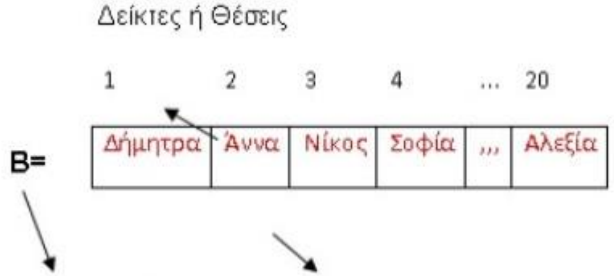
```
ΠΡΟΓΡΑΜΜΑ Θερμοκρασίες
ΜΕΤΑΒΛΗΤΕΣ
   ΠΡΑΓΜΑΤΙΚΕΣ: Θερμοκρασία[30], Μέση, Σύνολο
   ΑΚΕΡΑΙΕΣ: i, Ημέρες
ΑΡΧΗ
   Σύνολο <- 0
   ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 30
      ΓΡΑΨΕ  'Δώσε τη θερμοκρασία'
      ΔΙΑΒΑΣΕ  Θερμοκρασία[i]
      Σύνολο <- Σύνολο+ Θερμοκρασία[i]
   ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
   Μέση <- Σύνολο/30
   Ημέρες <- 0
   ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 30
      ΑΝ Θερμοκρασία[i] < Μέση ΤΟΤΕ
         Ημέρες <- Ημέρες+1
      ΤΕΛΟΣ_ΑΝ
   ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
   ΓΡΑΨΕ 'Μέση Θερμοκρασία:', Μέση
   ΓΡΑΨΕ 'Ημέρες με μικρότερη θερμοκρασία', Ημέρες
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ
```

Μονοδιάστατος

ΜΕΤΑΒΛΗΤΕΣ
ΑΚΕΡΑΙΕΣ: Α[20]
ΧΑΡΑΚΤΗΡΕΣ: Β[20]

Ο πίνακας Α[20] δηλώνει μια μεταβλητή Α με 20 θέσεις. Δηλ. αν τον αναπαριστούσα θα ήταν:





Όταν γράφουμε στο πρόγραμμα Β[2] αναφερόμαστε στο δεδομένο του πίνακα Β στη θέση 2, δηλαδή το δεδομένο "ΑΝΝΑ".

- Η αρίθμηση των θέσεων αρχίζει από το ένα και είναι ακέραιος αριθμός με βήμα 1.

1ος τρόπος
(χωρίς χρήση πίνακα)

ΔΙΑΒΑΣΕ Α1
ΔΙΑΒΑΣΕ Α2
ΔΙΑΒΑΣΕ Α3
ΔΙΑΒΑΣΕ Α4
...
ΔΙΑΒΑΣΕ Α100

Αυτός ο τρόπος δεν χρησιμοποιείται γιατί γράφουμε πολλές φορές την ίδια εντολή.

ή

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20
ΔΙΑΒΑΣΕ Α
ΤΕΛΟΣ_ΕΠΝΑΛΗΨΗΣ

Αυτός ο τρόπος δεν εξυπηρετεί στην περίπτωση που θέλω να έχω όλους τους αριθμούς για να τους επεξεργαστώ, εφόσον και οι 100 αριθμοί που πληκτρολογώ καταχωρούνται στην ίδια μεταβλητή με αποτέλεσμα αυτή να κρατά κάθε φορά τον τελευταίο που πληκτρολογήσαμε.

2ος τρόπος
(με χρήση πίνακα)

ΜΕΤΑΒΛΗΤΕΣ
ΑΚΕΡΑΙΕΣ: Α[100]
ΑΡΧΗ
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
ΔΙΑΒΑΣΕ Α[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
...
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Κατά την εκτέλεση των παραπάνω εντολών πληκτρολογούμε 100 αριθμούς έτσι ώστε ο κάθε ένας να μπαίνει σε διαφορετική θέση στο πίνακα.

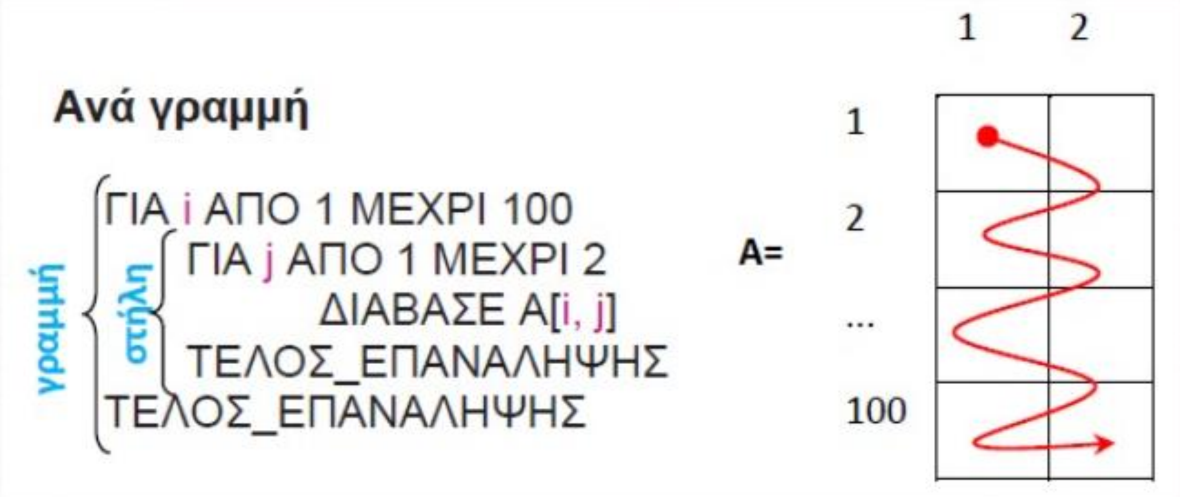
Αυτό έχει σαν αποτέλεσμα να μπορούμε στο πρόγραμμά μας παρακάτω, να τους επεξεργαστούμε όλους ή κάθε έναν απ' αυτούς, όποτε τους χρειαζόμαστε.

Σε δισδιάστατο πίνακα

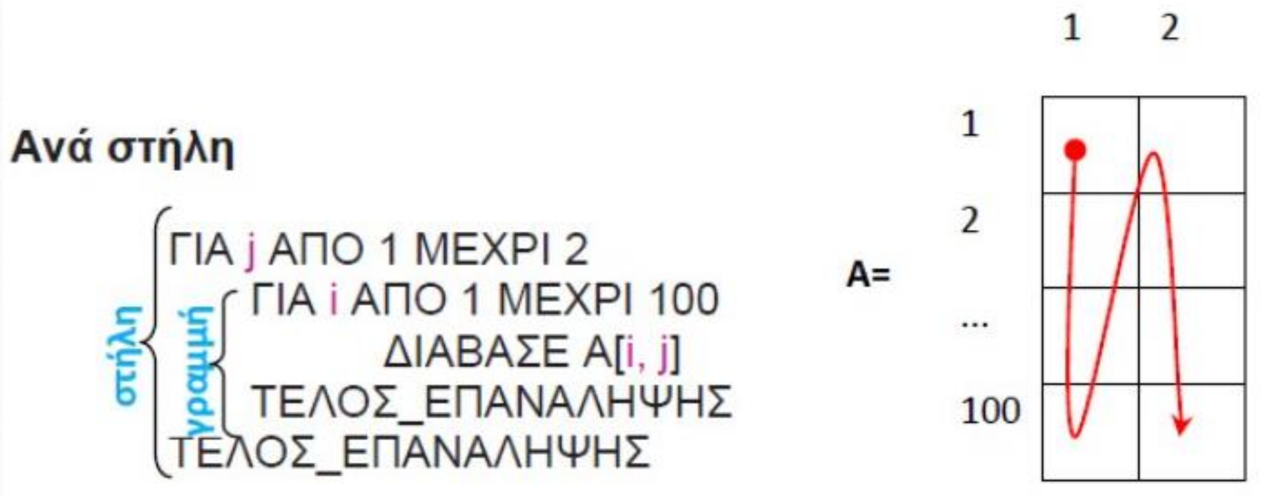
Στη περίπτωση αυτή μας ενδιαφέρει αν τα δεδομένα καταχωρηθούν στο πίνακα ανά γραμμή ή ανά στήλη. Οπότε έχουμε:

π.χ. Διάβασμα 20 ακέραιων και καταχώρηση στον πίνακα Α[10,2]

Με τις εντολές αυτές γεμίζουμε τον πίνακα ανά γραμμή, που σημαίνει ότι όταν γεμίζει μια γραμμή τότε συνεχίζει από την αρχή της επόμενης.



Με τις εντολές αυτές γεμίζουμε τον πίνακα ανά στήλη, που σημαίνει ότι όταν γεμίζει μια στήλη τότε συνεχίζει από την αρχή της επόμενης.



Κεφάλαιο 9 Πίνακες - Τετράδιο Μαθητή

ΔΤ1. Να γράψετε τις δηλώσεις των παρακάτω πινάκων, καθώς και τις εντολές με τις οποίες εκχωρούνται οι τιμές σε αυτά:

Α. Πίνακας 5 στοιχείων που κάθε στοιχείο έχει την τιμή του δείκτη του

Β. Πίνακας που θα περιέχει τα ψηφία

Γ. Πίνακας που περιέχει τα ονόματα των συμμαθητών σου

Δ. Πίνακας με 10 στοιχεία, πρώτο στοιχείο τον αριθμό 500 και κάθε επόμενο στοιχείο να είναι το μισό του προηγούμενου, δηλαδή το δεύτερο 250, το τρίτο 125 κ.ο.κ.

Α.	**ΑΚΕΡΑΙΕΣ** : Α[5] **ΓΙΑ** i **ΑΠΟ** 1 **ΜΕΧΡΙ** 5 Α[i] <- i **ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ**
Β.	**ΑΚΕΡΑΙΕΣ** : Α[10] **ΓΙΑ** i **ΑΠΟ** 1 **ΜΕΧΡΙ** 10 Α[i] <- i - 1 **ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ**
Γ.	**ΧΑΡΑΚΤΗΡΕΣ** : ΟΝΟΜΑΤΑ[30] **ΓΙΑ** i **ΑΠΟ** 1 **ΜΕΧΡΙ** 30 **ΔΙΑΒΑΣΕ** ΟΝΟΜΑΤΑ[i] **ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ**
Δ.	**ΠΡΑΓΜΑΤΙΚΕΣ** : ΠΙΝΑΚΑΣ[10] ΠΙΝΑΚΑΣ[1] <- 500 **ΓΙΑ** i **ΑΠΟ** 2 **ΜΕΧΡΙ** 10 Α[i] <- Α[i - 1] / 2 **ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ**

ΔΤ2. Έχουμε δύο πίνακες, ο ένας με τα μοντέλα των υπολογιστών και ο δεύτερος με τις τιμές τους. Να γράψετε τις εντολές που βρίσκουν και τυπώνουν το φθηνότερο μοντέλο καθώς και το ακριβότερο

```
μέγιστο <- ΤΙΜΗ[1]
θέση_μέγιστο <- 1
ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 30
  ΑΝ (ΤΙΜΗ[i] > μέγιστο) ΤΟΤΕ
    μέγιστο <- ΤΙΜΗ[i]
    θέση_μέγιστο <- i
  ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Το ακριβότερο μοντέλο είναι ', θέση_μέγιστο
ελάχιστο <- ΤΙΜΗ[1]
θέση_ελάχιστο <- 1
ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 30
  ΑΝ (ΤΙΜΗ[i] < ελάχιστο) ΤΟΤΕ
    ελάχιστο <- ΤΙΜΗ[i]
    θέση <- i
  ΤΕΛΟΣ_ΑΝ
```

ΔΤ 1

Θεωρούμε τον πίνακα table με ακεραίους που αντιστοιχούν στις ηλικίες των παιδιών. Οι μεταβλητές min_age και max_age αρχικοποιούνται με την τιμή της πρώτης θέσης του πίνακα, ενώ οι μεταβλητές min και max χρησιμοποιούνται για τη εύρεση της ταυτότητας των δύο παιδιών αντίστοιχα.

```
Αλγόριθμος Κατασκήνωση
Δεδομένα // table//
min_age ← table[1]
max_age ← table[1]
min ← 1
max ← 1
Για i από 2 μέχρι 300
    Αν table[i]<min_age τότε
        min_age ← table[i]
        min ← i
    Τέλος_αν
    Αν table[i]>max_age τότε
        max_age ← table[i]
        max ← i
    Τέλος_αν
Τέλος_επανάληψης
Εκτύπωσε min, min_age, max, max_age
Τέλος Κατασκήνωση
```

ΔΤ 4

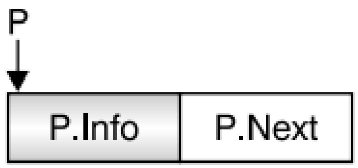
Οι αλγόριθμοι εισαγωγής και εξαγωγής από ουρά δίνονται στη συνέχεια. Χρησιμοποιείται μία λογική μεταβλητή, η σημαία done, που δηλώνει την επιτυχή εκτέλεση της διαδικασίας. Επίσης, οι μεταβλητές rear και front δηλώνουν τους δύο δείκτες που δείχνουν αντίστοιχα στην τελευταία θέση και στην πρώτη θέση της ουράς, που είναι ένας πίνακας queue μεγέθους size. Η μεταβλητή item χρησιμεύει για την αποθήκευση του στοιχείου που εισάγεται ή εξάγεται.

```
Αλγόριθμος Εισαγωγή_σε_Ουρά
Δεδομένα // rear, item //
Αν rear < size τότε
    rear ← rear+1
    queue[rear] ← item
    done ← Αληθής
αλλιώς
    done ← Ψευδής
Τέλος_αν
Αποτελέσματα // rear, done //
Τέλος Εισαγωγή_σε_Ουρά
```

ΔΤ 3

Στη συνέχεια δίνονται οι αλγόριθμοι ώθησης (push) και απώθησης (pop) από στοίβα. Χρησιμοποιείται μία λογική μεταβλητή, η σημαία done, που δηλώνει την επιτυχή εκτέλεση της διαδικασίας. Επίσης, η μεταβλητή top δηλώνει την επάνω θέση της στοίβας που είναι κατειλημμένη από κάποιο στοιχείο. Τα δεδομένα είναι αποθηκευμένα σε ένα μονοδιάστατο πίνακα, που ονομάζεται stack και έχει μέγεθος size. Η μεταβλητή item χρησιμεύει για την αποθήκευση του στοιχείου που εισάγεται ή εξάγεται.

```
Αλγόριθμος Ωθηση
Δεδομένα // top, item //
Αν top<size τότε
   top ← top+1
   stack[top] ← item
   done ← Αληθής
αλλιώς
   done ← Ψευδής
Τέλος_αν
Αποτελέσματα // top, done //
Τέλος Ωθηση
```



Σχήμα 3.6. Δομή κόμβου λίστας

Στη συνέχεια δίνεται ένας απλός αλγόριθμος, που εκτελεί την πράξη της διαγραφής από μία λίστα. Πιο συγκεκριμένα, μας δίνεται ο δείκτης p που δείχνει προς τον κόμβο, του οποίου ο επόμενος θα διαγραφεί. Έτσι αν εφαρμοσθεί ο αλγόριθμος αυτός στη δεύτερη δομή του προηγούμενου σχήματος, θα προκύψει η πρώτη δομή (κατ'αντίστροφη φορά απ'ότι με την εισαγωγή).

```
Αλγόριθμος Διαγραφή_από_λίστα
Δεδομένα // p: δείκτης του κόμβου, του οποίου ο επόμενος
             θα διαγραφεί //
q ← p.next
p.next ← q.next
Τέλος Διαγραφή_από_λίστα
```

Κεφάλαιο 3 Δομές Δεδομένων και Αλγόριθμοι - Τετράδιο Μαθητή

Στην τάξη

ΔΤ1. Σε μία κατασκήνωση υπάρχουν 300 παιδιά και καθένα από αυτά έχει μοναδικό αριθμό από το 1 έως και το 300 που του αντιστοιχεί. Για κάθε παιδί είναι γνωστή η ηλικία του. Να χρησιμοποιηθεί η δομή του πίνακα για να αποθηκεύονται οι ηλικίες των παιδιών και να βρεθεί ο κατάλληλος αλγόριθμος υπολογισμού του μικρότερου και μεγαλύτερου σε ηλικία παιδιού και να εκτυπώνεται τόσο η ηλικία όσο και ο κωδικός του μικρότερου και μεγαλύτερου παιδιού

Κεφάλαιο 3 Δομές Δεδομένων και Αλγόριθμοι - Τετράδιο Μαθητή

Στην τάξη

ΔΤ1. **Σε μία κατασκήνωση υπάρχουν 300 παιδιά και καθένα από αυτά έχει μοναδικό αριθμό από το 1 έως και το 300 που του αντιστοιχεί. Για κάθε παιδί είναι γνωστή η ηλικία του. Να χρησιμοποιηθεί η δομή του πίνακα για να αποθηκεύονται οι ηλικίες των παιδιών και να βρεθεί ο κατάλληλος αλγόριθμος υπολογισμού του μικρότερου και μεγαλύτερου σε ηλικία παιδιού και να εκτυπώνεται τόσο η ηλικία όσο και ο κωδικός του μικρότερου και μεγαλύτερου παιδιού**

```
Αλγόριθμος Κατασκήνωση
 Δεδομένα // ΗΛΙΚΙΑ //
 ελάχιστος ← ΗΛΙΚΙΑ[1]
 θέση_ελάχιστος ← 1
 μέγιστος ← ΗΛΙΚΙΑ[1]
 θέση_μέγιστος ← 1
 Για i από 2 μέχρι 300
  Αν (ΗΛΙΚΙΑ[i] < ελάχιστος) τότε
   ελάχιστος ← ΗΛΙΚΙΑ[i]
   θέση_ελάχιστος ← i
  Τέλος_Αν
  Αν (ΗΛΙΚΙΑ[i] > μέγιστος) τότε
   μέγιστος ← ΗΛΙΚΙΑ[i]
   θέση_μέγιστος ← i
  Τέλος_Αν
 Τέλος_Επανάληψης
 Εκτύπωσε "Η μικρότερη ηλικία είναι ", ελάχιστος, " απο το παιδί με κωδικό", θέση_ελάχιστος
 Εκτύπωσε "Η μεγαλύτερη ηλικία είναι ", μέγιστος, " απο το παιδί με κωδικό", θέση_μέγιστος
Τέλος Κατασκήνωση
```

ΔΣ2. Κατά τη διάρκεια ενός πρωταθλήματος μπάσκετ καταγράφεται ο αριθμός των πόντων που έχουν βάλει 5 παίκτες σε 5 διαφορετικά παιχνίδια. Να γραφτεί αλγόριθμος που θα σε βοηθήσει να κρατήσεις σε ένα δισδιάστατο πίνακα αυτά τα στοιχεία και στη συνέχεια να υπολογίσεις τον παίκτη που έχει πετύχει το μεγαλύτερο αριθμό πόντων από όλα τα παιχνίδια

```
Αλγόριθμος Πρωτάθλημα
 Δεδομένα // ΠΟΝΤΟΙ //
 Για i από 1 μέχρι 5
  άθροισμα ← 0
  Για j από 1 μέχρι 5
   άθροισμα ← άθροισμα + ΠΟΝΤΟΙ[i, j]
  Τέλος_Επανάληψης
  ΣΥΝ_ΠΟΝΤΟΙ[i] ← άθροισμα
 Τέλος_Επανάληψης
 μέγιστος ← ΣΥΝ_ΠΟΝΤΟΙ[1]
 θέση_μέγιστος ← 1
 Για i από 2 μέχρι 5
  Αν (ΣΥΝ_ΠΟΝΤΟΙ[i] > μέγιστος) τότε
   μέγιστος ← ΣΥΝ_ΠΟΝΤΟΙ[i]
   θέση_μέγιστος ← i
  Τέλος_Αν
 Τέλος_Επανάληψης
 Εκτύπωσε "Ο παίκτης με τους περισσότερους πόντους είναι ", θέση_μέγιστος
Τέλος Πρωτάθλημα
```

ΔΣ4. Ένας μαθητής έχει μία συλλογή από δίσκους CD και για κάθε CD έχει καταγράψει στον υπολογιστή τον τίτλο και τη χρονιά έκδοσής του. Να ταξινομηθούν τα CD με βάση τη χρονιά τους και να υπολογιστεί ο αριθμός των CD που έχει ο μαθητής με χρονολογία έκδοσης πριν από το 1995

Θεωρούμε δεδομένο πίνακα CD[N] όπου κάθε γραμμή του πίνακα περιέχει τον τίτλο του συγκεκριμένου CD και πίνακα ΕΤΟΣ[N] όπου περιέχει τη χρονιά έκδοσης, N είναι το πλήθος των διαθέσιμων τίτλων

```
Αλγόριθμος Συλλογή_CD
 Δεδομένα // N, CD, ΕΤΟΣ //
 Για i από 2 μέχρι N
  Για j από N μέχρι i με_βήμα –1
   Αν ΕΤΟΣ[j-1] > ΕΤΟΣ[j] τότε ! αύξουσα ταξινόμηση
    βοηθητική ← ΕΤΟΣ[j-1]
    ΕΤΟΣ[j-1] ← ΕΤΟΣ[j]
    ΕΤΟΣ[j] ← βοηθητική
    βοηθητική1 ← CD[j-1]
    CD[j-1] ← CD[j]
    CD[j] ← βοηθητική1
   Τέλος_αν
  Τέλος_επανάληψης
 Τέλος_επανάληψης
 Για i από 1 μέχρι N
  Αν (ΕΤΟΣ[i] < 1995) τότε
   πλήθος ← πλήθος + 1
  Τέλος_Αν
 Τέλος_Επανάληψης
 Εκτύπωσε "Τα CD πριν το 1995 είναι ", πλήθος
Τέλος Συλλογή_CD
```