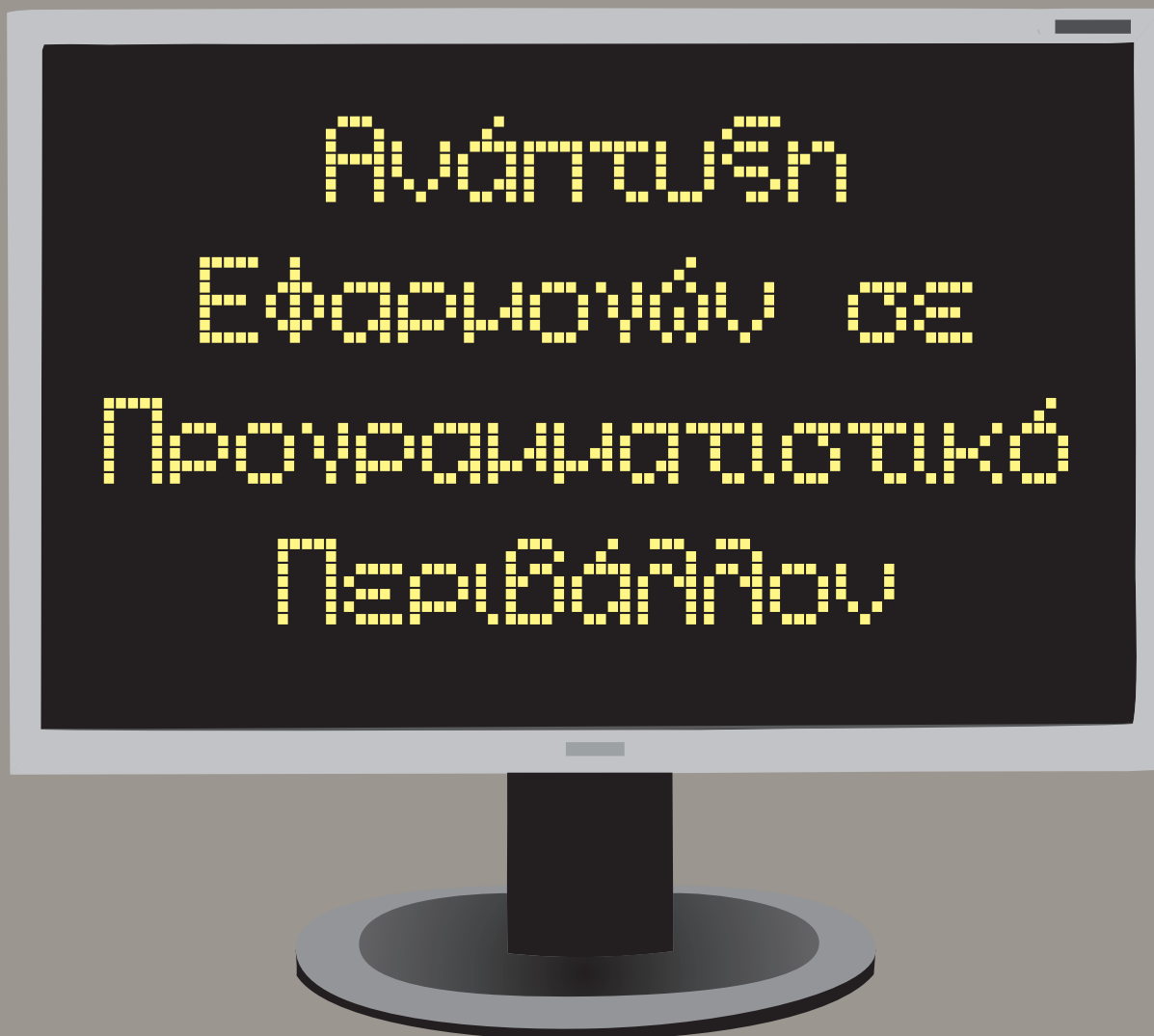


Κωτσάκης Σ., Ταταράκη Α.



ΠΑΡΑΡΤΗΜΑ Α
ΟΔΗΓΙΕΣ ΜΕΛΕΤΗΣ ΜΑΘΗΤΗ

Γ΄ ΓΕΝΙΚΟΥ ΛΥΚΕΙΟΥ

Ομάδας Προσανατολισμού Θετικών Σπουδών και Σπουδών Οικονομίας & Πληροφορικής



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ, ΕΡΕΥΝΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΙΝΣΤΙΤΟΥΤΟ ΕΚΠΑΙΔΕΥΤΙΚΗΣ ΠΟΛΙΤΙΚΗΣ

Κωτσάκης Σ., Ταταράκη Α.

ΑΝΑΠΤΥΞΗ ΕΦΑΡΜΟΓΩΝ
ΣΕ ΠΡΟΓΡΑΜΜΑΤΙΣΤΙΚΟ
ΠΕΡΙΒΑΛΛΟΝ

Γ' Τάξη ΓΕ.Λ.

ΠΑΡΑΡΤΗΜΑ Α
ΟΔΗΓΙΕΣ ΜΕΛΕΤΗΣ ΜΑΘΗΤΗ

ΙΝΣΤΙΤΟΥΤΟ ΕΚΠΑΙΔΕΥΤΙΚΗΣ ΠΟΛΙΤΙΚΗΣ

Πρόεδρος: *Σωτήριος Γκλαβάς*

ΓΡΑΦΕΙΟ ΕΡΕΥΝΑΣ, ΣΧΕΔΙΑΣΜΟΥ ΚΑΙ ΕΦΑΡΜΟΓΩΝ Β΄

Προϊστάμενος: *Παύλος Μάραντος*

ΤΟΜΕΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

Υπεύθυνος: *Θεοδόσιος Τσαπέλας*, Σύμβουλος Β΄ Πληροφορικής ΙΕΠ

ΣΥΓΓΡΑΦΕΙΣ:

Κωτσάκης Σταύρος, Σχολικός σύμβουλος πληροφορικής-ΠΕ19 Πελοποννήσου

Ταταράκη Αλεξάνδρα, Καθηγήτρια ΠΕ19

ΣΥΝΤΟΝΙΣΤΗΣ

Κωτσάκης Σταύρος, Σχολικός σύμβουλος πληροφορικής-ΠΕ19 Πελοποννήσου

ΑΘΗΝΑ ΙΟΥΝΙΟΣ 2015

ΠΡΟΕΚΤΥΠΩΤΙΚΕΣ ΕΡΓΑΣΙΕΣ: ΔΙΕΥΘΥΝΣΗ ΕΚΔΟΣΕΩΝ / ΙΤΥΕ - ΔΙΟΦΑΝΤΟΣ

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΓΡΑΜΜΑ - ΓΛΩΣΣΑ, ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ & ΕΝΤΟΛΕΣ	6
1. ΠΡΟΓΡΑΜΜΑ – ΓΛΩΣΣΑ. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ & ΕΝΤΟΛΕΣ.....	6
1.1. Δομή προγράμματος σε «Γλώσσα».....	6
1.2. Το αλφάβητο της γλώσσας αποτελείται από:.....	6
1.3. Τύποι δεδομένων.....	7
1.4. Σταθερές - Μεταβλητές – Ενσωματωμένες συναρτήσεις.....	7
1.5. Αριθμητικές εκφράσεις.....	10
1.6. Συγκριτικές – Λογικές πράξεις.....	11
1.7. Βασικές εντολές και απλά προγράμματα στη «Γλώσσα».....	13
1.8. Μη λυμένες Ασκήσεις.....	15
ΔΟΜΗ ΕΠΙΛΟΓΗΣ	17
2. ΔΟΜΗ ΕΠΙΛΟΓΗΣ.....	17
2.1. Απλή Επιλογή.....	17
2.2. Πολλαπλή Επιλογή.....	19
2.3. Γενικά Παραδείγματα.....	20
2.4. Μη λυμένες ασκήσεις.....	24
ΔΟΜΗ ΕΠΑΝΑΛΗΨΗΣ	27
3. ΔΟΜΗ ΕΠΑΝΑΛΗΨΗΣ.....	27
3.1. Η εντολή επανάληψης ΟΣΟ...ΕΠΑΝΑΛΑΒΕ.....	27
3.1.1. Χαρακτηριστικά της εντολής:.....	28
3.1.2. Πώς συντάσσεται:.....	28
3.1.3. Πού χρησιμοποιείται:.....	29
3.1.4. Παραδείγματα.....	29
3.2. Η εντολή επανάληψης ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ.....	29
3.2.1. Σύνταξη:.....	30
3.2.2. Χαρακτηριστικά της εντολής:.....	30
3.2.3. Πού χρησιμοποιείται:.....	30
3.2.4. Παραδείγματα.....	30
3.3. Η εντολή επανάληψης ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ.....	34
3.3.1. Τρόπος εκτέλεσης.....	35
3.3.2. Παραδείγματα.....	35
3.3.3. Παρατηρήσεις.....	35
3.3.4. Εμφωλευμένοι Βρόχοι.....	36

3.4.	Αλγόριθμος ελάχιστου ή μεγίστου από μια ομάδα αριθμών.....	36
3.4.1.	Παραδείγματα.....	37
3.5.	Μετατροπές από μια δομή επανάληψης σε άλλη.....	40
3.5.1.	Μετατροπή από ΑΡΧΗ ΕΠΑΝΑΛΗΨΗΣ ... ΣΕ ΟΣΟ...ΚΑΙ ΑΝΤΙΣΤΡΟΦΩΣ.....	40
3.5.1.1.	Παραδείγματα.....	41
3.5.2.	Μετατροπή από ΓΙΑ... σε ΟΣΟ... ή σε ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ και αντιστρόφως.....	42
3.5.2.1.	Παραδείγματα.....	43
3.6.	Μη λυμένες ασκήσεις.....	44
ΠΙΝΑΚΕΣ	47
4.	ΠΙΝΑΚΕΣ.....	47
4.1.	Πλεονεκτήματα χρήσης πινάκων:.....	47
4.2.	Μειονεκτήματα χρήσης πινάκων:.....	47
4.3.	Που χρησιμοποιούνται οι πίνακες;.....	47
4.4.	Δήλωση πίνακα στη ΓΛΩΣΣΑ.....	48
4.5.	Είσοδος δεδομένων σε πίνακα.....	49
4.5.1.	Σε μονοδιάστατο πίνακα.....	49
4.5.2.	Σε δισδιάστατο πίνακα.....	50
4.5.3.	Παραδείγματα.....	50
4.6.	Τυπικές επεξεργασίες πινάκων.....	56
4.6.1.	Ταξινόμηση πίνακα.....	56
4.6.1.1.	Ταξινόμηση ευθείας ανταλλαγής ή φυσαλίδας (Bubblesort).....	56
4.6.2.	Αναζήτηση στοιχείου πίνακα.....	58
4.6.2.1.	Σειριακή αναζήτηση.....	58
4.6.2.2.	Διαδική αναζήτηση.....	59
4.7.	Αλγόριθμος συγχώνευσης δύο πινάκων.....	61
4.8.	Παράλληλοι πίνακες.....	63
4.9.	Λυμένες ασκήσεις.....	66
4.9.1.	Άσκηση 1. Μονοδιάστατος πίνακας.....	66
4.9.2.	Άσκηση 2. Δισδιάστατος πίνακας.....	69
4.9.3.	Άσκηση 3.....	73
4.10.	Μη λυμένες ασκήσεις.....	75
ΤΜΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ - ΥΠΟΠΡΟΓΡΑΜΜΑΤΑ (ΔΙΑΔΙΚΑΣΙΕΣ ΣΥΝΑΡΤΗΣΕΙΣ)	78
5.	ΤΜΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ-ΥΠΟΠΡΟΓΡΑΜΜΑΤΑ (ΔΙΑΔΙΚΑΣΙΕΣ-ΣΥΝΑΡΤΗΣΕΙΣ).....	78
5.1.	Πού χρησιμοποιούνται τα υποπρογράμματα;.....	79
5.2.	Τι πρέπει να προσεχθεί;.....	79
5.3.	Διαδικασίες (Παράδειγμα διαδικασίας).....	80

5.3.1.	Πώς συντάσσεται (ορίζεται) μια διαδικασία;.....	83
5.3.2.	Πώς το κύριο πρόγραμμα καλεί μια διαδικασία;.....	83
5.3.3.	Πώς εκτελείται το πρόγραμμα και πώς λειτουργεί η επικοινωνία του κύριου προγράμματος με τη διαδικασία;.....	83
5.3.3.1.	Σημεία στα οποία πρέπει να δοθεί ιδιαίτερη σημασία.	88
5.3.4.	Κλήση Διαδικασίας από Διαδικασία.....	90
5.3.5.	Διαδικασίες και Πίνακες.....	92
5.4.	Συναρτήσεις (Παράδειγμα συνάρτησης).....	94
5.4.1.	Πώς συντάσσεται και πως καλείται μια συνάρτηση;.....	95
5.4.1.1.	Σύνταξη - ορισμός συνάρτησης.....	95
5.4.1.2.	Κλήση συνάρτησης.....	95
5.4.2.	Σε τι διαφέρουν οι διαδικασίες από τις συναρτήσεις;.....	96
5.4.3.	Διαδικασίες και Συναρτήσεις Μαζί.....	97
5.4.4.	Μη λυμένες ασκήσεις.....	99

ΠΡΟΓΡΑΜΜΑ - ΓΛΩΣΣΑ, ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ & ΕΝΤΟΛΕΣ

Περιεχόμενα

- Πρόγραμμα σε «Γλώσσα»
- Γλώσσα
- Βασικές εντολές και απλά προγράμματα

ΠΡΟΓΡΑΜΜΑ – ΓΛΩΣΣΑ. ΒΑΣΙΚΕΣ ΕΝΝΟΙΕΣ & ΕΝΤΟΛΕΣ

1.1 Δομή προγράμματος σε «Γλώσσα»

Η δομή του προγράμματος στη «Γλώσσα» είναι η παρακάτω:

ΠΡΟΓΡΑΜΜΑ <όνομα προγράμματος>

ΣΤΑΘΕΡΕΣ

<δήλωση σταθερών>

ΜΕΤΑΒΛΗΤΕΣ

<δήλωση μεταβλητών>

ΑΡΧΗ

.....

..... <εντολές προγράμματος>

.....

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ <όνομα προγράμματος>

Παρατηρήσεις

- Το όνομα του προγράμματος πρέπει να υπακούει στους κανόνες δημιουργίας ονομάτων της “ΓΛΩΣΣΑΣ”.
- Οι ΣΤΑΘΕΡΕΣ ή ΜΕΤΑΒΛΗΤΕΣ χρησιμοποιούνται στη περίπτωση που αυτές υπάρχουν στο πρόγραμμα, αλλιώς παραλείπονται.
- Η ΑΡΧΗ και το ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ δηλώνουν αντίστοιχα την αρχή και το τέλος εκτέλεσης του προγράμματος. Μέσα σ’ αυτές είναι το κύριο μέρος του προγράμματος και περιλαμβάνει τις εκτελέσιμες εντολές.
- Κάθε εντολή γράφεται σε ξεχωριστή γραμμή.
- Αν αυτή πρέπει να συνεχιστεί στην επόμενη γραμμή, τότε ο πρώτος χαρακτήρας αυτής της γραμμής πρέπει να είναι ο χαρακτήρας **&**.
- Αν ο πρώτος χαρακτήρας μιας εντολής είναι το θαυμαστικό (!), τότε η γραμμή περιέχει σχόλια και όχι εκτελέσιμες εντολές. Δηλαδή, τη γραμμή αυτή την αγνοεί ο υπολογιστής κατά την εκτέλεση του προγράμματος. Το σχόλιο μάς βοηθά να γράφουμε επεξηγήσεις σε διάφορα σημεία του προγράμματος, ώστε όταν το βλέπουμε μετά από καιρό να καταλαβαίνουμε τι έχει γίνει.

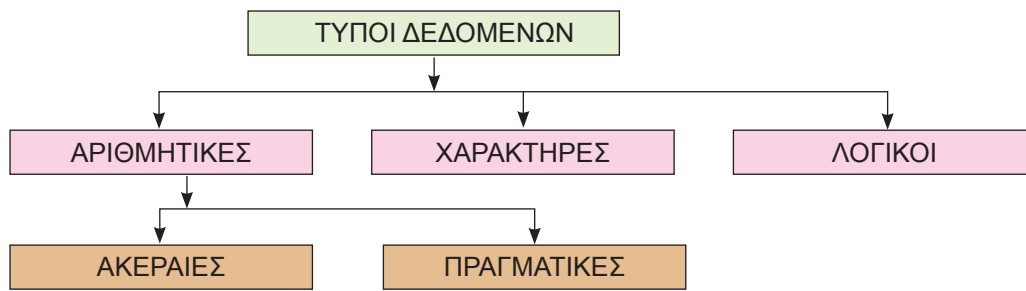
*Από τη δομή του προγράμματος είναι εμφανές ότι για τη δημιουργία οπουδήποτε προγράμματος σε «Γλώσσα», είναι απαραίτητο να γνωρίζουμε τα βασικά στοιχεία της «Γλώσσας» (αλφάβητο, τύπους δεδομένων, σταθερές, μεταβλητές και εντολές).

1.2 Το αλφάβητο της γλώσσας αποτελείται από:

- ΓΡΑΜΜΑΤΑ (Κεφαλαία ελληνικού αλφαβήτου, Πεζά ελληνικού αλφαβήτου, Κεφαλαία λατινικού αλφαβήτου, Πεζά λατινικού αλφαβήτου)
- ΨΗΦΙΑ (0,1,2,3,4,5,6,7,8,9)
- ΕΙΔΙΚΟΥΣ ΧΑΡΑΚΤΗΡΕΣ (+, -, *, /, =, (,), !, &, το κενό

1.3 Τύποι δεδομένων

Η “ΓΛΩΣΣΑ” διαθέτει τους εξής τέσσερις βασικούς τύπους δεδομένων, με τους οποίους καθορίζουμε το τύπο της μεταβλητής ή σταθεράς που θα χρησιμοποιηθεί στο πρόγραμμα.



ΑΚΕΡΑΙΕΣ: περιλαμβάνει όλους τους ακέραιους αριθμούς. Μπορεί να είναι θετικοί, αρνητικοί ή μηδέν.

ΠΡΑΓΜΑΤΙΚΕΣ: περιλαμβάνει όλους τους πραγματικούς αριθμούς. Μπορεί να είναι θετικοί, αρνητικοί ή μηδέν.

ΧΑΡΑΚΤΗΡΕΣ: αναφέρεται τόσο σε ένα χαρακτήρα όσο και σε μια σειρά από χαρακτήρες. Οι χαρακτήρες πρέπει να βρίσκονται υποχρεωτικά μέσα σε εισαγωγικά. Επίσης σαν χαρακτήρες μπορούμε να χρησιμοποιούμε και αριθμούς, γι' αυτό συχνά λέγονται και αλφαριθμητικά. Η διαφορά ενός αριθμού που είναι δηλωμένος σαν Ακέραιος ή Πραγματικός, από τον αριθμό που είναι δηλωμένος σαν Χαρακτήρας, είναι ότι ο πρώτος μπορεί να συμμετάσχει σε αριθμητικές πράξεις, ενώ ο δεύτερος δεν μπορεί.

Παραδείγματα χαρακτήρων: 'Κ', 'Μαρία', 'Αύριο θα πάμε σινεμά', '123', κλπ.

ΛΟΓΙΚΕΣ: Αυτός ο τύπος δέχεται μόνο δύο τιμές: ΑΛΗΘΗΣ και ΨΕΥΔΗΣ.

1.3 Σταθερές - Μεταβλητές – Ενσωματωμένες συναρτήσεις

Σταθερές

Οι σταθερές είναι προκαθορισμένες τιμές που δεν μεταβάλλονται κατά την διάρκεια εκτέλεσης του προγράμματος. Οι σταθερές, ανάλογα με την τιμή που «παίρνουν», διακρίνονται σε:

- ΑΚΕΡΑΙΕΣ π.χ. - 3, 205, +1097
- ΠΡΑΓΜΑΤΙΚΕΣ π.χ. 0.18, - 2.397
- ΧΑΡΑΚΤΗΡΕΣ π.χ. 'Μαρία', 'Πώληση προϊόντων'
- ΛΟΓΙΚΕΣ που είναι μόνο δύο οι τιμές: ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ.

Τρόπος σύνταξης των σταθερών σ' ένα πρόγραμμα:

ΣΤΑΘΕΡΕΣ

Όνομα_1=τιμή1

Όνομα_2=τιμή2

...

Όνομα_v=τιμήν

Παράδειγμα

ΣΤΑΘΕΡΕΣ

ΦΠΑ = 0.19

π = 3.14

Όνομα = 'Μαρία'

Τα ονόματα των σταθερών ΦΠΑ, π ή Όνομα μπορούν να χρησιμοποιηθούν μέσα στο πρόγραμμα και να αντιπροσωπεύουν τις τιμές που έχουμε ορίσει. Οι τιμές αυτές δεν μπορούν να αλλάξουν κατά την διάρκεια εκτέλεσης του προγράμματος. Η χρήση ονομάτων σταθερών κάνει το πρόγραμμα πιο κατανοητό και κατά συνέπεια ευκολότερο να διορθωθεί και να συντηρηθεί.

Μεταβλητές

- Μια μεταβλητή παριστάνει μια ποσότητα που η τιμή της μπορεί να μεταβάλλεται.
- Σ' ένα πρόγραμμα χρησιμοποιούμε διάφορες τιμές, τις οποίες επεξεργαζόμαστε. Αυτές οι τιμές πρέπει να αποθηκεύονται στη μνήμη του υπολογιστή. Άρα κάθε τιμή δεσμεύει μια περιοχή μνήμης. Για να γνωρίζουμε σε ποια περιοχή μνήμης είναι η τιμή που θέλουμε να χρησιμοποιήσουμε, δίνουμε όνομα σε κάθε περιοχή. Το όνομα αυτό είναι το όνομα της μεταβλητής.
- Συνεπώς η μεταβλητή εκφράζει μια περιοχή μνήμης και όταν δώσουμε τιμή στη μεταβλητή, η τιμή αυτή αποθηκεύεται στη περιοχή μνήμης με το αντίστοιχο όνομα.
- Σε μια μεταβλητή, κατά τη διάρκεια εκτέλεσης του προγράμματος, μπορούμε να αλλάξουμε όσες φορές θέλουμε τη τιμή της. Όμως κάθε φορά χάνεται η προηγούμενη τιμή και αντικαθίσταται με τη νέα τιμή της.
- Η τιμή που μπορεί να πάρει μια μεταβλητή είναι ακέραια, πραγματική, χαρακτήρας ή λογική.
- Οι μεταβλητές, ανάλογα με την τιμή που «παίρνουν», διακρίνονται σε:
 - ΑΚΕΡΑΙΕΣ π.χ. - 3, 205, +1097
 - ΠΡΑΓΜΑΤΙΚΕΣ π.χ. 0.18, - 2.397
 - ΧΑΡΑΚΤΗΡΕΣ π.χ. 'Μαρία', 'Πώληση προϊόντων'
 - ΛΟΓΙΚΕΣ που είναι μόνο δύο οι τιμές: ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ

Τρόπος σύνταξης των μεταβλητών σ' ένα πρόγραμμα:

ΜΕΤΑΒΛΗΤΕΣ

τύπος1:Λίστα_μεταβλητών

τύπος2:Λίστα_μεταβλητών

...

τύποςn: Λίστα_μεταβλητών

Παράδειγμα

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ: Χ, Ψ, Μ_Ο

ΑΚΕΡΑΙΕΣ: Ν, Τιμή

ΧΑΡΑΚΤΗΡΕΣ: Όνομα, Επίθετο

ΛΟΓΙΚΕΣ: Έλεγχος

Τα ονόματα Χ, Ψ, Μ_Ο, Ν, Τιμή, Όνομα, Επίθετο και Έλεγχος εκφράζουν τα ονόματα των μεταβλητών, που θα χρησιμοποιήσουμε στο πρόγραμμά μας, όπως επίσης και τον τύπο που εκφράζει κάθε μια απ' αυτές.

Π.χ. η μεταβλητή Χ δέχεται ακέραιες τιμές.

Κανόνες ονοματολογίας Μεταβλητών - Σταθερών

Το όνομα μιας μεταβλητής ή σταθεράς μπορεί να αποτελείται:

- από γράμματα πεζά ή κεφαλαία του ελληνικού ή λατινικού αλφαβήτου
- από τα ψηφία 0, 1, 2, ..., 9
- από σύμβολα μόνο την κάτω παύλα _

Σύμφωνα με τα παραπάνω απαγορεύονται τα εξής:

- Ο πρώτος χαρακτήρας του ονόματος να είναι αριθμός
- Η ύπαρξη σημείων στίξεως ή ειδικών συμβόλων όπως +, -, /, [,], &, # κλπ εκτός από το σύμβολο της κάτω παύλας _
- Η χρήση δεσμευμένων λέξεων της γλώσσας π.χ. οι λέξεις ΑΚΕΡΑΙΕΣ, ΧΑΡΑΚΤΗΡΕΣ κ.λπ.

Παραδείγματα

1. Ποιο από τα παρακάτω ονόματα μεταβλητών ή σταθερών, είναι σωστό ή λάθος και γιατί;
- | | | | | |
|-----------------|---------------|------------|--------|---------|
| 1) α1 | 2) B_1 | 3) B_ | 4) 1_Φ | 5) χ23 |
| 6) 23χ | 7) 256 | 8) χ+ψ | 9) χ/ψ | 10) χ!1 |
| 11) α& | 12) ΠΡΟΓΡΑΜΜΑ | 13) Άννα | | |
| 14) Τιμή_Αγοράς | 15) 'ΜΑΡΙΑ' | 16) ΑΛΗΘΗΣ | | |
| 17) Ψευδής1. | | | | |

Λύση

- 1) α1: Σωστό
 - 2) B_1: Σωστό
 - 3) B_: Σωστό
 - 4) 1_Φ: Λάθος, γιατί ο 1ος χαρακτήρας είναι αριθμός
 - 5) χ23: Σωστό
 - 6) 23χ: Λάθος, γιατί ο 1ος χαρακτήρας είναι αριθμός
 - 7) 256: Λάθος, γιατί ο 1ος χαρακτήρας είναι αριθμός
 - 8) χ+ψ: Λάθος, γιατί υπάρχει το σύμβολο '+'
 - 9) χ/ψ: Λάθος, γιατί υπάρχει το σύμβολο '/'
 - 10) χ!1: Λάθος, γιατί υπάρχει το σύμβολο '!'
 - 11) α&: Λάθος, γιατί υπάρχει το σύμβολο '&'
 - 12) ΠΡΟΓΡΑΜΜΑ: Λάθος, γιατί η λέξη ΠΡΟΓΡΑΜΜΑ είναι δεσμευμένη λέξη
 - 13) Άννα: Σωστό
 - 14) Τιμή_Αγοράς: Σωστό
 - 15) 'ΜΑΡΙΑ': Λάθος, γιατί υπάρχουν τα εισαγωγικά
 - 16) ΑΛΗΘΗΣ: Λάθος, γιατί η λέξη Αληθής είναι δεσμευμένη λέξη
 - 17) Ψευδής1: Σωστό.
2. Στον παρακάτω πίνακα στην 1η στήλη δίνεται ο τύπος της μεταβλητής και στην 2η στήλη η τιμή μιας μεταβλητής. Να αντιστοιχηθούν οι τύποι των μεταβλητών της πρώτης στήλης, στις τιμές των μεταβλητών της δεύτερης στήλης.

Τύπος μεταβλητής	Τιμή μεταβλητής
1. ΑΚΕΡΑΙΕΣ	A. Αληθής
2. ΠΡΑΓΜΑΤΙΚΕΣ	B. 2345
3. ΛΟΓΙΚΕΣ	Γ. 56.9
4. ΠΡΑΓΜΑΤΙΚΕΣ	Δ. 'ΑΛΗΘΗΣ'
	Ε. Ψευδής
	Στ. '90.89'
	Z. -30,89
	H. Άννα'

Λύση

1. → B (γιατί σε κάθε ακέραιο τύπο αντιστοιχούν σαν τιμές οι ακέραιοι αριθμοί)
2. → Γ, Z (γιατί σε κάθε πραγματικό τύπο αντιστοιχούν σαν τιμές οι πραγματικοί αριθμοί, δηλ. αριθμοί με δεκαδικό μέρος)
3. → A, E (γιατί σε κάθε λογικό τύπο αντιστοιχούν οι τιμές Αληθής ή Ψευδής, μόνο).
4. → Δ, ΣΤ, Η (γιατί σε κάθε αλφαριθμητικό τύπο αντιστοιχούν οι τιμές που βρίσκονται μέσα σε εισαγωγικά).

3. Δίνονται οι μεταβλητές με τις αντίστοιχες τιμές τους.

$\alpha \leftarrow 10.89$	$\delta \leftarrow \text{'Αληθής'}$	$\zeta \leftarrow \text{'Νίκος'}$
$\beta \leftarrow \text{'Πρόγραμμα'}$	$x \leftarrow 45$	$\kappa \leftarrow -56$
$\gamma \leftarrow 275$	$y \leftarrow \text{Ψευδής}$	$\lambda \leftarrow -206.9$
		$\epsilon \leftarrow \text{'-4'}$

Λύση

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: γ, κ, κ

ΠΡΑΓΜΑΤΙΚΕΣ: α, λ

ΧΑΡΑΚΤΗΡΕΣ: δ, ζ, ε, β

ΛΟΓΙΚΕΣ: ψ

Να γραφεί το τμήμα του προγράμματος που ορίζουμε τις μεταβλητές, δηλώνοντας τις παραπάνω στους αντίστοιχους τύπους τους.

1.5 Αριθμητικές εκφράσεις

Τελεστές

Οι τελεστές είναι σύμβολα που χρησιμοποιούνται στις διάφορες πράξεις. Οι τελεστές διακρίνονται σε:

- αριθμητικούς
- συγκριτικούς και
- λογικούς

Αριθμητικοί τελεστές

Οι αριθμητικοί τελεστές καλύπτουν τις βασικές πράξεις: πρόσθεση, αφαίρεση, πολλαπλασιασμό και διαίρεση, ενώ υποστηρίζεται η ύψωση σε δύναμη, η ακέραια διαίρεση και το υπόλοιπο της ακέραιας διαίρεσης.

Οι τελεστές με τις αντίστοιχες πράξεις είναι:

Αριθμητικός τελεστής	Πράξη	Παραδείγματα
+	Πρόσθεση	3+5 και επιστρέφει 8
-	Αφαίρεση	5-2 και επιστρέφει 3
*	Πολλαπλασιασμός	2*4 και επιστρέφει 8
/	Διαίρεση	5/2 και επιστρέφει 2.5
^	Ύψωση σε δύναμη	2^3 και επιστρέφει 8
DIV	Ακέραια διαίρεση	5 div 2 και επιστρέφει 2
MOD	Υπόλοιπο ακέραιας διαίρεσης	5 mod 2 και επιστρέφει 1

Συναρτήσεις

Πολλές γνωστές συναρτήσεις από τα μαθηματικά χρησιμοποιούνται συχνά και περιέχονται στη ΓΛΩΣΣΑ και στον αλγόριθμο. Οι συναρτήσεις αυτές είναι:

Συνάρτηση	Περιγραφή	Παράδειγμα
HM(x)	Υπολογισμός ημίτονου	HM(45)
ΣΥΝ(x)	Υπολογισμός συνημίτονου	ΣΥΝ(45)
ΕΦ(x)	Υπολογισμός εφαπτομένης	ΕΦ(45)
T_P(x)	Υπολογισμός τετραγωνικής ρίζας	T_P(9) αντιστοιχεί στο $\sqrt{9}$
ΛΟΓ(x)	Υπολογισμός φυσικού λογαρίθμου	ΛΟΓ(32)
E(x)	Υπολογισμός του e^x	E(a+3) αντιστοιχεί στο e^{a+3}
A_M(x)	Ακέραιο μέρος του x	A_M(3.597) δίνει το 3
A_T(x)	Απόλυτη τιμή του x	A_T(-3+1) δίνει το 2

- Για τη σύνταξη μιας αριθμητικής έκφρασης χρησιμοποιούνται αριθμητικοί τελεστές, μεταβλητές, σταθερές, συναρτήσεις και παρενθέσεις.
- Οι αριθμητικές εκφράσεις υλοποιούν απλές ή σύνθετες μαθηματικές πράξεις.
- Κάθε αριθμητική έκφραση μάς δίνει σαν αποτέλεσμα μια αριθμητική τιμή, που βρίσκεται μετά την εκτέλεση πράξεων.
- Οι πράξεις σε μια αριθμητική έκφραση εκτελούνται με την παρακάτω ιεραρχία:
 1. Ύψωση σε δύναμη (^)
 2. Πολλαπλασιασμός (*) και διαίρεση (/ , DIV, MOD)
 3. Πρόσθεση (+) και αφαίρεση (-)
- Σε περίπτωση που υπάρχει παρένθεση στην αριθμητική έκφραση, τότε εκτελούνται πρώτα οι πράξεις στη παρένθεση σύμφωνα με την παραπάνω ιεραρχία.

- Όταν έχουμε πράξεις της ίδιας ιεραρχίας π.χ. πολλαπλασιασμό και διαίρεση, τότε οι πράξεις εκτελούνται διαδοχικά από τα αριστερά προς τα δεξιά.

Παραδείγματα

1. Να γραφούν οι παρακάτω αριθμητικές εκφράσεις σε ΓΛΩΣΣΑ, λαμβάνοντας υπόψη τη σειρά προτεραιότητας εκτέλεσης των πράξεων:

- 1) $\alpha+10$ 2) $1/2\alpha^3$ 3) $\frac{3\chi+2\psi}{\alpha-\beta}$ 4) $2\eta\mu\psi$ 5) $2+3\chi+\frac{3}{\psi}$
 6) $2\frac{2\chi+3}{2*\psi}$ 7) $\frac{9}{8+\chi}+\sqrt{\beta+8}$ 8) $\frac{\chi+\psi}{\sqrt{7+\chi}}$ 9) $|\alpha|+\frac{\chi+9}{10}\alpha$ 10) $\sqrt{a^2+b^2}$

Λύση

- 1) $\alpha+10$
 2) $1/2*\alpha^3$
 3) $(3*\chi+2*\psi)/(α-β)$
 4) $2*HM(\psi)$
 5) $2+3*\psi+3/\psi$
 6) $2*(2*\chi+3)/(2*\psi)$
 7) $9/(9+\chi)+T_P(\beta+8)$
 8) $(\chi+\psi)/T_P(7+\chi)$
 9) $A_T(\alpha)+(\chi+9)/10*\alpha$
 10) $T_P(\alpha^2+\beta^2)$

1.6 Συγκριτικές – Λογικές πράξεις

Οι συγκριτικοί τελεστές είναι:

Τελεστές	Πράξεις	Παραδείγματα
=	Ισότητα	$x = 0$
< >	Διάφορο	ΟΝ < > 'Κώστας'
>	Μεγαλύτερο	Τιμή > 10000
> =	Μεγαλύτερο ή ίσο	$A + B > 5 / (\chi+3)$
<	Μικρότερο	$B^2-4*A*\Gamma < 0$
< =	Μικρότερο ή ίσο	Βάρος < = 1000

Οι συγκρίσεις σε μια μαθηματική έκφραση μας βοηθούν να λαμβάνονται αποφάσεις. Π.Χ. Αν η ηλικία σου είναι μεγαλύτερη των 18 ετών τότε είσαι ενήλικας, διαφορετικά δεν είσαι ενήλικας.

Λογικοί τελεστές

Σε πολλά προβλήματα οι επιλογές δεν αρκεί να γίνονται με απλές λογικές παραστάσεις, δηλ με μια απλή σύγκριση, αλλά χρειάζεται να συνδυαστούν μια ή περισσότερες λογικές εκφράσεις. Οι λογικές εκφράσεις αποτελούνται από λογικές μεταβλητές. Αυτό επιτυγχάνεται με τη χρήση τριών βασικών λογικών τελεστών: ΟΧΙ, ΚΑΙ και Ή. Οι λογικές παραστάσεις αποτελούνται από λογικές μεταβλητές, οι οποίες παίρνουν μια λογική τιμή, ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ. Το αποτέλεσμα που επιστρέφει μια λογική έκφραση είναι μια λογική τιμή ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ, ανάλογα με τις τιμές των μεταβλητών της.

Στον παρακάτω πίνακα δίνονται τις τιμές των τριών αυτών λογικών πράξεων για όλους τους συνδυασμούς τιμών των μεταβλητών τους A, B:

Μεταβλητή A	Μεταβλητή B	A Η B	A ΚΑΙ B	OXI A
Αληθής	Αληθής	Αληθής	Αληθής	Ψευδής
Αληθής	Ψευδής	Αληθής	Ψευδής	Ψευδής
Ψευδής	Αληθής	Αληθής	Ψευδής	Αληθής
Ψευδής	Ψευδής	Ψευδής	Ψευδής	Αληθής

Οι πράξεις σε μια λογική έκφραση εκτελούνται με την παρακάτω ιεραρχία:

1. OXI
2. ΚΑΙ
3. Η

Σε περίπτωση που υπάρχει παρένθεση στην λογική έκφραση, τότε εκτελούνται πρώτα οι πράξεις στη παρένθεση σύμφωνα με την παραπάνω ιεραρχία.

Στη περίπτωση που σε μια έκφραση συνυπάρχουν περισσότερα από ένα είδη πράξεων, τότε η ιεραρχία των πράξεων είναι η εξής:

- 1η: Αριθμητικοί τελεστές
- 2η: Συγκριτικοί τελεστές και
- 3η: Λογικοί τελεστές.

Παραδείγματα

1. Να βρεθούν οι τιμές που δίνουν οι παρακάτω μαθηματικές εκφράσεις:

<p>α) $(20 \text{ div } 2) * 2^2 / 2 =$ $5 * 2^2 / 2 =$ $5 * 4 / 2 =$ $20 / 2 =$ 10</p>	<p>β) $(T_P(49) + 1) \text{ mod } 3 =$ $(7 + 1) \text{ mod } 3 =$ $8 \text{ mod } 3 =$ 2</p>	<p>γ) $A_M(7/2) + 4^2 =$ $A_M(3,5) + 4^2 =$ $3 + 4^2 =$ $3 + 16 =$ 19</p>
<p>δ) $20 / 2 / 4 =$ $10 / 4 =$ 2,5</p>	<p>ε) $12 \text{ mod } 2 + 10 / 4 =$ $0 + 2,5 =$ 2,5</p>	<p>στ) $12 \text{ div } 2 + 6 \text{ mod } 2^2 =$ $12 \text{ div } 2 + 6 \text{ mod } 4 =$ $6 + 2 =$ 8</p>

2. Να βρεθεί ποια από τις παρακάτω εκφράσεις είναι Αληθής ή Ψευδής, αν η μεταβλητή A έχει την τιμή 12, η μεταβλητή B την τιμή 6 και η μεταβλητή Γ την τιμή 2. (Όπου A= Αληθής και Ψ= Ψευδής)

<p>α) $A > B \text{ ΚΑΙ } (A < \Gamma \text{ Η } \Gamma <= B) = \text{Αληθής}$ $12 > 6 \text{ ΚΑΙ } (12 < 2 \text{ Η } 2 < 6) =$ $A \text{ ΚΑΙ } (\Psi \text{ Η } A) =$ $A \text{ ΚΑΙ } A = A$</p>	<p>γ) $A = B + \Gamma + 4 \text{ ΚΑΙ } (A < B \text{ Η } A + \Gamma > B) = \text{Αληθής}$ $12 = 6 + 2 + 4 \text{ ΚΑΙ } (12 < 6 \text{ Η } 12 + 2 > 6) =$ $12 = 12 \text{ ΚΑΙ } (12 < 6 \text{ Η } 14 > 6) =$ $A \text{ ΚΑΙ } (\Psi \text{ Η } A) =$ $A \text{ ΚΑΙ } A = A$</p>
<p>β) $A > B \text{ ΚΑΙ } A < \Gamma \text{ ΚΑΙ } \Gamma < B = \text{Ψευδής}$ $12 > 6 \text{ ΚΑΙ } 12 < 2 \text{ ΚΑΙ } 2 <= 6 =$ $A \text{ ΚΑΙ } \Psi \text{ ΚΑΙ } A =$ $\Psi \text{ ΚΑΙ } A = \Psi$</p>	<p>δ) $(A > B \text{ ΚΑΙ } \Gamma < B) \text{ Η } (B < \Gamma \text{ ΚΑΙ } A < \Gamma) = \text{Αληθής}$ $(12 > 6 \text{ ΚΑΙ } 2 < 6) \text{ Η } (6 < 2 \text{ ΚΑΙ } 12 < 2) =$ $(A \text{ ΚΑΙ } A) \text{ Η } (A \text{ ΚΑΙ } \Psi) =$ $A \text{ Η } \Psi = A$</p>

1.7 Βασικές εντολές και απλά προγράμματα στη «Γλώσσα»

Βασικές εντολές

1. Εντολή εκχώρησης

Σύνταξη: Όνομα μεταβλητής ← <Έκφραση>

όπου < Έκφραση> = κάποια τιμή (ακέραια ή πραγματική ή χαρακτήρας ή λογική) ή κάποια άλλη μεταβλητή ή κάποια μαθηματική παράσταση.

Τρόπος εκτέλεσης: Με την εντολή αυτή καταχωρείται στη μνήμη, στο όνομα της μεταβλητής που έχουμε ορίσει στο αριστερό μέλος της σχέσης, η τιμή που δίνουμε ή που προκύπτει από πράξεις στο δεξί μέλος της σχέσης.

Παρατηρήσεις

1. Σε μια εντολή εκχώρησης μπορεί η μεταβλητή που βρίσκεται στο αριστερό μέρος αυτής να υπάρχει και στο δεξί μέρος της.

Π.χ. 1) $a \leftarrow a+1$ Αύξηση τας μεταβλητής a κατά $+1$. Δηλαδή η νέα τιμή του a είναι η παλιά συν ένα (μετρητής)

2) $\sigma \leftarrow \sigma+x$ Αύξηση τας μεταβλητής σ κατά x . Δηλαδή η νέα τιμή του σ είναι η παλιά συν τη τιμή της μεταβλητής x (αθροιστής)

2. Δεν επιτρέπεται στο αριστερό μέρος αυτής να υπάρχει πράξη ή μια οποιαδήποτε παράσταση ή τιμή. Το μόνο που δέχεται είναι το όνομα μεταβλητής.

3. Η μεταβλητή και η έκφραση πρέπει να είναι του ίδιου τύπου. Δηλαδή αν στο αριστερό μέρος έχουμε μεταβλητή που ο τύπος της είναι ΑΚΕΡΑΙΕΣ, τότε και η έκφραση που έχουμε στο δεξί μέρος της πρέπει να είναι ΑΚΕΡΑΙΕΣ.

Παραδείγματα

1. Ποιες από τις παρακάτω εντολές είναι σωστές, ποιες είναι λάθος και γιατί;

- | | |
|---------------------------------|---|
| 1. $a\beta \leftarrow x1$ | 11. $a_ \leftarrow a1$ |
| 2. $a+\beta \leftarrow x1$ | 12. $B \leftarrow 'ΓΡΑΨΕ'$ |
| 3. $a1 \leftarrow a+\beta$ | 13. $x \leftarrow \text{Όνομα}$ |
| 4. $a \leftarrow 3+2*6^2$ | 14. $z \leftarrow A>B$ |
| 5. $2\psi \leftarrow \psi$ | 15. $z1 \leftarrow 'A>B'$ |
| 6. $ΑΛΗΘΗΣ \leftarrow \alpha$ | 16. $A \leftarrow 3*(5+6^2)/2$ |
| 7. Επώνυμο \leftarrow ΝΙΚΟΣ | 17. Προϊόν \leftarrow Όνομα_Προϊόντος |
| 8. Επώνυμο \leftarrow 'ΝΙΚΟΣ' | 18. Πρ \leftarrow 'H/Y' |
| 9. $a*3 \leftarrow 2*a*\beta$ | 19. Σημαία $\leftarrow 5>2*5$ |
| 10. done \leftarrow ΨΕΥΔΗΣ | 20. Νίκος \leftarrow 'ΜΑΡΙΑ' |

Λύση

1. Σωστή, στη μεταβλητή $a\beta$ καταχωρείται η τιμή της μεταβλητής $x1$.
2. Λάθος, γιατί το όνομα της μεταβλητής $a+\beta$ δεν είναι αποδεκτό εφόσον υπάρχει το σύμβολο $+$.
3. Σωστή, στη μεταβλητή $a1$ καταχωρείται το αποτέλεσμα της πράξης $a+\beta$.
4. Σωστή, στη μεταβλητή a καταχωρείται το αποτέλεσμα της πράξης $3+2*6^2$
5. Λάθος, γιατί το όνομα της μεταβλητής 2ψ δεν είναι αποδεκτό εφόσον ο πρώτος χαρακτήρας είναι αριθμός.
6. Λάθος, γιατί το όνομα της μεταβλητής $ΑΛΗΘΗΣ$ δεν είναι αποδεκτό εφόσον είναι δεσμευμένη λέξη.
7. Σωστό, στη μεταβλητή Επώνυμο καταχωρείται η τιμή της μεταβλητής ΝΙΚΟΣ
8. Σωστό, στη μεταβλητή Επώνυμο καταχωρείται η τιμή ΝΙΚΟΣ.
9. Λάθος, γιατί το όνομα της μεταβλητής $a*3$ δεν είναι αποδεκτό εφόσον υπάρχει το σύμβολο $*$.
10. Σωστό, στη μεταβλητή done καταχωρείται η τιμή Ψευδής.
11. Σωστό, στη μεταβλητή $a_$ καταχωρείται η τιμή της μεταβλητής $a1$.

12. Σωστό, στη μεταβλητή B καταχωρείται η τιμή ΓΡΑΨΕ.
13. Σωστό, στη μεταβλητή x καταχωρείται η τιμή της μεταβλητής Όνομα.
14. Σωστό, στη μεταβλητή z καταχωρείται η τιμή της λογικής πράξης $A > B$, η τιμή αυτή μπορεί να είναι Αληθής ή Ψευδής.
15. Σωστό, στη μεταβλητή z1 καταχωρείται η τιμή $A > B$.
16. Σωστό, στη μεταβλητή A καταχωρείται το αποτέλεσμα της πράξης.
17. Σωστό, στη μεταβλητή Προϊόν καταχωρείται η τιμή της μεταβλητής Όνομα_Προϊόντος
18. Σωστό, στη μεταβλητή Πρ καταχωρείται η τιμή Η/Υ.
19. Σωστό, στη μεταβλητή Σημαία καταχωρείται το αποτέλεσμα της λογικής πράξης $3 > 2 * 5$, δηλαδή Ψευδής.
20. Σωστό, στη μεταβλητή Νίκος καταχωρείται η τιμή ΜΑΡΙΑ.

<p>2. Ποια είναι η τελική τιμή της μεταβλητής α, όταν εκτελεστούν οι παρακάτω εντολές, αν σαν είσοδο δεδομένων από το πληκτρολόγιο δώσουμε τις τιμές 5, 10 και 25.</p> <p style="text-align: center;">ΔΙΑΒΑΣΕ α $\beta \leftarrow 2^3 + \alpha$ ΔΙΑΒΑΣΕ γ,β $\alpha \leftarrow \beta + \gamma + \alpha$</p>	<p style="text-align: center;">Λύση</p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #e1f5fe;"> <th colspan="4">Πίνακας Τιμών</th> </tr> <tr style="background-color: #e1f5fe;"> <th>Εντολές</th> <th>α</th> <th>β</th> <th>γ</th> </tr> </thead> <tbody> <tr> <td>1η</td> <td>5</td> <td></td> <td></td> </tr> <tr> <td>2η</td> <td></td> <td>13</td> <td></td> </tr> <tr> <td>3η</td> <td></td> <td>25</td> <td>10</td> </tr> <tr> <td>4η</td> <td>30</td> <td></td> <td></td> </tr> </tbody> </table> <p>Η τελική τιμή της μεταβλητής α είναι 30</p>	Πίνακας Τιμών				Εντολές	α	β	γ	1η	5			2η		13		3η		25	10	4η	30		
Πίνακας Τιμών																									
Εντολές	α	β	γ																						
1η	5																								
2η		13																							
3η		25	10																						
4η	30																								

3. Έχουμε το παρακάτω τμήμα προγράμματος

```

α ← 3
χ ← 10 * α
έλεγχος ← Αληθής
ψ ← χ < α
έλεγχος ← έλεγχος ΚΑΙ ψ
α ← α + χ

```

Να γίνει ο πίνακας τιμών.

Λύση

Εντολές	α	χ	έλεγχος	ψ
α ← 3	3			
χ ← 10 * α		30		
έλεγχος ← Αληθής			Αληθής	
ψ ← χ < α				Ψευδής
έλεγχος ← έλεγχος ΚΑΙ ψ			Ψευδής	
α ← α + χ	33			

4. Τι εμφανίζει το παρακάτω τμήμα προγράμματος;

```

α ← 3^2
ΓΡΑΨΕ 'α =', α
β ← 5
ΓΡΑΨΕ 'β =', β
γ ← α * β
ΓΡΑΨΕ 'α * β = ', γ

```

Λύση

Οθόνη

Πίνακας Τιμών			
Εντολές	α	β	γ
1η	9		
2η		5	
5η			45

$$\alpha=9$$

$$\beta=5$$

$$\alpha*\beta=45$$

5. Σε μια εταιρεία οι αποδοχές ενός υπαλλήλου καθορίζονται από το βασικό μισθό, τα επιδόματα και τις κρατήσεις. Ο βασικός μισθός είναι 537€. Το επίδομα για κάθε υπάλληλο είναι 12€ για κάθε χρόνο εργασίας. Οι κρατήσεις είναι 15% για φόρο επί του βασικού μισθού και 2% για χαρτόσημο επί του βασικού μισθού. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, που να διαβάζει όνομα υπαλλήλου και χρόνια υπηρεσίας στην εταιρεία. Στη συνέχεια να υπολογίζει το επίδομα και τις κρατήσεις του και τέλος να εμφανίζει το βασικό μισθό, το επίδομα, τις κρατήσεις κάθε μια χωριστά και το μισθό που του καταβάλλεται.

Λύση

ΠΡΟΓΡΑΜΜΑ Μισθός
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Χρ, Επ

ΠΡΑΓΜΑΤΙΚΕΣ: Φόρος, Χαρτόσημο, Πληρωτέο

ΧΑΡΑΚΤΗΡΕΣ: Ον

ΑΡΧΗ

ΓΡΑΨΕ 'Όνομα υπαλλήλου: '

ΔΙΑΒΑΣΕ Ον

ΓΡΑΨΕ 'Χρόνια Υπηρεσίας: '

ΔΙΑΒΑΣΕ Χρ

Επίδομα \leftarrow Χρ * 12

Φόρος \leftarrow 537 * 15/100

Χαρτόσημο \leftarrow 537 * 2/100

Πληρωτέο \leftarrow 537 + Επίδομα - (Χαρτόσημο + Φόρος)

ΓΡΑΨΕ 'Βασικός Μισθός 650 €'

ΓΡΑΨΕ 'Επίδομα:', Επίδομα

ΓΡΑΨΕ 'Φόρος:', Φόρος

ΓΡΑΨΕ 'Χαρτόσημο:', Χαρτόσημο

ΓΡΑΨΕ 'Μισθός:', Πληρωτέο

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μισθός

1.8. Μη λυμένες Ασκήσεις

1. Ποιες από τις παρακάτω εντολές είναι σωστές, ποιες είναι λάθος και γιατί;

1. $x * x \leftarrow \alpha 1$

2. $x\psi \leftarrow \beta$

3. $\psi \leftarrow 5^2 + 3$

4. $\alpha \leftarrow 'x > 3'$

5. $yx \leftarrow x > 3$

6. $x + \psi \beta \leftarrow 5$

7. $3\kappa \leftarrow \alpha + \beta$

8. $1x \leftarrow x + \psi$

9. $\alpha 1 \leftarrow 5 + x$

10. $x \leftarrow 'Ψευδής'$

11. $y \leftarrow \Psiευδής$

12. Όνομα \leftarrow ΜΑΡΙΑ

13. Όνομα \leftarrow 'ΜΑΡΙΑ'

14. Τιμή \leftarrow '12'

15. Τιμή \leftarrow 12

16. $\alpha \leftarrow 3 < 2 * 5$

17. $\alpha \& \beta \leftarrow \alpha$

18. ΑΛΗΘΗΣ \leftarrow α

19. $x \leftarrow 3 * 2^2 + (10 - 6)^2$

20. ΕΛΕΝΗ \leftarrow X + ψ

2. Να υπολογιστούν οι τιμές των παρακάτω αριθμητικών εκφράσεων:

1. $(3^2 + T_P(9))/2$

2. $A_T(10 - 5^2) + 7$

3. $3 \bmod 2 + 5^2 - 2 \bmod 3$

4. $(3 * 2 \bmod 2) \bmod 5$

5. $T_P(3^2 + 4^2)/(2 + 1)$

6. $7/2 + 6$

10. $(7 + 6)/2$

3. Να γράψετε τις παρακάτω μαθηματικές εκφράσεις σε ΓΛΩΣΣΑ:

1. $\frac{\chi^2 + \psi^2}{2}$

2. $5\beta + \frac{2\alpha + \beta^2}{\alpha + \beta}$

3. $\sqrt{\chi + \psi} - \frac{4}{3 + \chi}$

4. $\sqrt{\frac{\chi}{7}} + 5$

5. $\sqrt{\frac{\chi + 5}{\psi}} + |\alpha + \beta|$

6. $\alpha^2 + \frac{3 + \sqrt{\chi}}{\sqrt{4 + \psi^2}}$

4. Να βρεθούν οι τιμές των παρακάτω λογικών εκφράσεων, αν η μεταβλητή Α έχει τη τιμή 6, η μεταβλητή Β έχει τη τιμή -2 και η μεταβλητή Γ έχει τη τιμή 20.

(Α > 12 ΚΑΙ Β > -6) Ή (ΟΧΙ Α < 6 ΚΑΙ Β > 11)

ΟΧΙ (Α < 3 Ή Β > -1)

(ΟΧΙ Α > -2 ΚΑΙ (ΟΧΙ Β < -13)

(Α < 43 Ή Β > Γ) ΚΑΙ Α > Γ

ΟΧΙ (Α > Β ΚΑΙ Α > Γ)

ΟΧΙ (Α > Β) ΚΑΙ (Α > Γ)

5. Ποια είναι η τιμή της μεταβλητής x, όταν εκτελεστούν οι παρακάτω εντολές;

(Να γίνει πίνακας τιμών.)

α ← 10

β ← α + 3²

γ ← α + β

χ ← (β + γ * 2) * 100

α ← χ

χ ← β + α

6. Τι εμφανίζουν τα παρακάτω προγράμματα, αν σαν είσοδο δοθούν οι αριθμοί 12 και 7;

<p>ΠΡΟΓΡΑΜΜΑ Πρόγραμμα1 ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: α,β,γ,δ ΑΡΧΗ ΓΡΑΨΕ 'Δώσε 2 ακέραιες τιμές τιμές' ΔΙΑΒΑΣΕ χ,ψ γ ← χ div ψ δ ← χ mod ψ ΓΡΑΨΕ χ=',χ, ψ=',ψ ΓΡΑΨΕ 'Υπόλοιπο=',δ ΓΡΑΨΕ 'Πηλίκο=',γ ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Πρόγραμμα1</p>	<p>ΠΡΟΓΡΑΜΜΑ Πρόγραμμα2 ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: x,y ΑΡΧΗ ΓΡΑΨΕ 'Δώσε το χ' ΔΙΑΒΑΣΕ x ΓΡΑΨΕ 'Δώσε το y' ΔΙΑΒΑΣΕ y x ← 3 ^ 2 + 10 div 2 ^ 2 y ← 5 + 2 ^ 2 * 2 + 7 div 2 ΓΡΑΨΕ 'Νέα τιμή x =',x ΓΡΑΨΕ 'Νέα τιμή y =',y ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Πρόγραμμα2</p>
---	---

7. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, που να διαβάζει το πλήθος των άτοκων δόσεων που επιθυμεί να πληρώσει ένας πελάτης για κάποια αγορά του και το ποσό της αγοράς. Να υπολογίζει και να εμφανίζει με κατάλληλο μήνυμα το ποσό της κάθε δόσης.

8. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, που να διαβάζει έναν τριψήφιο αριθμό. Να υπολογίζει και να εμφανίζει το άθροισμα των ψηφίων του.

9. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, που να διαβάζει το κεφάλαιο που καταθέσαμε σε κάποια τράπεζα με επιτόκιο 1,6%. Να υπολογίζει και να εμφανίζει το τόκο που θα εισπράξουμε μετά από 3 χρόνια.

10. Σε μια εταιρεία ο μισθός ενός υπαλλήλου παίρνει επίδομα 20€ για κάθε παιδί και 30€ επίδομα γάμου και έχει κρατήσεις 15% στο βασικό μισθό του. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, που να διαβάζει το βασικό μισθό του υπαλλήλου, και τον αριθμό των παιδιών του (θεωρούμε ότι ο υπάλληλος είναι παντρεμένος). Να υπολογίζει και να εμφανίζει το τελικό μισθό του, τυπώνοντας κατάλληλο μήνυμα.

- > Απλή επιλογή
- > Σύνθετη επιλογή
- > Πολλαπλή επιλογή

2. ΔΟΜΗ ΕΠΙΛΟΓΗΣ

Σε πολλά προβλήματα χρειάζεται να λαμβάνονται αποφάσεις σχετικά με το ποια θα είναι η επόμενη εντολή με βάση κάποια κριτήρια. Η διαδικασία της επιλογής περιλαμβάνει τον έλεγχο κάποιας συνθήκης που μπορεί να έχει δύο τιμές (ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ). Ανάλογα με το αποτέλεσμα της σύγκρισης επιλέγονται οι εντολές που θα εκτελεστούν.

2.1 Απλή Επιλογή

Γενική μορφή σύνταξης:

AN <συνθήκη> **TOTE**

.....

.....

.....

ΤΕΛΟΣ_AN

Τρόπος εκτέλεσης:

Ελέγχεται η λογική συνθήκη και αν αυτή είναι **ΑΛΗΘΗΣ** τότε εκτελούνται οι εντολές μετά το **TOTE** μέχρι το **ΤΕΛΟΣ_AN**. Στην περίπτωση που η λογική συνθήκη πάρει τη τιμή **ΨΕΥΔΗΣ** τότε οι εντολές αυτές αγνοούνται και η εκτέλεση του προγράμματος συνεχίζεται με την εντολή που ακολουθεί μετά το **ΤΕΛΟΣ_AN**.

Παραδείγματα

<p>1. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να διαβάζει την ακτίνα κύκλου και να εμφανίζει το εμβαδόν του κύκλου. Ο αριθμός που θα διαβάζει για ακτίνα πρέπει να είναι θετικός αριθμός.</p> <p style="text-align: center;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Εμβαδόν_κύκλου ΜΕΤΑΒΛΗΤΕΣ ΠΡΑΓΜΑΤΙΚΕΣ: α,Ε ΑΡΧΗ ΔΙΑΒΑΣΕ α AN α > 0 TOTE E←3.14* α^2 ΓΡΑΨΕ 'Εμβαδόν=',E ΤΕΛΟΣ_AN ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Εμβαδόν_κύκλου</p>	<p>2. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να διαβάζει έναν ακέραιο αριθμό και να εμφανίζει την απόλυτη τιμή του.</p> <p style="text-align: center;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Απόλυτη_Τιμή ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: α ΑΡΧΗ ΔΙΑΒΑΣΕ α AN α < 0 TOTE α← (-1)*α ΤΕΛΟΣ_AN ΓΡΑΨΕ ' α = ',α ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Απόλυτη_Τιμή</p>
---	---

1.1 Σύνθετη Επιλογή

Γενική μορφή σύνταξης:

AN <συνθήκη> **TOTE**

.....

..... εντολές 1

.....

ΑΛΛΙΩΣ

.....

..... εντολές 2


.....

ΤΕΛΟΣ_ΑΝ

Τρόπος εκτέλεσης:

Ελέγχεται η λογική συνθήκη και αν αυτή είναι **ΑΛΗΘΗΣ** εκτελούνται οι εντολές 1, ενώ αν αυτή είναι **ΨΕΥΔΗΣ** εκτελούνται οι εντολές 2. Η εκτέλεση του προγράμματος συνεχίζεται με την εντολή που ακολουθεί μετά το **ΤΕΛΟΣ_ΑΝ**.

Παραδείγματα

<p>1. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να διαβάζει την ακτίνα κύκλου και αν αυτή είναι θετικός αριθμός τότε να υπολογίζει και να τυπώνεται το εμβαδόν του κύκλου, αλλιώς να εμφανίζεται κατάλληλο μήνυμα λάθους.</p> <p style="text-align: center;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Εμβαδόν_κύκλου ΜΕΤΑΒΛΗΤΕΣ ΠΡΑΓΜΑΤΙΚΕΣ: α,Ε ΑΡΧΗ ΔΙΑΒΑΣΕ α ΑΝ α > 0 TOTE E ← 3.14 * α² ΓΡΑΨΕ 'Εμβ_κύκλου=',E ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Εμβαδόν_κύκλου</p>	<p>2. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να διαβάζει έναν ακέραιο αριθμό και να εμφανίζει το διπλάσιό του αν το τελευταίο ψηφίο του αριθμού είναι 2 ή 5, ενώ σε διαφορετική περίπτωση, να υπολογίζει και να εμφανίζει το τριπλάσιό του.</p> <p style="text-align: center;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Ψηφία ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: x,y,z ΑΡΧΗ ΔΙΑΒΑΣΕ x y ← x mod 10 ΑΝ y =2 ή y =5 TOTE z ← 2*x ΑΛΛΙΩΣ z ← 3*x ΤΕΛΟΣ_ΑΝ ΓΡΑΨΕ z ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ψηφία</p>
<p>3. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να διαβάζει δύο ακέραιους αριθμούς. Αν είναι και οι δύο άρτιοι ή και οι δύο περιττοί, τότε να εμφανίζει το γινόμενό τους, αλλιώς να εμφανίζει τη διαφορά τους.</p> <p style="text-align: center;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Αριθμοί ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: x,y,α,β,z ΑΡΧΗ ΔΙΑΒΑΣΕ x,y α ← x mod 2 β ← y mod 2 ΑΝ (α=0 ΚΑΙ β=0) Ή (α < >0 ΚΑΙ β < >0) TOTE z ← x*y ΑΛΛΙΩΣ z ← A_T(x - y) ΤΕΛΟΣ_ΑΝ ΓΡΑΨΕ z ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Αριθμοί</p>	<p> Σημείωση: Για να ελέγξουμε αν ένας αριθμός είναι άρτιος ή περιττός, αρκεί να ελέγξουμε το υπόλοιπο της διαιρέσεως με το 2 αν είναι 0 ή όχι (εφόσον όλοι οι άρτιοι είναι διαιρέτες του 2). Δηλαδή: ΑΝ x mod 2=0 TOTE <0 x είναι άρτιος> ΕΝΩ ΑΝ x mod 2< >0 TOTE <0 x είναι περιττός>.</p>

2.2 Πολλαπλή Επιλογή

Γενική μορφή σύνταξης:

1η Μορφή	2η Μορφή
AN <συνθήκη1> TOTE <εντολές1> ΑΛΛΙΩΣ AN <συνθήκη2> TOTE <εντολές2> ΑΛΛΙΩΣ AN <συνθήκη3> TOTE <εντολές3> ΑΛΛΙΩΣ AN <συνθήκη4> TOTE <εντολές4> ΑΛΛΙΩΣ <εντολές5> ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΑΝ	AN <συνθήκη1> TOTE <εντολές1> ΑΛΛΙΩΣ_ΑΝ <συνθήκη2> TOTE <εντολές2> ΑΛΛΙΩΣ_ΑΝ <συνθήκη3> TOTE <εντολές3> ΑΛΛΙΩΣ_ΑΝ <συνθήκη4> TOTE <εντολές4> ΑΛΛΙΩΣ <εντολές5> ΤΕΛΟΣ_ΑΝ

Τρόπος εκτέλεσης:

Αρχικά ελέγχεται η <συνθήκη1> αν είναι αληθής τότε εκτελούνται οι <εντολές1>, αν είναι ψευδής ελέγχεται η <συνθήκη2> και αν αυτή είναι αληθής τότε εκτελούνται οι <εντολές2> αν είναι ψευδής ελέγχεται η <συνθήκη3> και αν αυτή είναι αληθής τότε εκτελούνται οι <εντολές3> κ.ο.κ.

Στη συνέχεια και σε κάθε περίπτωση η εκτέλεση του προγράμματος συνεχίζεται με την εντολή που ακολουθεί μετά το ΤΕΛΟΣ_ΑΝ.



Ένα συχνό λάθος που παρατηρείται στα προγράμματα είναι ο έλεγχος περιττών συνθηκών. Οι παραπάνω έλεγχοι αυξάνουν την πολυπλοκότητα του προγράμματος.

Παραδείγματα

<p>1. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει το πλήθος των Η/Υ που αγοράζονται από κάποιο κατάστημα. Η τιμή αγοράς των Η/Υ εξαρτάται από το πλήθος των Η/Υ που αγοράζονται, σύμφωνα με τον παρακάτω πίνακα:</p> <table border="1" style="width: 100%; text-align: center;"> <thead> <tr> <th>Πλήθος</th> <th>Τιμή ανά τεμάχιο</th> </tr> </thead> <tbody> <tr> <td>1 μέχρι 10</td> <td>500 €</td> </tr> <tr> <td>11 μέχρι 20</td> <td>450 €</td> </tr> <tr> <td>21 μέχρι 50</td> <td>360 €</td> </tr> <tr> <td>51 και πάνω</td> <td>210 €</td> </tr> </tbody> </table> <p>Επίσης να υπολογίζει και να εμφανίζει τη συνολική αξία των υπολογιστών που αγοράστηκαν.</p>	Πλήθος	Τιμή ανά τεμάχιο	1 μέχρι 10	500 €	11 μέχρι 20	450 €	21 μέχρι 50	360 €	51 και πάνω	210 €	<p style="text-align: center;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Υπολογιστές ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: Πλήθος, Αξία ΑΡΧΗ ΓΡΑΨΕ 'Δώσε πλήθος αγοράς ΗΥ' ΔΙΑΒΑΣΕ Πλήθος ΑΝ Πλήθος <=10 TOTE Αξία ← Πλήθος * 500 ΑΛΛΙΩΣ_ΑΝ Πλήθος <=20 TOTE Αξία ← Πλήθος * 450 ΑΛΛΙΩΣ_ΑΝ Πλήθος <=50 TOTE Αξία ← Πλήθος * 360 ΑΛΛΙΩΣ Αξία ← Πλήθος * 210 ΤΕΛΟΣ_ΑΝ ΓΡΑΨΕ 'Συνολική Αξία=', Αξία ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Υπολογιστές</p>
Πλήθος	Τιμή ανά τεμάχιο										
1 μέχρι 10	500 €										
11 μέχρι 20	450 €										
21 μέχρι 50	360 €										
51 και πάνω	210 €										

2. Μια εταιρεία κινητής τηλεφωνίας ακολουθεί ανά μήνα την εξής πολιτική τιμών:

Πάγιο 20€	
Χρόνος ομιλίας σε δευτερ.	Χρονοχρέωση (€ / δευτερ.)
1 - 300	0.50
301 - 1000	0.45
1001 και άνω	0.30

Να γίνει πρόγραμμα σε «ΓΛΩΣΣΑ» το οποίο:

- Να διαβάζει τη χρονική διάρκεια σε δευτερόλεπτα των τηλεφωνημάτων ενός συνδρομητή σε διάστημα ενός μήνα.
- Να υπολογίζει τη μηνιαία χρέωση του συνδρομητή.
- Να εμφανίζει κατάλληλο μήνυμα με τη μηνιαία χρέωση του συνδρομητή.

Λύση

ΠΡΟΓΡΑΜΜΑ Κινητή_Τηλεφωνία
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Χρόνος, Αξία

ΠΡΑΓΜΑΤΙΚΕΣ: Χρέωση

ΑΡΧΗ

ΓΡΑΨΕ 'Χρονική διάρκεια τηλεφώνου'

ΔΙΑΒΑΣΕ Χρόνος

ΑΝ Χρόνος ≥ 1 **ΚΑΙ** Χρόνος ≤ 300 **ΤΟΤΕ**

Χρέωση \leftarrow Χρόνος * 0.50

ΑΛΛΙΩΣ_ΑΝ Χρόνος ≤ 1000 **ΤΟΤΕ**

Χρέωση $\leftarrow (300 * 0.50) + (\text{Χρόνος} - 300) * 0.45$

ΑΛΛΙΩΣ

Χρέωση $\leftarrow (300 * 0.50) + (700 * 0.45) + (\text{Χρόνος} - 1000) * 0.30$

ΤΕΛΟΣ_ΑΝ

Χρέωση \leftarrow Χρέωση + 20

ΓΡΑΨΕ 'Χρέωση=', Χρέωση, '€'

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Κινητή_Τηλεφωνία



Στο παραπάνω πρόβλημα η χρέωση γίνεται **κλιμακωτά**. Δηλαδή, αν π.χ. έχω χρονική διάρκεια τηλεφωνημάτων 650 δευτερόλεπτα, τότε τα πρώτα 300 δευτερόλεπτα χρεώνονται με 0,50€ το δευτερόλεπτο, ενώ τα υπόλοιπα 350 δευτερόλεπτα (650-300) χρεώνονται με 0.45€ το δευτερόλεπτο.

2.3 Γενικά Παραδείγματα

1. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να διαβάζει τρεις πραγματικούς αριθμούς. Αν το άθροισμά τους είναι μεγαλύτερο του 10, να υπολογίζει και να εμφανίζει το μέσο όρο τους. Διαφορετικά να βρίσκει τον μεγαλύτερο απ' αυτούς και να τον εμφανίζει με κατάλληλο μήνυμα.

Λύση

ΠΡΟΓΡΑΜΜΑ Αριθμοί3

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ: α, β, γ, ΜΟ, Max

ΑΡΧΗ

ΔΙΑΒΑΣΕ α, β, γ

ΑΝ $(\alpha + \beta + \gamma) > 10$ **ΤΟΤΕ**

ΜΟ $\leftarrow (\alpha + \beta + \gamma) / 3$

ΓΡΑΨΕ 'ΜΟ=', ΜΟ

ΑΛΛΙΩΣ

ΑΝ $\alpha > \beta$ **ΚΑΙ** $\alpha > \gamma$ **ΤΟΤΕ**

Max \leftarrow α

ΑΛΛΙΩΣ_ΑΝ $\beta > \alpha$ **ΚΑΙ** $\beta > \gamma$ **ΤΟΤΕ**

Max \leftarrow β

ΑΛΛΙΩΣ

Max \leftarrow γ

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Max=', Max

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Αριθμοί3

<p>2. Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο να υπολογίζει το μισθό ενός εργαζομένου σε μια εταιρεία που δουλεύει με την ώρα. Να διαβάζει το όνομά του, τις ώρες εργασίας του για τον συγκεκριμένο μήνα και το ποσό που πληρώνεται για την κάθε ώρα. Οι κρατήσεις που του γίνονται εξαρτώνται από το ύψος του μισθού. Αν ο μισθός είναι μέχρι 1000 € το μήνα έχει 15% κρατήσεις, ενώ διαφορετικά έχει 20% κρατήσεις. Επίσης, αν είναι έγγαμος έχει ένα επίδομα 50 €. Να υπολογίζει και να εμφανίζει το τελικό μισθό που λαμβάνει ο υπάλληλος, όπως επίσης να τυπώνει και όλες οι κρατήσεις και επιδόματα που έχει.</p>	<p style="text-align: right;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Υπάλληλος ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: Ωρες, Επίδομα ΠΡΑΓΜΑΤΙΚΕΣ: Ωρομίσθιο, Μισθός, Κρατήσεις, Τελικός_Μισθός ΧΑΡΑΚΤΗΡΕΣ: Ον, Απ ΑΡΧΗ ΓΡΑΨΕ 'Όνοματεπώνυμο Υπαλ' ΔΙΑΒΑΣΕ Ον ΓΡΑΨΕ 'Ωρες εργασίας' ΔΙΑΒΑΣΕ Ωρες ΓΡΑΨΕ 'Ωρομίσθιο:' ΔΙΑΒΑΣΕ Ωρομίσθιο Μισθός ← Ωρομίσθιο * Ωρες ΓΡΑΨΕ 'Έγγαμος ΝΑΙ ή ΟΧΙ' ΔΙΑΒΑΣΕ Απ ΑΝ Απ ='ΝΑΙ' ΤΟΤΕ Επίδομα ← 50 ΑΛΛΙΩΣ Επίδομα ← 0 ΤΕΛΟΣ_ΑΝ ΓΡΑΨΕ 'Μισθός=', Μισθός ΑΝ Μισθός <=1000 ΤΟΤΕ Κρατήσεις ← Μισθός * 15/100 ΑΛΛΙΩΣ Κρατήσεις ← Μισθός * 20/100 ΤΕΛΟΣ_ΑΝ ΓΡΑΨΕ 'Κρατήσεις:', Κρατήσεις Τελικός_Μισθός ← Μισθός + Επίδομα - Κρατήσεις ΓΡΑΨΕ 'Τελικός Μισθός=', Τελικός_Μισθός ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Υπάλληλος</p>
<p>3. Σ' ένα σχολείο ένας μαθητής μπορεί να κάνει 60 δικαιολογημένες απουσίες και 55 αδικαιολόγητες, ενώ όσες δικαιολογημένες απουσίες κάνει πέρα των 60 θεωρούνται αδικαιολόγητες. Αν ο μαθητής κάνει πάνω των 55 αδικαιολόγητων απουσιών, τότε δίνει όλα τα μαθήματα τον Σεπτέμβριο. Αν ο μαθητής κάνει πάνω των 140 αδικαιολόγητων απουσιών, τότε επαναλαμβάνει τη τάξη. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει δικαιολογημένες και αδικαιολόγητες απουσίες ενός μαθητή και να εμφανίζει ένα από τα παρακάτω μηνύματα 'Ολική το Σεπτέμβρη' ή 'Επανάληψη τάξης' ή 'Προβιβασμός'.</p>	<p style="text-align: right;">Λύση</p> <p>ΠΡΟΓΡΑΜΜΑ Απουσίες ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: Δ,Α ΑΡΧΗ ΓΡΑΨΕ 'Δώσε (Δ)ικαιολογημένες και (Α)δικαιολόγητες απουσίες μαθητή:' ΔΙΑΒΑΣΕ Α,Δ ΑΝ Δ>60 ΤΟΤΕ A ← A+(Δ-60) ΤΕΛΟΣ_ΑΝ ΑΝ Α>140 ΤΟΤΕ ΓΡΑΨΕ 'Επανάληψη Τάξης' ΑΛΛΙΩΣ_ΑΝ Α>55 ΤΟΤΕ ΓΡΑΨΕ 'Ολική το Σεπτέμβρη' ΑΛΛΙΩΣ ΓΡΑΨΕ 'Προβιβασμός' ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Απουσίες</p>

4. Σε κάποια εξεταστική διαδικασία κάθε γραπτό αξιολογείται αρχικά από δύο βαθμολογητές και υπάρχει περίπτωση το γραπτό να χρειάζεται αναβαθμολόγηση από τρίτο βαθμολογητή. Στην περίπτωση αναβαθμολόγησης ο τελικός βαθμός υπολογίζεται ως εξής:
- ✓ Αν ο βαθμός του τρίτου βαθμολογητή είναι ίσος με το μέσο όρο (Μ.Ο.) των βαθμών των δύο πρώτων βαθμολογητών, τότε ο τελικός βαθμός είναι ο Μ.Ο.
 - ✓ Αν ο βαθμός του τρίτου βαθμολογητή είναι μικρότερος από το μικρότερο βαθμό MIN των δύο πρώτων βαθμολογητών, τότε ο τελικός βαθμός είναι ο MIN.
 - ✓ Διαφορετικά, ο τελικός βαθμός είναι ο μέσος όρος του βαθμού του τρίτου βαθμολογητή με τον πλησιέστερο προς αυτόν βαθμό των δύο πρώτων βαθμολογητών.

Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να υπολογίζει τον τελικό βαθμό ενός γραπτού με αναβαθμολόγηση, ο οποίος:

- 1) να διαβάζει τους βαθμούς του πρώτου, του δεύτερου και του τρίτου βαθμολογητή ενός γραπτού
 - 2) να υπολογίζει και να εμφανίζει το μεγαλύτερο (MAX) και το μικρότερο (MIN) από τους βαθμούς του πρώτου και του δεύτερου βαθμολογητή
 - 3) να υπολογίζει και να εμφανίζει τον τελικό βαθμό του γραπτού σύμφωνα με την παραπάνω διαδικασία.
- Παρατήρηση: Θεωρήστε ότι και οι τρεις βαθμοί είναι θετικοί ακέραιοι αριθμοί και δεν απαιτείται έλεγχος των δεδομένων.

Λύση

ΠΡΟΓΡΑΜΜΑ Μέσος_Όρος

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: B1, B2, B3, MAX, MIN, Τελικός_Βαθμός, Διαφορά1, Διαφορά 2

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε βαθμό Α', Β' και Γ' βαθμολογητή:'

ΔΙΑΒΑΣΕ B1, B2, B3

ΑΝ B1>B2 **ΤΟΤΕ**

MAX ← B1

MIN ← B2

ΑΛΛΙΩΣ

MAX ← B2

MIN ← B1

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ'Μεγαλύτερη βαθμολογία από τι δύο πρώτες η', MAX

ΓΡΑΨΕ'Μικρότερη βαθμολογία από τι δύο πρώτες η', MIN

MO ← (B1+B2)/2

ΑΝ B3=MO **ΤΟΤΕ**

Τελικός_Βαθμός ← MO

ΑΛΛΙΩΣ_ΑΝ B3<MIN **ΤΟΤΕ**

Τελικός_Βαθμός ← MIN

ΑΛΛΙΩΣ

Διαφορά1 ← A_T (B3-B1)

Διαφορά2 ← A_T (B3-B2)

ΑΝ Διαφορά1< Διαφορά2 **ΤΟΤΕ**

Τελικός_Βαθμός ← (B3+B1)/2

ΑΛΛΙΩΣ

Τελικός_Βαθμός ← (B3+B2)/2

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Τελικός Βαθμός=', Τελικός_Βαθμός

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μέσος_Όρος

5. Μια εταιρεία ταχυδρομικών υπηρεσιών εφαρμόζει για τα έξοδα αποστολής ταχυδρομικών επιστολών εσωτερικού και εξωτερικού, χρέωση σύμφωνα με τον παρακάτω πίνακα:

Βάρος επιστολής σε γρ.	Χρέωση εσωτερικού σε €	Χρέωση εξωτερικού σε €
Από 0 έως και 500	2	4.8
Από 500 έως και 1000	3.50	7.20
Από 1000 έως και 2000	4.60	11.50

Για παράδειγμα τα έξοδα αποστολής μιας επιστολής βάρους 800 γραμμαρίων και προορισμού εσωτερικού είναι 3,5 €.

Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο:

1. Να διαβάζει το βάρος της επιστολής.
2. Να διαβάζει τον προορισμό της επιστολής. Η τιμή 'ΕΣ' δηλώνει προορισμό εσωτερικού και η τιμή 'ΕΞ' δηλώνει προορισμό εξωτερικού.
3. Να υπολογίζει τα έξοδα αποστολής ανάλογα με τον προορισμό και το βάρος της επιστολής.

Λύση

ΠΡΟΓΡΑΜΜΑ Ταχυδρομείο
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Β, Απ, Χρ

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε βάρος επιστολής:'

ΔΙΑΒΑΣΕ Β

ΓΡΑΨΕ 'Δώσε προορισμό επιστολής'

ΓΡΑΨΕ '(ΕΣ)ωτερικό, (ΕΞ)ωτερικό'

ΔΙΑΒΑΣΕ Απ

ΑΝ Απ='ΕΣ' **ΤΟΤΕ**

ΑΝ Β>=0 ΚΑΙ Β<=500 **ΤΟΤΕ**

Χρ ← 2

ΑΛΛΙΩΣ_ΑΝ Β<=1000 **ΤΟΤΕ**

Χρ ← 3.5

ΑΛΛΙΩΣ

Χρ ← 4.6

ΤΕΛΟΣ_ΑΝ

ΑΛΛΙΩΣ

ΑΝ Β>=0 ΚΑΙ Β<=500 **ΤΟΤΕ**

Χρ ← 4.8

ΑΛΛΙΩΣ_ΑΝ Β<=1000 **ΤΟΤΕ**

Χρ ← 7.2

ΑΛΛΙΩΣ

Χρ ← 11.5

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Έξοδα αποστολής:', Χρ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ταχυδρομείο

6. Κάποια δημοτική αρχή ακολουθεί την εξής τιμολογική πολιτική για την κατανάλωση νερού ανά μήνα: Χρεώνει πάγιο ποσό 12€ και εφαρμόζει κλιμακωτή χρέωση σύμφωνα με τον παρακάτω πίνακα:

Κατανάλωση νερού σε κυβικά	Χρέωση ανά κυβικό
Από 0 έως και 5	δωρεάν
Από 5 έως και 10	0,5 €
Από 10 έως και 20	0,7 €
Από 20 και άνω	1,00€

Στο ποσό που προκύπτει από την αξία του νερού και το πάγιο υπολογίζεται ο ΦΠΑ με συντελεστή 23%. Το τελικό ποσό προκύπτει από την άθροιση της αξίας του νερού, το πάγιο, το ΦΠΑ και το δημοτικό φόρο που είναι 10€.

Να γραφεί πρόγραμμα σε ΓΛΩΣΣΑ, το οποίο :

1. Να διαβάζει τη μηνιαία κατανάλωση του νερού.
2. Να υπολογίζει την αξία του νερού που καταναλώθηκε σύμφωνα με την παραπάνω τιμολογική πολιτική.
3. Να υπολογίζει το ΦΠΑ.
4. Να υπολογίζει και να εμφανίζει το τελικό ποσό.

Λύση

ΠΡΟΓΡΑΜΜΑ Νερό

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Κ, Αξία, Τ_Αξία, ΦΠΑ,

Τελ_Ποσό

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε μηνιαία κατανάλωση νερού:'

ΔΙΑΒΑΣΕ Κ

ΑΝ Κ>=0 ΚΑΙ Κ<=5 **ΤΟΤΕ**

Αξία ← 0

ΑΛΛΙΩΣ_ΑΝ Κ<=10 **ΤΟΤΕ**

Αξία ← (Κ-5) * 0.5

ΑΛΛΙΩΣ_ΑΝ Κ<=20 **ΤΟΤΕ**

Αξία ← 5* 0.5+(Κ-10) * 0.7

ΑΛΛΙΩΣ

Αξία ← 5* 0.5+10* 0.7+(Κ-20) *

ΤΕΛΟΣ_ΑΝ

Τ_Αξία ← Αξία+12

ΦΠΑ ← Τ_Αξία *23/100

Τελ_Ποσό ← Τ_Αξία+ΦΠΑ+10

ΓΡΑΨΕ 'Ποσό πληρωμής νερού:', Τελ_Ποσό
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Νερό

7. Στο τέλος κάθε έτους οι ιδιοκτήτες των αυτοκινήτων προπληρώνουν τα Τέλη Κυκλοφορίας για την επόμενη χρονιά, σύμφωνα με τον κυβισμό των αυτοκινήτων. Το ποσό που αναλογεί για το κάθε αυτοκίνητο ανάλογα με τον κυβισμό του, φαίνεται στον παρακάτω πίνακα:

Κυβισμός	Τέλη Κυκλοφορίας
Έως 300	22 €
301 - 785	55 €
786 - 1.071	120 €
1.072 - 1.357	135 €
1.358 - 1.548	255 €
1.549 - 1.738	280 €
1.739 - 1.928	320 €
1.929 - 2.357	690 €
2.358 - 3.000	920 €
3.001- 4.000	1.150 €
άνω των 4.000	1.380 €

Να γραφεί το πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει τον κυβισμό ενός αυτοκινήτου και να υπολογίζει το φόρο που του αναλογεί, στην συνέχεια να τυπώνεται το ποσό του φόρου με κατάλληλο μήνυμα.

Λύση

ΠΡΟΓΡΑΜΜΑ Φόρος_Αυτοκινήτου

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Κ,Φ

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε κυβισμό αυτοκινήτου'

ΔΙΑΒΑΣΕ Κ

ΑΝ Κ<=300 **TOTE**

Φ ← 22

ΑΛΛΙΩΣ_ΑΝ Κ<=785 **TOTE**

Φ ← 55

ΑΛΛΙΩΣ_ΑΝ Κ<=1071 **TOTE**

Φ ← 120

ΑΛΛΙΩΣ_ΑΝ Κ<=1357 **TOTE**

Φ ← 135

ΑΛΛΙΩΣ_ΑΝ Κ<=1548 **TOTE**

Φ ← 255

ΑΛΛΙΩΣ_ΑΝ Κ<=1738 **TOTE**

Φ ← 280

ΑΛΛΙΩΣ_ΑΝ Κ<=1928 **TOTE**

Φ ← 320

ΑΛΛΙΩΣ_ΑΝ Κ<=2357 **TOTE**

Φ ← 690

ΑΛΛΙΩΣ_ΑΝ Κ<=3000 **TOTE**

Φ ← 920

ΑΛΛΙΩΣ_ΑΝ Κ<=4000 **TOTE**

Φ ← 1150

ΑΛΛΙΩΣ

Φ ← 1380

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Φόρος=',Φ, '€'

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Φόρος_Αυτοκινήτου

2.4 Μη λυμένες ασκήσεις

1. Να δημιουργήσετε τον πίνακα τιμών των μεταβλητών του δίπλα προγράμματος.

ΠΡΟΓΡΑΜΜΑ Α1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Α, Β, Γ, χ, ψ

ΑΡΧΗ

χ ← 100

ψ ← 200

Α ← χ+1

Β ← 3 * χ

Γ ← Α+Β

ΑΝ C>400 **TOTE**

Α ← Β

Β ← χ

ΑΛΛΙΩΣ

ΑΝ B>200 **TOTE**

Β ← Α

Α ← 20

ΑΛΛΙΩΣ

Α ← Γ

Β ← Α

ΤΕΛΟΣ_ΑΝ


ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Α1

2. Τι θα εμφανίσουν τα παρακάτω προγράμματα αν σαν είσοδο δοθούν οι τιμές 5 και 10, στις μεταβλητές χ και ψ αντίστοιχα;

<p>ΠΡΟΓΡΑΜΜΑ Π1 ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: χ, ψ ΑΡΧΗ ΓΡΑΨΕ 'Δώσε τιμή για χ και ψ:' ΔΙΑΒΑΣΕ χ, ψ $\chi \leftarrow \chi \text{ div } 2$ $\psi \leftarrow \psi \text{ mod } 3$ ΑΝ $\chi = \psi$ ΤΟΤΕ $\chi \leftarrow \chi^2 + 1$ $\psi \leftarrow 2 * \psi - 4^2$ ΑΛΛΙΩΣ $\chi \leftarrow \chi \text{ mod } 2$ $\psi \leftarrow \psi \text{ div } 3$ ΤΕΛΟΣ_ΑΝ ΓΡΑΨΕ χ =', χ, 'ψ=', ψ ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Π1</p>	<p>ΠΡΟΓΡΑΜΜΑ Π2 ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: χ, ψ ΛΟΓΙΚΕΣ: f ΑΡΧΗ ΓΡΑΨΕ 'Δώσε 2 ακέραιες τιμές:' ΔΙΑΒΑΣΕ χ, ψ $\chi \leftarrow \chi^2 + 1$ $\psi \leftarrow 2 * \psi + \chi^2$ $f \leftarrow (\chi \text{ mod } 2 = 0)$ ΚΑΙ ($\psi \text{ mod } 2 = 0$) ΑΝ f = ΑΛΗΘΗΣ ΤΟΤΕ ΓΡΑΨΕ 'Άρτιοι:', χ, ψ ΤΕΛΟΣ_ΑΝ ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Π2</p>
--	--

3. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει από το πληκτρολόγιο τρεις ακέραιους αριθμούς. Στη συνέχεια υπολογίζει και εμφανίζει τον αριθμό που είναι ανάμεσα στους άλλους δύο. Δηλαδή, αν δώσω τους αριθμούς 107,53,42 τότε να εμφανιστεί ο 53, γιατί $107 > 53 > 42$ ή $42 < 53 < 107$.
4. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει το όνομα ενός υπαλλήλου, τις ώρες εργασίας και το ωρομίσθιο του. Οι κρατήσεις που γίνονται στον υπάλληλο είναι 15% αν ο μισθός του είναι μέχρι 1.000 €, διαφορετικά είναι 20%. Σε περίπτωση που ο υπάλληλος έχει λιγότερο από 15 ώρες εργασίας τότε οι παραπάνω κρατήσεις μειώνονται κατά 2%.

 Ο μισθός κάθε υπαλλήλου υπολογίζεται από τον τύπο:
 $\text{Ωρες} \times \text{Ωρομίσθιο} - \text{Κρατήσεις}$
Και οι κρατήσεις από τον τύπο:
 $\text{Κρατήσεις} = (\text{Ωρες} \times \text{Ωρομίσθιο}) \times \text{Ποσοστό} / 100$

5. Σε κάποιο σχολείο η φοίτηση ενός μαθητή χαρακτηρίζεται 'επαρκής' εάν το σύνολο όλων των απουσιών του δεν ξεπερνά τις 100, εκ των οποίων οι 60 είναι δικαιολογημένες ή το σύνολο των απουσιών του δεν ξεπερνά τις 160 και οι 120 είναι δικαιολογημένες και ο Μέσος όρος τους μεγαλύτερος του 15. Διαφορετικά η φοίτηση είναι 'ανεπαρκής'
Να γίνει το πρόγραμμα σε ΓΛΩΣΣΑ, που:
- 1) Να διαβάζει από το πληκτρολόγιο το Μέσο Όρο του μαθητή, το σύνολο Δικαιολογημένων και το σύνολο Αδικαιολόγητων απουσιών του.
 - 2) Να εμφανίζει το μήνυμα 'Φοίτηση επαρκής' ή 'Φοίτηση ανεπαρκής', ανάλογα με τις απουσίες του.
6. Το Υπουργείο Οικονομικών φορολογεί τους Ελεύθερους Επαγγελματίες σύμφωνα με τον παρακάτω πίνακα:

Από 1€ μέχρι 20.000€	22%
Από 20.001€ μέχρι 30.000€	29%
Από 30.001€ μέχρι 40.000€	37%
Από 40.001€ και άνω	45%

Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει το εισόδημα ενός ελεύθερου επαγγελματία και στη συνέχεια να υπολογίζει και να εμφανίζει το φόρο που του αναλογεί. Ο υπολογισμός φόρου γίνεται σύμφωνα με τον παραπάνω πίνακα, κλιμακωτά.

7. Με το σύστημα πληρωμής των διοδίων, οι οδηγοί των τροχοφόρων έχουν τη δυνατότητα να πληρώνουν το αντίτιμο των διοδίων με ειδική μαγνητική κάρτα. Υποθέστε ότι υπάρχει μηχανήμα το οποίο διαθέτει είσοδο για την κάρτα και φωτοκύτταρο. Το μηχανήμα διαβάζει από την κάρτα το υπόλοιπο των χρημάτων και το αποθηκεύει σε μια μεταβλητή $Υ$ και με το φωτοκύτταρο αναγνωρίζει τον τύπο του τροχοφόρου και το αποθηκεύει σε μια μεταβλητή $Τ$. Υπάρχουν τρεις τύποι τροχοφόρων : δίκυκλα (Δ), επιβατικά ($Ε$) και φορτηγά (Φ), με αντίτιμο διοδίων 1.20€, 2.40€ και 3.50€ αντίστοιχα.

Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο:

- 1) Να ελέγχει τον τύπο του τροχοφόρου και εκχωρεί στη μεταβλητή A το αντίτιμο των διοδίων, ανάλογα με τον τύπο του τροχοφόρου.
 - 2) Να ελέγχει την πληρωμή των διοδίων με τον παρακάτω τρόπο:
 - Αν το υπόλοιπο της κάρτας επαρκεί για την πληρωμή του αντιτίμου των διοδίων, αφαιρεί το ποσό αυτό από την κάρτα.
 - Αν η κάρτα δεν έχει υπόλοιπο, το μηχανήμα ειδοποιεί με μήνυμα για το ποσό που πρέπει να πληρωθεί.
 - Αν το υπόλοιπο δεν επαρκεί, μηδενίζεται η κάρτα και δίνεται με μήνυμα το ποσό που απομένει να πληρωθεί.
8. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να δίνει από το πληκτρολόγιο έναν ακέραιο αριθμό και να ελέγχει εάν είναι διψήφιος και θετικός. Στη περίπτωση που ισχύει να τυπώνεται το μήνυμα 'Θετικός, Διψήφιος αριθμός', ενώ διαφορετικά να τυπώνονται για κάθε περίπτωση 'Μόνο θετικός' ή 'Μόνο διψήφιος' ή 'Ούτε θετικός, ούτε διψήφιος'.
9. Ένα μαγαζί με ηλεκτρονικά παιχνίδια χρεώνει τους πελάτες του κλιμακωτά, με το πόση ώρα χρησιμοποιούν τα παιχνίδια, σύμφωνα με τον παρακάτω πίνακα:

Χρόνος σε λεπτά	Χρέωση ανά λεπτό
60	0,15€
90	0,22€
120	0,37€
180	0,42€
>180	0,55€

Να γραφεί το πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει το χρόνο χρήσης του ηλεκτρονικού παιχνιδιού και στη συνέχεια να υπολογίζει και να εμφανίζει το ποσό χρέωσης, σύμφωνα με τον παραπάνω πίνακα.

10. Ένα ταξιδιωτικό γραφείο οργανώνει εκδρομές 3ήμερες και 5ήμερες στην Κρήτη, Ρόδο και Μύκονο με αεροπλάνο ή πλοίο. Οι τιμές για κάθε περίπτωση, βρίσκονται στον παρακάτω πίνακα.

Προορισμός	Αεροπλάνο		Πλοίο	
	3ήμερη	5ήμερη	3ήμερη	5ήμερη
Κρήτη	400€	600€	550€	750€
Ρόδος	500€	700€	700€	900€
Μύκονος	550€	650€	650€	850€

Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ», το οποίο να διαβάζει τον προορισμό, το μέσο μεταφοράς (Π =πλοίο ή A =Αεροπλάνο), τις ημέρες διαμονής (3 ή 5) και το πλήθος των ατόμων. Στη συνέχεια υπολογίζει και εμφανίζει το κόστος της εκδρομής

ΟΣΟ ... ΕΠΑΝΑΛΑΒΕ ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ ... ΜΕΧΡΙΣ_ΟΤΟΥ ... ΓΙΑ ... ΑΠΟ ... ΜΕΧΡΙ ...

Μετατροπές μεταξύ των δομών επανάληψης

3. ΔΟΜΗ ΕΠΑΝΑΛΗΨΗΣ

Η «ΓΛΩΣΣΑ» υποστηρίζει τρεις εντολές επανάληψης:

Την εντολή «**ΟΣΟ...**» που εκτελεί τις εντολές που περικλείει όσο η συνθήκη είναι ΑΛΗΘΗΣ και αυτή γράφεται στην αρχή της εντολής.

Την εντολή «**ΜΕΧΡΙΣ_ΟΤΟΥ...**» που εκτελεί τις εντολές, που περικλείει, όσο η συνθήκη είναι ΨΕΥΔΗΣ και αυτή γράφεται στο τέλος της εντολής.

Την εντολή «**ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ...**» που εκτελεί τις εντολές που περικλείει, προκαθορισμένες φορές

Πού χρησιμοποιούνται οι δομές επανάληψης;

Χρησιμοποιούνται στη περίπτωση που έχουμε επαναλαμβανόμενες εντολές. Υπάρχουν τρεις μορφές επανάληψης γιατί η κάθε μια καλύπτει διαφορετικά είδη επανάληψης. Δηλαδή, όταν είναι γνωστό πόσες φορές γίνεται η επανάληψη χρησιμοποιούμε τη «ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ...», εάν δεν ξέρουμε το πλήθος των επαναλήψεων και η επανάληψη να σταματά με μια ερώτηση ή με το πάτημα ενός «κουμπιού» - επανάληψη υπό συνθήκη - τότε χρησιμοποιούμε την «ΟΣΟ... ή την ΜΕΧΡΙΣ_ΟΤΟΥ...».

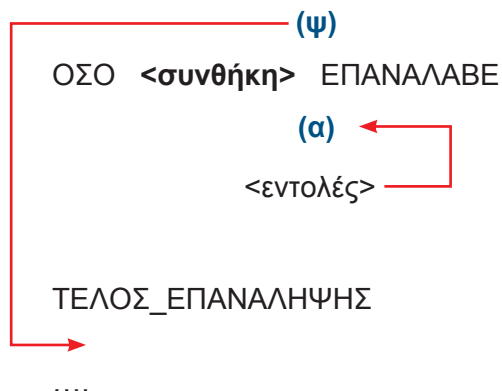
Η ΟΣΟ... χρησιμοποιείται στις περιπτώσεις που έχουμε τιμή φρουρό, δηλαδή η επανάληψη σταματά όταν πληκτρολογήσουμε 0 ή αρνητικό αριθμό ή # κλπ. ενώ σε όλες τις άλλες περιπτώσεις χρησιμοποιείται η ΜΕΧΡΙΣ_ΟΤΟΥ...

Παρακάτω θα ελέγξουμε αναλυτικά την κάθε μια ξεχωριστά.

3.1 Η εντολή επανάληψης ΟΣΟ.,ΕΠΑΝΑΛΑΒΕ

Οι εντολές, που περιλαμβάνει η εντολή «**ΟΣΟ...ΕΠΑΝΑΛΑΒΕ**», επαναλαμβάνονται όσο η συνθήκη είναι ΑΛΗΘΗΣ και μόνο όταν η συνθήκη γίνει ΨΕΥΔΗΣ σταματά η επανάληψή τους και εκτελείται η αμέσως επόμενη εντολή, μετά το «**ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ**».

Σύνταξη:



3.1.1 Χαρακτηριστικά της εντολής:

- Επειδή η συνθήκη βρίσκεται στην αρχή της εντολής, υπάρχει περίπτωση να «βγει» ΨΕΥΔΗΣ από την αρχή, οπότε και να μην εκτελεστούν καμία φορά οι εντολές της επανάληψης.
- Αν η συνθήκη δεν «βγει» ΨΕΥΔΗΣ ποτέ, τότε οι εντολές της επανάληψης εκτελούνται συνεχώς και το πρόγραμμα δεν τελειώνει ποτέ. Τότε λέμε ότι έχουμε ατέρμονα βρόχο (**ατέρμων βρόχος**).

3.1.2 Πώς συντάσσεται:

- Δίνουμε **Αρχική Τιμή στη μεταβλητή της συνθήκης**, γιατί η συνθήκη βρίσκεται στην αρχή της ΟΣΟ και πρέπει η μεταβλητή να έχει αρχικοποιηθεί για να μπορεί να εκτελέσει ο υπολογιστής τη συνθήκη και να δώσει στη μεταβλητή τη τιμή ΑΛΗΘΗΣ ή ΨΕΥΔΗΣ
- Πριν τελειώσει η επανάληψη, δηλαδή πριν το «ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ» πρέπει να **αλλάξουμε τη τιμή της μεταβλητής της συνθήκης**, για να μην έχουμε ατέρμονα βρόχο.

1^η περίπτωση

```
Αρχική τιμή ΔΙΑΒΑΣΕ α
ΟΣΟ α>0 ΕΠΑΝΑΛΑΒΕ
    ΓΡΑΨΕ α
    ...
Αρχική τιμή ΔΙΑΒΑΣΕ α
ΤΕΛΟΣ ΕΠΑΝΑΛΗΨΗΣ
```

Στο παράδειγμα αυτό η μεταβλητή της συνθήκης είναι η α .

Η μεταβλητή αυτή παίρνει τιμές από το πληκτρολόγιο, γι' αυτό η αρχική τιμή της είναι η εντολή «ΔΙΑΒΑΣΕ α » την πρώτη φορά πριν την εντολή «ΟΣΟ...», για να μπορέσει να ελεγχθεί η συνθήκη « $\alpha > 0$ ».

Η «ΔΙΑΒΑΣΕ α » ξαναγράφεται πριν το «ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ» για να αλλάζει τη τιμή της α , με τη νέα τιμή που πληκτρολογούμε.

Η παραπάνω επανάληψη κάνει το εξής: **'Πληκτρολογούμε θετικούς αριθμούς'**.

Στο παράδειγμα αυτό η μεταβλητή της συνθήκης είναι η α .

2η περίπτωση

```
Αρχική τιμή α←100
ΟΣΟ α<= 1000 ΕΠΑΝΑΛΑΒΕ ΓΡΑΨΕ α
    ...
Αρχική τιμή α←α+2
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Η μεταβλητή αυτή παίρνει τιμές με εντολή καταχώρησης. Αρχική τιμή της α είναι το 100, έτσι πριν το ΟΣΟ γράφουμε την εντολή $\alpha \leftarrow 100$, για να μπορέσει να ελεγχθεί η συνθήκη $\alpha < 1000$. Η αλλαγή της μεταβλητής α γίνεται πριν το τέλος της επανάληψης με την « $\alpha \leftarrow \alpha + 2$ », έτσι ώστε να αλλάξει τη τιμή της α και να μην έχουμε ατέρμονα βρόχο.

Η παραπάνω επανάληψη κάνει το εξής: Τυπώνει όλους τους άρτιους αριθμούς από το 100 μέχρι και το 1000.

3.1.3 Πού χρησιμοποιείται:

Σε όλες τις περιπτώσεις που έχουμε επανάληψη υπό συνθήκη, η οποία σταματά με μια τιμή φρουρό.

3.1.3 Παραδείγματα

Πληκτρολογούμε θετικούς αριθμούς, η πληκτρολόγηση σταματά όταν δώσουμε αρνητικό αριθμό ή τον αριθμό «0». Στο τέλος τυπώνεται το άθροισμα των θετικών αριθμών.

Λύση

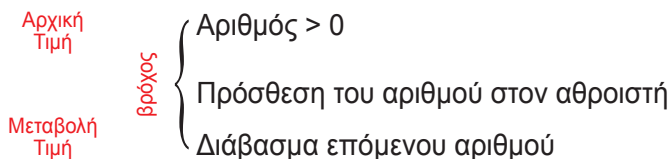
Ανάλυση:

1. Εφόσον πληκτρολογούμε συνέχεια θετικούς αριθμούς, έχουμε επανάληψη.
2. Η πληκτρολόγηση σταματά με ένα αρνητικό αριθμό ή το μηδέν, άρα η επανάληψη είναι με συνθήκη που ελέγχει αν ο αριθμός που πληκτρολογήσαμε είναι θετικός (. (Τιμή φρουρός: αριθμός>0).
3. Επειδή χρειαζόμαστε το άθροισμα των θετικών αριθμών, που δεν είναι αποθηκευμένοι σε διαφορετικές μεταβλητές αλλά σε μια, θα χρησιμοποιήσουμε αθροιστή, που σημαίνει μηδενισμός του αθροιστή στην αρχή του προγράμματος.

Γενική μορφή:

Μηδενισμός Αθροιστή

Διάβασμα του 1^{ου} αριθμού



Εκτύπωση αθροίσματος

ΠΡΟΓΡΑΜΜΑ Άθροισμα

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: sum, x

ΑΡΧΗ

sum <- 0

ΔΙΑΒΑΣΕ x

ΟΣΟ x > 0 **ΕΠΑΝΑΛΑΒΕ**

sum <- sum + x

ΔΙΑΒΑΣΕ x

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ sum

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Άθροισμα

3.2 Η εντολή επανάληψης ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ...

Με την εντολή «ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ» οι εντολές που περιλαμβάνει επαναλαμβάνονται μέχρις ότου η συνθήκη γίνει αληθής. Δηλαδή, όσο η συνθήκη είναι ΨΕΥΔΗΣ έχουμε επανάληψη, ενώ όταν η συνθήκη γίνει ΑΛΗΘΗΣ η επανάληψη σταματά και εκτελείται η αμέσως επόμενη εντολή μετά το «ΜΕΧΡΙΣ_ΟΤΟΥ...».

3.2.1. Σύνταξη:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

<εντολές>

(ψ)

ΜΕΧΡΙΣ_ΟΤΟΥ <συνθήκη>

(α)

.....

3.2.2. Χαρακτηριστικά της εντολής:

- Επειδή η συνθήκη στην εντολή επανάληψης «ΜΕΧΡΙΣ_ΟΤΟΥ...» βρίσκεται στο τέλος τη εντολής, οι εντολές, που περιλαμβάνονται σ' αυτήν, θα εκτελεστούν τουλάχιστον μια φορά.
- Οι εντολές επανάληψης «ΟΣΟ...» και «ΜΕΧΡΙΣ_ΟΤΟΥ...» είναι εντολές επανάληψης υπό συνθήκη. δηλαδή τις χρησιμοποιούμε όπου έχουμε επανάληψη εντολών και η επανάληψη σταματά όταν ισχύει ή δεν ισχύει μια συνθήκη. Επίσης μπορεί η μια να αντικατασταθεί από την άλλη σ' όλες τις περιπτώσεις επανάληψης υπό συνθήκη. Όπως θα δούμε και παρακάτω. Όμως, ανάλογα με το πρόβλημα, μόνο μια από τις δύο είναι η βέλτιστη λύση.

3.2.3. Που χρησιμοποιείται:

- Στα προβλήματα που έχουν *Μενού επιλογής*.
- Στα προβλήματα που έχουμε *Έλεγχο δεδομένων από το πληκτρολόγιο*.
- Στα προβλήματα που θέλουμε να επαναλάβουμε μια ολόκληρη επεξεργασία (π.χ. Αναζήτηση δεδομένων) απαντώντας στην ερώτηση *'Θέλεις να συνεχίσεις (N/O)?'*.

Στις παραπάνω τρεις περιπτώσεις, που αναφέρθηκαν για επιλογή της εντολής «ΜΕΧΡΙΣ_ΟΤΟΥ...», μπορεί να χρησιμοποιηθεί και η «ΟΣΟ...» αλλά δεν μας συμφέρει γιατί γράφονται περισσότερες εντολές.

3.2.4. Παραδείγματα

1. Έλεγχος δεδομένων

Να δώσετε από το πληκτρολόγιο το μέσο προφορικό βαθμό και το γραπτό βαθμό ενός μαθητή. Να υπολογισθεί και να τυπωθεί ο μέσος όρος του μαθητή. Να γίνεται έλεγχος για τους βαθμούς, ώστε να βρίσκονται στο διάστημα [0,20].

Λύση

Ανάλυση:

1. Διαβάζετε προφορικό βαθμό **με έλεγχο**.
2. Διαβάζετε τον γραπτό βαθμό **με έλεγχο**.
3. Υπολογίζετε τον Μέσο Όρο του μαθητή.

ΠΡΟΓΡΑΜΜΑ Έλεγχος

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ: Πρ, Γρ, ΜΟ

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε προφορικό βαθμό:'

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Πρ

ΜΕΧΡΙΣ_ΟΤΟΥ Πρ <= 20 ΚΑΙ Πρ >= 0

ΓΡΑΨΕ 'Δώσε γραπτό βαθμό''

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Γρ

ΜΕΧΡΙΣ_ΟΤΟΥ Γρ <= 20 ΚΑΙ Γρ >= 0

ΜΟ ← (Πρ + Γρ) / 2

ΓΡΑΨΕ 'Μέσος Όρος=', Μο

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Έλεγχος

Έλεγχος βαθμού
από και 0 έως και
20

Παρατήρηση:

Οι εντολές:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Πρ

ΜΕΧΡΙΣ_ΟΤΟΥ Πρ <= 20 ΚΑΙ Πρ >= 0

Διαβάζετε από το πληκτρολόγιο έναν αριθμό, λόγω της «ΔΙΑΒΑΣΕ Πρ», αν ο αριθμός βρίσκεται στο διάστημα [0,20] τότε το πρόγραμμα εκτελεί την εντολή μετά το «ΜΕΧΡΙΣ_ΟΤΟΥ...», δηλαδή η συνθήκη βγαίνει ΑΛΗΘΗΣ. Εάν όμως ο αριθμός δεν βρίσκεται στο διάστημα [0,20], δηλ. η συνθήκη βγει ΨΕΥΔΗΣ, τότε επαναλαμβάνεται η εντολή και το πρόγραμμα περιμένει να ξαναπληκτρολογήσουμε το βαθμό. Αυτό σημαίνει ότι το πρόγραμμα δεν μας επιτρέπει να προχωρήσουμε αν δεν πληκτρολογήσουμε τα σωστά δεδομένα.

Οι εντολές:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Πρ

ΑΝ Πρ < 0 Ή Πρ > 20 **ΤΟΤΕ**

ΓΡΑΨΕ 'Λάθος βαθμός. Ξαναπροσπάθησε!'

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ Πρ <= 20 ΚΑΙ Πρ >= 0

Οι παραπάνω εντολές κάνουν ακριβώς το ίδιο με αυτές στην προηγούμενη περίπτωση, αλλά τυπώνεται και μήνυμα λάθους στην περίπτωση λάθους πληκτρολόγησης.

2. Μενού επιλογής

Δίνετε από το πληκτρολόγιο 2 αριθμούς ακέραιους και πληκτρολογείτε στη συνέχεια την επιλογή της πράξης που θα εκτελεστεί για τους παραπάνω αριθμούς, σύμφωνα με το μενού:

1. Πρόσθεση
2. Αφαίρεση
3. Πολλαπλασιασμός
4. Διαίρεση
5. Έξοδος

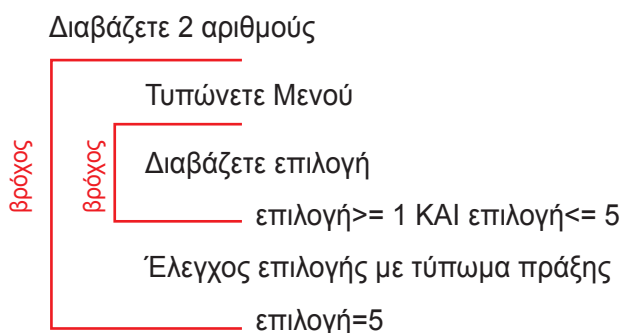
Όταν επιλέξετε '5' τότε το πρόγραμμα τελειώνει, διαφορετικά μπορείτε να επιλέξετε πάλι μια νέα πράξη. Να γίνεται έλεγχος της επιλογής για τα νούμερα 1,2,3,4 και 5.

Λύση

Ανάλυση:

1. Διαβάζετε τους 2 αριθμούς.
2. ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ.
3. Τυπώνετε το ΜΕΝΟΥ.
4. Διαβάζετε επιλογή με έλεγχο.
5. Αν επιλογή = 1 τότε προσθέτετε.
6. Αν επιλογή = 2 τότε αφαιρείτε.
7. Αν επιλογή = 3 τότε πολλαπλασιάζετε.
8. Αν επιλογή = 4 τότε διαιρείτε.
9. Αν επιλογή = 5 ΤΕΛΟΣ προγράμματος αλλιώς επιστρέφει στο Βήμα 2.

Γενική μορφή:



ΠΡΟΓΡΑΜΜΑ Μενού

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ: α, β, Αρ

ΑΚΕΡΑΙΕΣ: Ε

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε δύο ακέραιους αριθμούς:'

ΔΙΑΒΑΣΕ α, β

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ '1. Πρόσθεση'

ΓΡΑΨΕ '2. Αφαίρεση'

ΓΡΑΨΕ '3. Πολλαπλασιασμός'

ΓΡΑΨΕ '4. Διαίρεση'

ΓΡΑΨΕ '5. Έξοδος'

ΓΡΑΨΕ 'Επιλογή:'

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΔΙΑΒΑΣΕ Ε

ΜΕΧΡΙΣ_ΟΤΟΥ Ε >=1 ΚΑΙ Ε <= 5

ΑΝ Ε = 1 ΤΟΤΕ

Αρ <- α + β

ΓΡΑΨΕ 'Πρόσθεση=', Αρ

ΑΛΛΙΩΣ_ΑΝ Ε = 2 ΤΟΤΕ

Αρ <- α - β

ΓΡΑΨΕ 'Αφαίρεση=', Αρ

ΑΛΛΙΩΣ_ΑΝ Ε = 3 ΤΟΤΕ

Αρ <- α * β

ΓΡΑΨΕ 'Πολλαπλασιασμός=', Αρ

ΑΛΛΙΩΣ_ΑΝ Ε = 4 ΤΟΤΕ

ΑΝ β <> 0 **ΤΟΤΕ**

Αρ <- α / β

ΓΡΑΨΕ 'Διαίρεση=', Αρ

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Η διαίρεση δεν γίνεται'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΜΕΧΡΙΣ_ΟΤΟΥ Ε = 5

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μενού

3. Επανάληψη μετά από ερώτηση

Να διαβάζετε ονοματεπώνυμο και 3 βαθμούς ενός μαθητή. Να υπολογίζετε και να τυπώνετε το ΜΟ και το μήνυμα 'ΕΠΙΤΥΧΩΝ', αν $ΜΟ > 10$, διαφορετικά 'ΑΠΟΤΥΧΩΝ'. Στη συνέχεια να βγαίνει το μήνυμα: 'Θέλεις να συνεχιστεί για άλλο μαθητή (Ν/Ο)?' Αν πληκτρολογήσετε 'Ν' τότε να επαναλαμβάνονται τα παραπάνω και για άλλο μαθητή, διαφορετικά να τελειώνει το πρόγραμμα.

Λύση

Ανάλυση:

1. ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ.
2. Διαβάζετε όνομα μαθητή.
3. Διαβάζετε τους τρεις βαθμούς.
4. Υπολογίζετε τον Μέσο Όρο.
5. Τυπώνετε τον Μέσο Όρο και Έλεγχος Μέσου Όρου.
6. Τυπώνετε το μήνυμα: 'Θέλεις να συνεχίσεις για άλλο μαθητή (Ν/Ο)?'
7. Διαβάζετε την Απάντηση .
8. Αν η Απάντηση είναι 'Ν' τότε επαναλαμβάνεται το πρόγραμμα από το 1ο Βήμα.

ΠΡΟΓΡΑΜΜΑ Μαθητές

ΜΕΤΑΒΛΗΤΕΣ

ΠΡΑΓΜΑΤΙΚΕΣ: Β1, Β2, Β3, ΜΟ

ΧΑΡΑΚΤΗΡΕΣ: Απ, Ον

ΑΡΧΗ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Όνοματεπώνυμο:'

ΔΙΑΒΑΣΕ Ον

ΓΡΑΨΕ 'Δώσε 3 βαθμούς:'

ΔΙΑΒΑΣΕ Β1, Β2, Β3

$ΜΟ \leftarrow (Β1 + Β2 + Β3) / 3$

ΑΝ ΜΟ ≥ 10 **ΤΟΤΕ**

ΓΡΑΨΕ ΜΟ, 'ΕΠΙΤΥΧΩΝ'

ΑΛΛΙΩΣ

ΓΡΑΨΕ ΜΟ, 'ΑΠΟΤΥΧΩΝ'

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Θέλεις να συνεχίσεις για άλλο μαθητή (Ν/Ο)?'

ΔΙΑΒΑΣΕ Απ

ΜΕΧΡΙΣ_ΟΤΟΥ Απ = 'Ο'

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μαθητές

3.3 Η εντολή επανάληψης ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ ...

Η εντολή επανάληψης «ΓΙΑ...ΑΠΟ...ΜΕΧΡΙ ...» χρησιμοποιείται όταν είναι γνωστό το πλήθος των επαναλήψεων, π.χ. να βρούμε το ΜΟ 35 μαθητών.

Η εντολή αυτή έχει σχεδιαστεί για να απλοποιήσει τον έλεγχο των επαναλήψεων, αν και μπορούν όλα τα προβλήματα να υλοποιηθούν με τις άλλες δύο εντολές επανάληψης υπό συνθήκη.

Σύνταξη:

ΓΙΑ <μεταβλητή> ΑΠΟ <τιμή1> ΜΕΧΡΙ <τιμή2> ΜΕ_ΒΗΜΑ χ

.
.
<εντολές>
.

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

3.3.1. Τρόπος εκτέλεσης

Η εντολή αυτή χειρίζεται μια <μεταβλητή> στην οποία καταχωρείται η <τιμή1> αρχικά. Η τιμή που παίρνει η <μεταβλητή> συγκρίνεται με τη <τιμή2> και εφόσον είναι μικρότερη απ' αυτήν, τότε εκτελούνται οι εντολές που βρίσκονται στο βρόχο. Στη συνέχεια η μεταβλητή αυξάνεται κατά την τιμή χ που ορίζει το ΒΗΜΑ. Αν η νέα τιμή είναι μικρότερη της τελικής, τότε ο βρόχος εκτελείται ξανά. Η διαδικασία επαναλαμβάνεται μέχρι η <μεταβλητή> γίνει μεγαλύτερη από την <τιμή2> αλλιώς το πρόγραμμα συνεχίζει με την εντολή μετά το «ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ».

Ο αριθμός του βήματος μπορεί να είναι θετικός ή αρνητικός και αυτό καθορίζει τον έλεγχο για την επανάληψη. Έτσι έχουμε τις περιπτώσεις:

1. **Βήμα = 1** : παραλείπεται το ΜΕ_ΒΗΜΑ 1 και η μεταβλητή αυξάνεται κατά +1.
2. **Βήμα > 0** : τότε πρέπει να ισχύει $\text{τιμή1} \leq \text{τιμή2}$ για να εκτελεστεί τουλάχιστον μια φορά ο βρόχος.
3. **Βήμα < 0** : τότε πρέπει να ισχύει $\text{τιμή1} \geq \text{τιμή2}$ για να εκτελεστεί τουλάχιστον μια φορά ο βρόχος.
4. **Βήμα = 0** : τότε ο βρόχος είναι **ατέρμων**.

3.3.2. Παραδείγματα

1. ΓΙΑ Κ ΑΠΟ 3 ΜΕΧΡΙ 3
Ο βρόχος εκτελείται ακριβώς μια φορά.
2. ΓΙΑ i ΑΠΟ 10 ΜΕΧΡΙ 5
Ο βρόχος δεν εκτελείται καμία φορά, γιατί $i=10 > 5$ και $\text{Βήμα} > 0$
3. ΓΙΑ i ΑΠΟ 3 ΜΕΧΡΙ 0 ΜΕ_ΒΗΜΑ -1
Ο βρόχος εκτελείται 4 φορές.
4. ΓΙΑ i ΑΠΟ 10 ΜΕΧΡΙ 5 ΜΕ_ΒΗΜΑ -0.5
Ο βρόχος εκτελείται 11 φορές.
5. ΓΙΑ κ ΑΠΟ 5 ΜΕΧΡΙ 10 ΜΕ_ΒΗΜΑ -2
Ο βρόχος δεν εκτελείται καμία φορά, γιατί $\kappa=5 < 10$ και $\text{Βήμα} < 0$.
6. ΓΙΑ κ ΑΠΟ 5 ΜΕΧΡΙ 10 ΜΕ_ΒΗΜΑ 0
Ο βρόχος εκτελείται άπειρες φορές

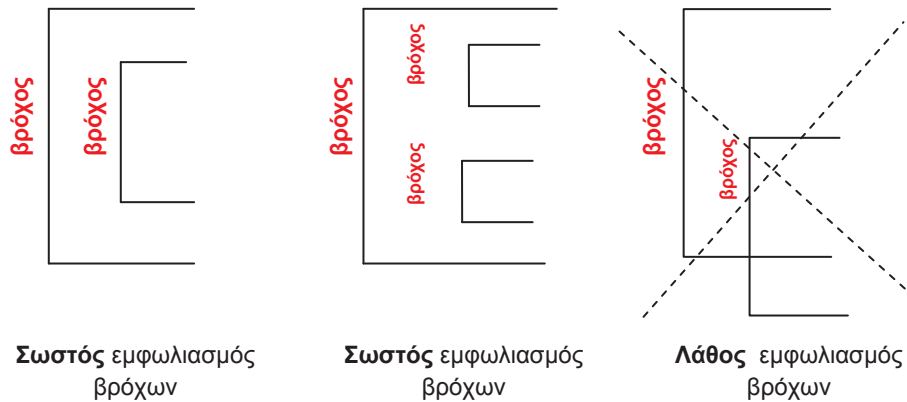
3.3.3. Παρατηρήσεις

1. Μέσα στην εντολή «ΓΙΑ ... ΑΠΟ ... ΜΕΧΡΙ ...» δεν επιτρέπεται να αλλάζετε την τιμή της <μεταβλητής> του ΓΙΑ, γιατί τότε δεν λειτουργεί σωστά η εντολή επανάληψης.
2. Το βήμα δεν είναι απαραίτητο να είναι ακέραιος αριθμός. Μπορεί να πάρει οποιαδήποτε πραγματική τιμή.
3. Μια εντολή επανάληψης «ΓΙΑ ...» μπορεί να μετατραπεί σε «ΟΣΟ ...» και «ΜΕΧΡΙΣ_ΟΤΟΥ ...» ενώ το αντίστροφο δεν ισχύει πάντα.

3.3.4. Εμφωλευμένοι Βρόχοι

Στην περίπτωση αυτή ο ένας βρόχος βρίσκεται μέσα στον άλλο. Για τη σωστή λειτουργία των εμφωλευμένων βρόχων πρέπει να ισχύουν συγκεκριμένοι κανόνες:

- Ο εσωτερικός βρόχος πρέπει να βρίσκεται ολόκληρος μέσα στον εξωτερικό βρόχο. Δηλαδή:



- Η είσοδος σε βρόχο υποχρεωτικά γίνεται από την αρχή του.
- Δεν πρέπει να χρησιμοποιηθεί η ίδια <μεταβλητή> ως μετρητής δύο ή περισσότερων βρόχων που ο ένας βρίσκεται στο εσωτερικό του άλλου.

3.4. Αλγόριθμος ελάχιστου ή μεγίστου από μια ομάδα αριθμών

Όταν έχουμε μια ομάδα αριθμών που θέλουμε να βρούμε τον ελάχιστο ή το μέγιστο αριθμό αυτών, δεν χρησιμοποιούμε τόσες μεταβλητές όσοι είναι οι αριθμοί, αφού μερικές φορές δεν γνωρίζουμε και το πλήθος των αριθμών που θα δώσουμε, αλλά χρησιμοποιούμε μια επαναληπτική εντολή που περιέχει την «ΔΙΑΒΑΣΕ <μεταβλητή>».

Για το λόγο αυτό δεν μπορούμε με πολλά και εμφωλευμένα «AN...ΑΛΛΙΩΣ_ΑΝ...» να βρούμε τον ελάχιστο ή τον μικρότερο και έτσι χρησιμοποιούμε τον παρακάτω αλγόριθμο:

1. Σε μια μεταβλητή Max ή Min, που εκφράζουν αντίστοιχα την μεγαλύτερη ή μικρότερη τιμή που θα έχουν στο τέλος, δίνουμε **Αρχική Τιμή**.
2. Κάθε νέα τιμή που δίνουμε στη <μεταβλητή> την ελέγχουμε με το Max ή Min αντίστοιχα, <μεταβλητή> < Min ή <μεταβλητή> > Max, και αν ισχύουν οι παραπάνω συνθήκες καταχωρούμε στο Min ή στο Max τη μεταβλητή με τη νέα τιμή. Όπου **Αρχική Τιμή** υπάρχουν δύο περιπτώσεις.

1η περίπτωση

Αυθαίρετες τιμές

Στο Max δίνουμε την μικρότερη δυνατή τιμή.
Στο Min δίνουμε την μεγαλύτερη δυνατή Τιμή.
Αυτό τον τρόπο τον χρησιμοποιούμε στις περιπτώσεις που το πεδίο τιμών της μεταβλητής είναι συγκεκριμένο. Π.χ δίνουμε τους βαθμούς μαθητών [0,20]. Τότε σαν αρχικές τιμές Μπορούμε να δώσουμε

Max ← 0
και Min ← 20

2η περίπτωση

Συγκεκριμένες τιμές

Στο Max ή στο Min δίνουμε την πρώτη τιμή από την ομάδα που θα δώσω.
Αυτό τον τρόπο τον χρησιμοποιούμε στις περιπτώσεις που το πεδίο τιμών της μεταβλητής δεν είναι συγκεκριμένο.
Π.χ. Πληκτρολογούμε αριθμούς να βρούμε τον μεγαλύτερο ή τον μικρότερο απ' αυτούς.

ΔΙΑΒΑΣΕ B
Max ← B
Min ← B

3.4.1. Παραδείγματα

1. Να γίνει πρόγραμμα που να δίνονται 1000 αριθμοί από το πληκτρολόγιο να βρεθεί και να τυπωθεί ο μεγαλύτερος απ' αυτούς.

Λύση

Ανάλυση:

1. Χρησιμοποιούμε την επαναληπτική εντολή ΓΙΑ, για να πληκτρολογήσουμε τους 1000 αριθμούς.
2. Για να βρούμε τον μέγιστο χρησιμοποιούμε τον αλγόριθμο του ΜΕΓΙΣΤΟΥ με αρχική τιμή τον πρώτο αριθμό που πληκτρολογούμε.

Γενική μορφή:

```
    Διαβάζω αριθμό
    Καταχωρώ αριθμό στην
    Μαχ
    βρόχος
     $i = 2$ 
    Διαβάζω αριθμό
    Ελέγχω αριθμό με την Μαχ
    Μέχρι  $i = 1000$ 
    Τυπώνω τον Μαχ
```

ΠΡΟΓΡΑΜΜΑ Μέγιστος

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A, I, Μαχ

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε αριθμό'

ΔΙΑΒΑΣΕ A

Αρχική τιμή Μαχ \leftarrow A

ΓΙΑ i **ΑΠΟ** 2 **ΜΕΧΡΙ** 1000

ΓΡΑΨΕ 'Δώσε αριθμό'

ΔΙΑΒΑΣΕ A

ελεγχος
ΑΝ A > Μαχ **ΤΟΤΕ**
 Μαχ \leftarrow A
ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Μεγαλύτερος ο:', Μαχ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μέγιστος

Παρατήρηση

Στο παραπάνω παράδειγμα παρατηρούμε ότι όταν στο Μαχ ή αντίστοιχα Min δίνουμε τον πρώτο αριθμό από την λίστα που θα πληκτρολογήσουμε τότε το πρώτο ΔΙΑΒΑΣΕ δίνεται έξω από την επανάληψη, για να δώσω την τιμή μου σαν αρχική τιμή στο Μαχ ή στο Min.

2. Να βρεθεί ο μεγαλύτερος και ο μικρότερος από 1000 αριθμός που πληκτρολογώ.

Λύση

Ανάλυση:

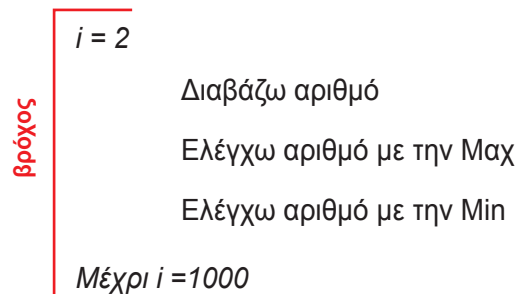
1. Χρησιμοποιούμε την επαναληπτική εντολή «ΓΙΑ...» για να πληκτρολογήσουμε τους 1000 αριθμούς.
2. Εφαρμόζουμε μέσα στο «ΓΙΑ...» τον έλεγχο για τον μεγαλύτερο και τον έλεγχο για τον μικρότερο σύμφωνα με τον αλγόριθμο του μεγαλύτερου και του μικρότερου.

Γενική μορφή:

Διαβάζω αριθμό

Καταχωρώ τον αριθμό στην Max

Καταχωρώ τον αριθμό στην Min



Τυπώνω την Max και την Min

ΠΡΟΓΡΑΜΜΑ Max_Min

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A, i, Max, Min

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε αριθμό'

ΔΙΑΒΑΣΕ A

Αρχικές
Τιμές

Max ← A

Min ← A

ΓΙΑ i **ΑΠΟ** 2 **ΜΕΧΡΙ** 1000

ΓΡΑΨΕ 'Δώσε αριθμό'

ΔΙΑΒΑΣΕ A

έλεγχος
Max

ΑΝ A > Max **ΤΟΤΕ**

Max ← A

ΤΕΛΟΣ-ΑΝ

έλεγχος
Min

ΑΝ A < Min **ΤΟΤΕ**

Min ← A

ΤΕΛΟΣ_ΑΝ και M

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Μεγαλύτερος:', Max

ΓΡΑΨΕ 'Μικρότερος:', Min

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Max_Min

Παρατήρηση

Όταν έχουμε ταυτόχρονα να ελέγχουμε Max και Min σ' ένα πρόγραμμα δεν πρέπει να χρησιμοποιούμε την «ΑΝ ... ΑΛΛΙΩΣ ...» γιατί δεν θα έχουμε σωστά αποτελέσματα, επειδή αν ένας αριθμός δεν είναι μεγαλύτερος από την Max, αυτό δεν σημαίνει ότι είναι μικρότερος από την Min.

Δηλαδή, θα γράφουμε δύο διαφορετικά «ΑΝ...», ένα για το Max και ένα για το Min.

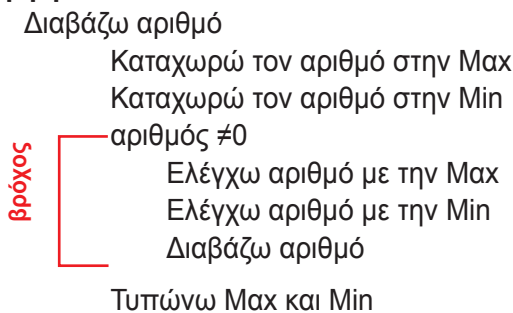
3. Διαβάζουμε αριθμούς από το πληκτρολόγιο, το τέλος της πληκτρολόγησης δηλώνεται αν δώσουμε τον αριθμό 0. Να βρίσκεται και να τυπώνεται ο μεγαλύτερος και ο μικρότερος απ' αυτούς.

Λύση

Ανάλυση:

1. Χρησιμοποιούμε την επαναληπτική εντολή «ΟΣΟ...» για να πληκτρολογήσουμε τους αριθμούς, γιατί εδώ η πληκτρολόγηση σταματά όταν δώσουμε την **τιμή φρουρό '0'**. Δηλαδή, έχουμε επανάληψη υπό συνθήκη.
2. Μέσα στην επανάληψη κάνουμε έλεγχο για τον μεγαλύτερο και έλεγχο για τον μικρότερο.

Γενική μορφή:



ΠΡΟΓΡΑΜΜΑ Max_Min

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A, Max, Min

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε αριθμό'

ΔΙΑΒΑΣΕ A

Max ← A

Min ← A

ΟΣΟ A <> 0 **ΕΠΑΝΑΛΑΒΕ**

ΑΝ A > Max **ΤΟΤΕ**

Max ← A

ΤΕΛΟΣ_ΑΝ

ΑΝ A < Min **ΤΟΤΕ**

Min ← A

ΤΕΛΟΣ_ΑΝ

ΓΡΑΨΕ 'Δώσε αριθμό'

ΔΙΑΒΑΣΕ A

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Μεγαλύτερος ',Max

ΓΡΑΨΕ 'Μικρότερος ',Min

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Max_Min

4. Να διαβάζονται τα ονόματα και ο αντίστοιχος ΜΟ 70 μαθητών της Β' Λυκείου ενός σχολείου. Να υπολογισθεί και να τυπωθεί το όνομα και ο μέσος όρος του μαθητή με το μεγαλύτερο ΜΟ και του μαθητή με το μικρότερο ΜΟ.

Λύση

Ανάλυση:

1. Χρησιμοποιούμε την επαναληπτική εντολή «ΓΙΑ...» για να διαβάσουμε τα 70 ονόματα και μέσους όρους των μαθητών.
2. Θα χρησιμοποιηθεί ο αλγόριθμος του μεγαλύτερου και του μικρότερου, με την διαφορά ότι θα χρησιμοποιηθούν και βοηθητικές μεταβλητές που θα κρατούν (αποθηκεύουν) αντίστοιχα το όνομα του μαθητή με το μεγαλύτερο ΜΟ και του μαθητή με το μικρότερο ΜΟ.
3. Σαν αρχική τιμή του μεγαλύτερου και του μικρότερου μπορούμε να δώσουμε το 20 και το 0 αντίστοιχα, εφόσον το πεδίο τιμών είναι [0,20].

Γενική μορφή:

Max \leftarrow 0

Min \leftarrow 20

$i = 1$

Διαβάζω Όνομα και Μέσο Όρο μαθητή

Ελέγχω τον Μέσο Όρο με την Max

Ελέγχω τον Μέσο Όρο με την Min

Μέχρι $i = 70$

Τυπώνω την Max και το Όνομα του μεγαλύτερου βαθμού

Τυπώνω την Min και το Όνομα του μικρότερου βαθμού

ΠΡΟΓΡΑΜΜΑ Ο_μεγαλύτερος_και_ο_μικρότερος_βαθμός

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: i

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ, Max, Min

ΧΑΡΑΚΤΗΡΕΣ: Ον, Μαχ_ον, Μιν_ον

ΑΡΧΗ

Max \leftarrow 0

Min \leftarrow 20

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 70

ΓΡΑΨΕ 'Δώσε Όνομα και Μ.Ο. μαθητή'

ΔΙΑΒΑΣΕ Ον, ΜΟ

ΑΝ ΜΟ > Max **ΤΟΤΕ**

Max \leftarrow ΜΟ

Μαχ_ον \leftarrow Ον

ΤΕΛΟΣ_ΑΝ

ΑΝ ΜΟ < Min **ΤΟΤΕ**

Min \leftarrow ΜΟ

Μιν_ον \leftarrow Ον

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Καλύτερος ο ', Μαχ_ον, 'με Μ.Ο.=', Max

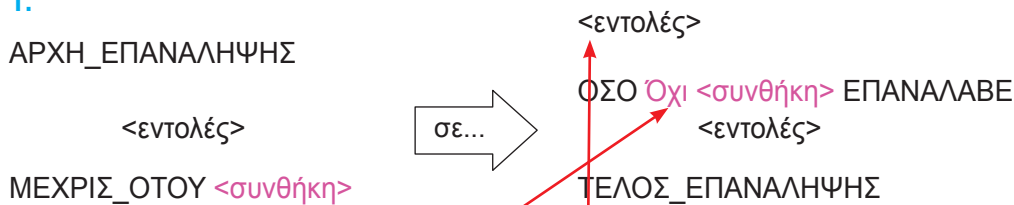
ΓΡΑΨΕ 'Χειρότερος ο ', Μιν_ον, 'με Μ.Ο.=', Min

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ο_μεγαλύτερος_και_ο_μικρότερος_βαθμός

3.5 Μετατροπές από μια δομή επανάληψης σε άλλη

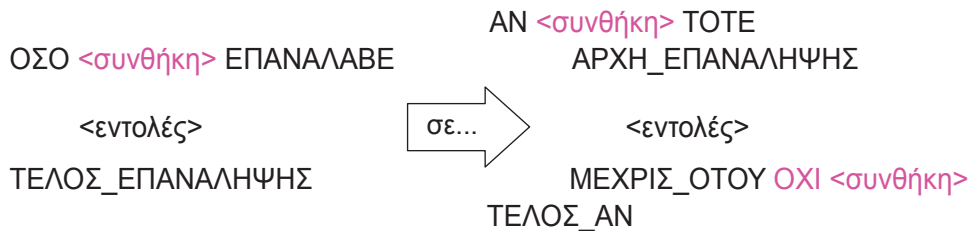
3.5.1. Μετατροπή από ΑΡΧΗ ΕΠΑΝΑΛΗΨΗΣ ... ΣΕ ΟΣΟ...ΚΑΙ ΑΝΤΙΣΤΡΟΦΩΣ

1.



- Επειδή η εντολή επανάληψης «ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ» εκτελείται όσο η <συνθήκη> είναι Ψευδής, ενώ στην εντολή επανάληψης «ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ» εκτελείται όσο η <συνθήκη> της είναι Αληθής, γι' αυτό κατά την μετατροπή από την μια εντολή επανάληψης στην άλλη αρκεί να γράφουμε την άρνηση της <συνθήκη> από την μια στην άλλη ή να γράφουμε τον τελεστή ΟΧΙ την συνθήκη.
- Επίσης επειδή η ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ" εκτελεί τουλάχιστον μια φορά όλες τις εντολές της, γι' αυτό όταν την μετατρέπουμε στην «ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ», πρέπει όλες τις εντολές της να τις γράψουμε μια φορά, πριν την «ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ» και άλλη μια φορά μέσα σ' αυτήν.

2.



Επειδή η εντολή επανάληψης «ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ» εκτελείται όσο η <συνθήκη> είναι Αληθής, ενώ στην εντολή επανάληψης «ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ» εκτελείται όσο η συνθήκη είναι Ψευδής, γι' αυτό κατά την μετατροπή από την μια εντολή στην άλλη αρκεί να γράφουμε την άρνηση της <συνθήκης> ή να γράφουμε τον τελεστή ΟΧΙ την συνθήκη. .

- Επίσης επειδή η «ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ» μπορεί να μην εκτελεστεί καμία φορά σε αντίθεση με την «ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ» που τουλάχιστον μια φορά θα εκτελεστεί, γι' αυτό η εντολή «ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ» θα «μπει» μέσα σε μια εντολή «ΑΝ...» με την ίδια συνθήκη, που μόνο αν ισχύει θα εκτελεστεί η «ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ».

3.5.1.1. Παραδείγματα

1. Μετατροπή από ΟΣΟ.. σε ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

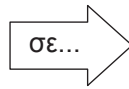
ΔΙΑΒΑΣΕ x
ΟΣΟ $x < > 0$ ΕΠΑΝΑΛΑΒΕ
 $y \leftarrow x^2$
ΓΡΑΨΕ y
ΔΙΑΒΑΣΕ x
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ



ΔΙΑΒΑΣΕ x
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ $x < > 0$ ΤΟΤΕ
 $y \leftarrow x^2$
ΓΡΑΨΕ y
ΔΙΑΒΑΣΕ x
ΤΕΛΟΣ_ΑΝ
ΜΕΧΡΙΣ_ΟΤΟΥ $x = 0$

2. Μετατροπή από ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ σε ΟΣΟ...

```
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
  ΔΙΑΒΑΣΕ x
  y <-x ^ 2
  ΓΡΑΨΕ y
ΜΕΧΡΙΣ_ΟΤΟΥ x=0
```

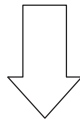


```
ΔΙΑΒΑΣΕ x
y <-x ^ 2
ΓΡΑΨΕ y
ΟΣΟ x<>0 ΕΠΑΝΑΛΑΒΕ
  ΔΙΑΒΑΣΕ x
  y <-x ^ 2
  ΓΡΑΨΕ y
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

3.5.2. Μετατροπή από ΓΙΑ... σε ΟΣΟ... ή σε ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ και αντιστρόφως

1. Περίπτωση $\text{τιμή1} \leq \text{τιμή2}$ και $\beta > 0$

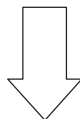
```
ΓΙΑ <μεταβλητή> ΑΠΟ τιμή1 ΜΕΧΡΙ τιμή 2 ΜΕ_ΒΗΜΑ β
  <εντολές>
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
<μεταβλητή> ← τιμή1
ΟΣΟ <μεταβλητή> <=τιμή2 ΕΠΑΝΑΛΑΒΕ
  <εντολές>
  <μεταβλητή> ← <μεταβλητή>+β
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

2. Περίπτωση $\text{τιμή1} \geq \text{τιμή2}$ και $\beta < 0$

```
ΓΙΑ <μεταβλητή> ΑΠΟ τιμή1 ΜΕΧΡΙ τιμή2 ΜΕ_ΒΗΜΑ β
  <εντολές>
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
<μεταβλητή> ← τιμή1
ΟΣΟ <μεταβλητή> >=τιμή2 ΕΠΑΝΑΛΑΒΕ
  <εντολές>
  <μεταβλητή> ← <μεταβλητή>+β
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

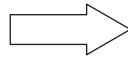
- Πριν την εντολή ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ δίνουμε στην <μεταβλητή> της ΓΙΑ... την αρχική τιμή δηλ. την **τιμή1**.
- Στη συνθήκη της ΟΣΟ...ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ συγκρίνουμε την <μεταβλητή> με την **τιμή2**. Η σύγκριση εξαρτάται από το βήμα αν είναι θετικό ή αρνητικό όπως βλέπουμε παραπάνω.
- Πριν το ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ της ΟΣΟ... αυξάνουμε την τιμή της μεταβλητής όσο είναι η τιμή του βήματος. Στη περίπτωση που το βήμα δεν υπάρχει, τότε η <μεταβλητή> αυξάνεται κατά 1.
- Ανάλογα, γίνεται η μετατροπή της ΟΣΟ... σε ΓΙΑ...

3.5.2.1. Παραδείγματα

1. Μετατροπή από ΓΙΑ... σε ΟΣΟ...

α)

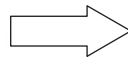
```
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10  
  ΓΡΑΨΕ i ^ 3  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
i ← 1  
ΟΣΟ i <= 100 ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ i ^ 3  
  i ← i+1  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

β)

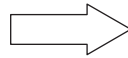
```
ΓΙΑ i ΑΠΟ 10 ΜΕΧΡΙ 100 ΜΕ_ΒΗΜΑ 2  
  ΓΡΑΨΕ i ^ 2  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
i ← 10  
ΟΣΟ i <= 100 ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ i ^ 2  
  i ← i + 2  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

γ)

```
ΓΙΑ i ΑΠΟ 1000 ΜΕΧΡΙ 100 ΜΕ_ΒΗΜΑ -1  
  ΓΡΑΨΕ i  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

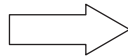


```
i ← 1000  
ΟΣΟ i >= 100 ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ i  
  i ← i - 1  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

2. Μετατροπή από ΟΣΟ... σε ΓΙΑ...

α)

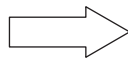
```
i ← 10  
ΟΣΟ i <= 100  
ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ i  
  i ← i+1  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
ΓΙΑ i ΑΠΟ 10 ΜΕΧΡΙ 100_ΒΗΜΑ 100  
  ΓΡΑΨΕ i  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

β)

```
i ← 10  
ΟΣΟ i <= 100 ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ i  
  i ← i + 2  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
ΓΙΑ i ΑΠΟ 10 ΜΕΧΡΙ 100 ΜΕ_ΒΗΜΑ 2  
  ΓΡΑΨΕ i  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

γ)

```
i ← 100  
ΟΣΟ i >= 1 ΕΠΑΝΑΛΑΒΕ  
  ΓΡΑΨΕ i  
  i ← i - 2  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```



```
ΓΙΑ i ΑΠΟ 100 ΜΕΧΡΙ 1 ΜΕ_ΒΗΜΑ -2  
  ΓΡΑΨΕ i  
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

3.6. Μη λυμένες ασκήσεις

1. Πόσες φορές θα εκτελεστούν οι παρακάτω βρόχοι επανάληψης;

- 1) ΓΙΑ i ΑΠΟ -3 ΜΕΧΡΙ 1
- 2) ΓΙΑ i ΑΠΟ 4 ΜΕΧΡΙ 2
- 3) ΓΙΑ i ΑΠΟ 5 ΜΕΧΡΙ 5
- 4) ΓΙΑ i ΑΠΟ 6 ΜΕΧΡΙ 6 ΜΕ_ΒΗΜΑ 0
- 5) ΓΙΑ i ΑΠΟ -9 ΜΕΧΡΙ -15 ΜΕ_ΒΗΜΑ -1
- 6) ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10 ΜΕ_ΒΗΜΑ 2
- 7) ΓΙΑ i ΑΠΟ -4 ΜΕΧΡΙ -1 ΜΕ_ΒΗΜΑ -2
- 8) ΓΙΑ i ΑΠΟ 5 ΜΕΧΡΙ 12 ΜΕ_ΒΗΜΑ 3
- 9) ΓΙΑ i ΑΠΟ 0.5 ΜΕΧΡΙ 2 ΜΕ_ΒΗΜΑ 0.5
- 10) ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 2 ΜΕ_ΒΗΜΑ 0

2. Πόσες φορές θα εκτελεστούν οι παρακάτω βρόχοι;

- | | |
|--|---|
| 1) $\alpha \leftarrow 0$
ΟΣΟ $\alpha < 15$ ΕΠΑΝΑΛΑΒΕ
$\chi \leftarrow \alpha + 1$
$\alpha \leftarrow \alpha + 3$
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ | 2) $\alpha \leftarrow -12$
ΟΣΟ $\alpha > 0$ ΕΠΑΝΑΛΑΒΕ
ΓΡΑΨΕ Α
$\alpha \leftarrow \alpha + 3$
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ |
| 3) $\alpha \leftarrow 10$
ΟΣΟ $A \geq 20$ ΕΠΑΝΑΛΑΒΕ
ΓΡΑΨΕ α
$\alpha \leftarrow \alpha + 5$
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ | 4) $\alpha \leftarrow 2$
ΟΣΟ $\alpha \leq 11$ ΕΠΑΝΑΛΑΒΕ
ΓΡΑΨΕ α
$\alpha \leftarrow \alpha + 5$
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ |

3. Πόσες φορές θα εκτελεστούν οι παρακάτω βρόχοι;

- | | |
|---|---|
| 1) $\beta \leftarrow 0$
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ β
$\beta \leftarrow \beta + 3$
ΜΕΧΡΙΣ_ΟΤΟΥ $\beta > 13$ | 2) $\alpha \leftarrow -9$
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ α
$\alpha \leftarrow \alpha - 3$
ΜΕΧΡΙΣ_ΟΤΟΥ $\alpha > -1$ |
| 3) $\beta \leftarrow 200$
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ β
$\beta \leftarrow \beta - 50$
ΜΕΧΡΙΣ_ΟΤΟΥ $\beta \leq 0$ | 4) $\beta \leftarrow 0$
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ β
$\beta \leftarrow \beta + 3$
ΜΕΧΡΙΣ_ΟΤΟΥ $\beta > 7$ |

4. Δίνεται το παρακάτω τμήμα προγράμματος σε ΓΛΩΣΣΑ:

```
sum ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 6 ΜΕ_ΒΗΜΑ 2
    sum ← sum+i
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

i) Να μετατραπεί σε ισοδύναμο με χρήση της δομής **ΟΣΟ...ΕΠΑΝΑΛΑΒΕ**

ii) Να μετατραπεί σε ισοδύναμο με χρήση της δομής **ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ... ΜΕΧΡΙΣ_ΟΤΟΥ**

5. Δίνεται το παρακάτω τμήμα προγράμματος σε ΓΛΩΣΣΑ:

```
sum ← 0
i ← 0
```

ΔΙΑΒΑΣΕ α
ΟΣΟ $i \leq 5$ **ΕΠΑΝΑΛΑΒΕ**
 sum ← sum+α
 i ← i+1
ΔΙΑΒΑΣΕ α
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

i) Να μετατραπεί σε ισοδύναμο με χρήση της δομής

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ...ΜΕΧΡΙΣ_ΟΤΟΥ

ii) Να μετατραπεί σε ισοδύναμο με χρήση της δομής **ΓΙΑ...**

6. Δίνεται το παρακάτω τμήμα προγράμματος σε ΓΛΩΣΣΑ:

sum ← 10
ΔΙΑΒΑΣΕ x
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
 sum ← sum+x
ΔΙΑΒΑΣΕ x
ΜΕΧΡΙΣ_ΟΤΟΥ $x \leq 0$

Να μετατραπεί σε ισοδύναμο με χρήση της δομής

ΟΣΟ... ΕΠΑΝΑΛΑΒΕ και της δομής **ΓΙΑ...**

7. Να γραφτεί ο πίνακας τιμών και η οθόνη που εμφανίζεται κατά την εκτέλεση του παρακάτω προγράμματος σε ΓΛΩΣΣΑ:

ΠΡΟΓΡΑΜΜΑ Ασκηση_Επανάληψης
 α ← 10
 β ← 5

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 20 **ΜΕ_ΒΗΜΑ** 5
ΓΙΑ j **ΑΠΟ** i **ΜΕΧΡΙ** 15 **ΜΕ_ΒΗΜΑ** 2

α ← α div 2
 β ← β mod 3
ΑΝ α < > 0 **ΤΟΤΕ**
ΓΡΑΨΕ α

ΤΕΛΟΣ_ΑΝ
ΑΝ β < > 0 **ΤΟΤΕ**
ΓΡΑΨΕ β
ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ασκηση_Επανάληψης

7. Να γραφεί το πρόγραμμα σε "ΓΛΩΣΣΑ" το οποίο υπολογίζει το παρακάτω αθροίσματα:

1) $1 + 3 + 5 + \dots + 99$

2) $\frac{1}{2} + \frac{1}{4} + \frac{1}{6} + \dots + \frac{1}{100}$

3) $2^2 + 4^2 + \dots + 100^2$

8. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να διαβάζει θετικό ακέραιο αριθμό N. Στη συνέχεια να διαβάζει θετικούς αριθμούς μέχρι το άθροισμά τους να γίνει μεγαλύτερο από ένα αριθμό N. Τέλος να υπολογίζει και εμφανίζει τον μεγαλύτερο αριθμό που πληκτρολογήθηκε.

Να γίνεται έλεγχος των αριθμών που πληκτρολογούνται ώστε να είναι θετικοί, σε αντίθετη περίπτωση να βγαίνει μήνυμα λάθους και να ξαναπληκτρολογούμε ένα θετικό αριθμό.

9. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να διαβάζει τα ονόματα των 26 μαθητών μιας τάξης και 10 βαθμούς για τον καθένα. Το πρόγραμμα να υπολογίζει και να εμφανίζει το Μέσο Όρο για κάθε μαθητή. Στο τέλος να τυπώνεται ο μαθητής με το μεγαλύτερο μέσο όρο, ο μαθητής με τον μικρότερο μέσο όρο και ο μέσος όρος τους, αντίστοιχα.
10. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ" , το οποίο να διαβάζει το πλήθος των μαθητών ενός σχολείου. Στη συνέχεια διαβάζει το μέσο όρο κάθε μαθητή. Να υπολογίζει και να τυπώνει το πλήθος και το αντίστοιχο ποσοστό των μαθητών που έχουν μέσο όρο <10 , των μαθητών με μέσο όρο στο διάστημα $[10,18]$ και των μαθητών με μέσο όρο >18 .
Ο βαθμός που πληκτρολογείται είναι από 1 μέχρι 20. το πρόγραμμα να κάνει έλεγχο εγκυρότητας του βαθμού και στη περίπτωση που δίνεται βαθμός έξω από τα επιτρεπτά όρια να τυπώνεται τονύνημα "Μη αποδεκτή βαθμολογία. Ξαναπροσπαθήστε....".
11. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ" , το να οποίο διαβάζει πραγματικούς αριθμούς μέχρι να πληκτρολογήσουμε το 100 ή όταν το άθροισμα γίνει ίσο με 1.000.000. Ο αλγόριθμος να υπολογίζει και να εμφανίζει το άθροισμα και το πλήθος των αριθμών που πληκτρολογήθηκαν. Ο αριθμός 100 που σηματοδοτεί και το τέλος της πληκτρολόγησης να λαμβάνεται υπόψη στο άθροισμα και στο πλήθος.
12. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ" , το οποίο διαβάζει ένα ακέραιο αριθμό.
Να υπολογίζει και να εμφανίζει:
I. το άθροισμα των ψηφίων του και
II. το πλήθος των ψηφίων του
13. Η σχολή Θετικών Επιστημών ενός Πανεπιστημίου έχει 4 τμήματα. Κάθε τμήμα δέχεται κάθε χρονιά 150 φοιτητές. Η πρυτανεία αποφάσισε για το τρέχον έτος να κάνει μια στατιστική μελέτη για κάθε τμήμα. Έτσι, αποφάσισε να υπολογίσει το ποσοστό των αριστούχων, δηλ. των νεοεισαχθέντων φοιτητών με βαθμό μεγαλύτερο ή ίσο του 19, ανά σχολή. Να γραφεί πρόγραμμα το οποίο να διαβάζει τους βαθμούς εισαγωγής των φοιτητών και να υπολογίζει και να εμφανίζει :
a) Το ποσοστό των αριστούχων κάθε τμήματος
b) Σε ποιο τμήμα εισήχθηκε ο φοιτητής με το μεγαλύτερο βαθμό.
14. Σ' ένα ασανσέρ επιτρέπεται να επιβιβασθούν 8 άτομα και το μέγιστο επιτρεπτό ωφέλιμο βάρος στο ασανσέρ είναι 900 κιλά. Το ασανσέρ ξεκινά όταν γεμίσει (όταν το σύνολο των ατόμων δεν ξεπερνά τα 8 άτομα ή αν το συνολικό βάρος δεν υπερβαίνει τα 900 κιλά) ή όταν δεν υπάρχει άλλο άτομο για να επιβιβασθεί στον όροφο που έχει σταματήσει. Το ασανσέρ κάνει συνολικά 8 στάσεις . Σε κάθε στάση, εκτός της τελευταίας, το πρόγραμμα εμφανίζει το μήνυμα "Υπάρχει άτομο να εισέλθει; (ΝΑΙ ή ΟΧΙ)". Αν η απάντηση είναι "ΝΑΙ" εισάγεται το βάρος του ατόμου και έτσι σηματοδοτείται η επιβίβασή του στο ασανσέρ. Η παραπάνω διαδικασία επαναλαμβάνεται μέχρι να δοθεί η απάντηση "ΟΧΙ" και να σταματήσει η επιβίβαση.
Επίσης σε κάθε στάση, εκτός από την πρώτη, το πρόγραμμα εμφανίζει το μήνυμα "Υπάρχει άτομο να εξέλθει; (ΝΑΙ ή ΟΧΙ)", αν η απάντηση είναι "ΝΑΙ" σηματοδοτείται η επιβίβαση στο ασανσέρ. Η παραπάνω διαδικασία επαναλαμβάνεται μέχρι να δοθεί η απάντηση "ΟΧΙ".
Να γραφεί πρόγραμμα που να περιγράφει την παραπάνω λειτουργία του ασανσέρ και στο τέλος να τυπώνει το σύνολο των ατόμων που εισείλθαν στο ασανσέρ στη 1η στάση και το σύνολο των ατόμων που αποβίβαστηκαν στην 8η στάση.

ΠΙΝΑΚΕΣ

Περιεχόμενα

- Μονοδιάστατος – Δισδιάστατος Πίνακας
- Τυπικές επεξεργασίες στοιχείων Πίνακα
- Παράλληλοι Πίνακες

4. ΠΙΝΑΚΕΣ

Στο πρόγραμμα ο πίνακας εκφράζει μια μεταβλητή που έχει περισσότερες από μια θέσεις, δηλαδή μια μεταβλητή πολλαπλών θέσεων. Για το λόγο αυτό όταν ορίζουμε ένα πίνακα, χρησιμοποιούμε το όνομα της μεταβλητής και το δείκτη, που είναι ένας αριθμός και εκφράζει τη θέση του κάθε δεδομένου.

Ένας πίνακας μπορεί να είναι μονοδιάστατος, αλλά στη γενικότερη περίπτωση μπορεί να είναι δισδιάστατος, τρισδιάστατος και γενικά ν-διάστατος πίνακας. Όσο αφορά τους δισδιάστατους πίνακες αν το μέγεθος των δύο διαστάσεων είναι ίσο, δηλαδή είναι $n \times n$, τότε ο πίνακας λέγεται **τετραγωνικός**.

Αρα:

Πίνακας είναι ένα σύνολο δεδομένων ίδιου τύπου, τα οποία έχουν το ίδιο όνομα. Κάθε ένα από τα δεδομένα του πίνακα λέγεται στοιχείο του πίνακα. Όταν αναφερόμαστε σε ένα δεδομένο από τον πίνακα τότε γράφουμε το όνομα του πίνακα ακολουθούμενο από έναν δείκτη.

Οι στατικές δομές υλοποιούνται με πίνακες.

Πίνακας είναι μια δομή που περιέχει στοιχεία του ίδιου τύπου.

4.1. Πλεονεκτήματα χρήσης πινάκων:

Η χρήση πίνακα είναι ένας βολικός τρόπος για τη διαχείριση πολλών δεδομένων ίδιου τύπου.

4.2. Μειονεκτήματα χρήσης πινάκων:

- Οι πίνακες απαιτούν μνήμη, η οποία δεσμεύεται από την αρχή του προγράμματος. Αυτό έχει
- σαν αποτέλεσμα, σε μεγάλα και σύνθετα προγράμματα την αδυναμία εκτέλεσης τους.
- Οι πίνακες περιορίζουν τις δυνατότητες του προγράμματος.

4.3. Που χρησιμοποιούνται οι πίνακες;

Χρησιμοποιούνται στη περίπτωση που έχουμε να διαχειριστούμε πολλά δεδομένα και πρέπει να είναι αποθηκευμένα για μεταγενέστερη επεξεργασία τους.

4.4. Δήλωση πίνακα στη ΓΛΩΣΣΑ.

<τύπος>: Όνομα_πίνακα [πλήθος θέσεων].

Μονοδιάστατος

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[20]

ΧΑΡΑΚΤΗΡΕΣ: B[20]

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

Δείκτες ή Θέσεις

1 2 3 4 5 6 7 8 ... 20

A=

7	-7	9	21	12	75	-2	5	...	214
---	----	---	----	----	----	----	---	-----	-----

Όνομα Πίνακα

Τιμή ή
Δεδομένο

Δείκτες ή Θέσεις

1 2 3 4 ... 20

B=

Δήμητρα	Άννα	Νίκος	Σοφία	...	Αλεξία
---------	------	-------	-------	-----	--------

Όνομα Πίνακα

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

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

Δισδιάστατος

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A1[20,2]

ΧΑΡΑΚΤΗΡΕΣ: B1[5,3]

Ο πίνακας A1[20,2] δηλώνει μια μεταβλητή A1 με $20 \times 2 = 40$ θέσεις, έτσι ώστε να έχει 20 γραμμές και 2 στήλες. Δηλ. αν τον αναπαριστούσα θα ήταν :

Θέσεις
γραμμή

Θέσεις
στήλης

A1=

	1	2
1	7	12
2	34	-6
...
20	107	34

Τιμή ή
Δεδομένο

- Για να ορίσουμε μια θέση σ' ένα δισδιάστατο πίνακα πρέπει να ορίσουμε σε ποια γραμμή και μετά σε ποια στήλη βρίσκεται το δεδομένο.

Όταν γράφουμε το A1[15,2] στο πρόγραμμα, αναφερόμαστε στο δεδομένο που βρίσκεται στην 15η γραμμή και τη 2η στήλη.

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

- Όταν ορίζουμε τη θέση ενός δισδιάστατου πίνακα, πάντα γράφουμε πρώτα την γραμμή και μετά τη στήλη του πίνακα.

Π.χ. A[10,2]

γραμμή στήλη

4.5 Είσοδος δεδομένων σε πίνακα

4.5.1. Σε μονοδιάστατο πίνακα

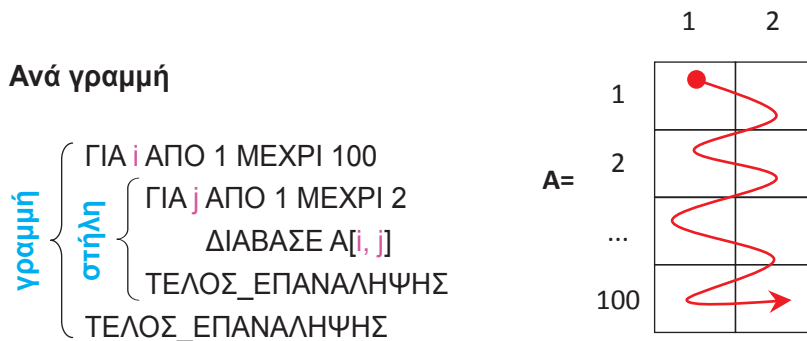
Έστω ότι θέλω να διαβάσω 100 ακέραιους αριθμούς από το πληκτρολόγιο:

1ος τρόπος (χωρίς χρήση πίνακα)	2ος τρόπος (με χρήση πίνακα)
<p data-bbox="255 426 455 644">ΔΙΑΒΑΣΕ A1 ΔΙΑΒΑΣΕ A2 ΔΙΑΒΑΣΕ A3 ΔΙΑΒΑΣΕ A4 ... ΔΙΑΒΑΣΕ A100</p> <p data-bbox="127 685 717 758">Αυτός ο τρόπος δεν χρησιμοποιείται γιατί γράφουμε πολλές φορές την ίδια εντολή.</p> <p data-bbox="271 762 574 903">ή ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 20 ΔΙΑΒΑΣΕ A ΤΕΛΟΣ_ΕΠΝΑΛΗΨΗΣ</p> <p data-bbox="127 944 717 1203">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.</p>	<p data-bbox="751 426 1167 644">ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ: A[100] ΑΡΧΗ ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100 ΔΙΑΒΑΣΕ A[i] ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ ... ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ</p> <p data-bbox="751 944 1372 1089">Κατά την εκτέλεση των παραπάνω εντολών πληκτρολογούμε 100 αριθμούς έτσι ώστε ο κάθε ένας να μπαίνει σε διαφορετική θέση στο πίνακα.</p> <p data-bbox="751 1094 1372 1239">Αυτό έχει σαν αποτέλεσμα να μπορούμε στο πρόγραμμά μας παρακάτω, να τους επεξεργαστούμε όλους ή κάθε έναν απ' αυτούς, όποτε τους χρειαζόμαστε.</p>

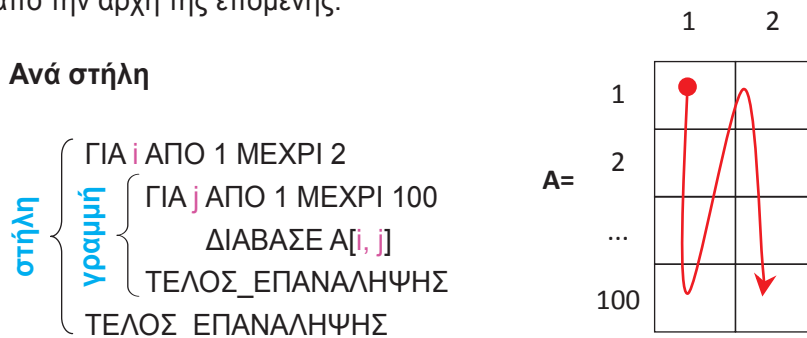
4.5.2. Σε δισδιάστατο πίνακα

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 $A[10,2]$



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.

Παρατήρηση

Η ανάγνωση, η επεξεργασία και η εκτύπωση των στοιχείων ενός πίνακα γίνεται πάντα από βρόχους, που επαναλαμβάνονται προκαθορισμένο αριθμό φορών όσα είναι τα στοιχεία του πίνακα. Επομένως συμφέρει να υλοποιηθούν με την εντολή επανάληψης «ΓΙΑ...».

4.5.3. Παραδείγματα

1. Να γραφτεί πρόγραμμα που να υπολογίζει και να τυπώνει τη μέση θερμοκρασία 30 ημερών και στη συνέχεια να υπολογίζει και να τυπώνει πόσες θερμοκρασίες ήταν μικρότερες από τη μέση θερμοκρασία.

Λύση

Ανάλυση

Αν χρησιμοποιήσουμε μια «ΓΙΑ...» για να διαβάσουμε τις 30 θερμοκρασίες, τότε θα βρούμε τον μέσο όρο τους αλλά για να ελέγξουμε πόσες είναι μικρότερες από τη μέση θερμοκρασία θα πρέπει να τις ξαναδιαβάσουμε. Αυτό δεν εξυπηρετεί γιατί θα πρέπει να πληκτρολογήσουμε 2 φορές τις 30 θερμοκρασίες. Έτσι θα χρησιμοποιηθεί μονοδιάστατος πίνακας 30 θέσεων στον οποίο θα καταχωρήσουμε τις 30 θερμοκρασίες και ταυτόχρονα θα βρούμε τον μέσο όρο τους. Στη συνέχεια θα ελέγξουμε κάθε μια από τις 30 θερμοκρασίες αν είναι μικρότερη της μέσης τιμής και θα την μετράμε.

ΠΡΟΓΡΑΜΜΑ θερμοκρασία

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Θ [30], i, M, sum

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ

ΑΡΧΗ

sum \leftarrow 0

ΓΡΑΨΕ "Δώσε 30 θερμοκρασίες"

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 30

ΔΙΑΒΑΣΕ Θ [i]

sum \leftarrow sum + Θ [i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΜΟ \leftarrow sum/30

ΓΡΑΨΕ "Μέση θερμοκρασία:", ΜΟ

M \leftarrow 0

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 30

ΑΝ Θ[i] < ΜΟ **ΤΟΤΕ**

M \leftarrow M + 1

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ "Ημέρες με θερμοκρασία<Μέση Θερ:", M

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ θερμοκρασία

2. Να γραφεί πρόγραμμα που να γεμίζει ένα δισδιάστατο πίνακα 4x3, αν γραμμή με ακέραιους αριθμούς που δίνονται από το πληκτρολόγιο. Στη συνέχεια να υπολογίζει και να τυπώνει:

1. Άθροισμα ανά γραμμή
2. Άθροισμα ανά στήλη
3. Το συνολικό άθροισμα όλων των αριθμών του πίνακα.

Λύση

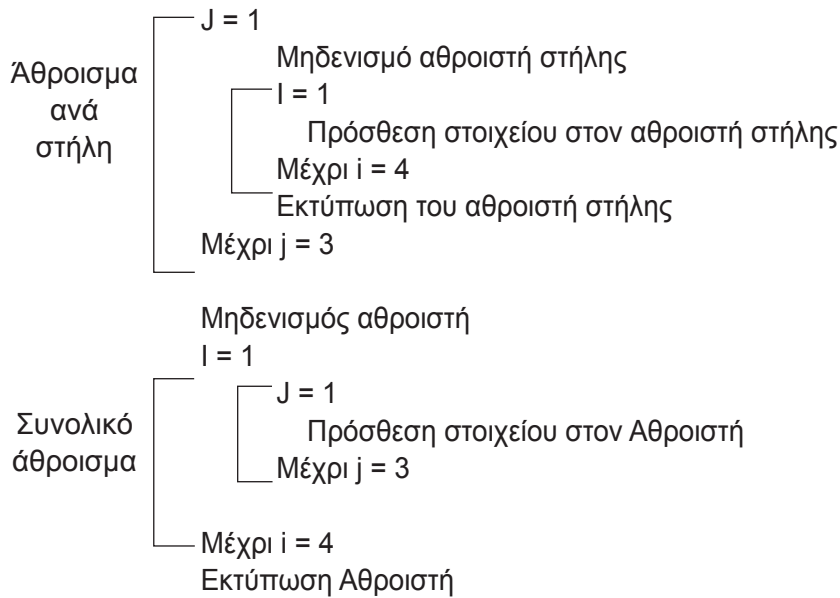
Γενική μορφή:

Δίνω τα δεδομένα

- i = 1
- j = 1
- Πληκτρολογώ τα δεδομένα και τα καταχωρώ στον πίνακα
- Μέχρι j = 3
- Μέχρι i = 4

Άθροισμα ανά γραμμή

- l = 1
- Μηδενισμό αθροιστή γραμμής
- J = 1
- Πρόσθεση στοιχείου στον αθροιστή γραμμής
- Μέχρι j = 3
- Εκτύπωση του αθροιστή γραμμής
- Μέχρι i = 4



ΠΡΟΓΡΑΜΜΑ Αθροίσματα
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[4,3], i, j, sum_στ, sum_γρ, sum

ΑΡΧΗ

```

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 4
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3
        ΓΡΑΨΕ 'Δώσε αριθμό ακέραιο:'
        ΔΙΑΒΑΣΕ A [i, j]
        ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 4
    sum_γρ ← 0
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3
        sum_γρ ← sum_γρ + A [i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΡΑΨΕ 'Αθροισμα της', i, 'γραμμής', 'είναι', sum_γρ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3
    sum_στ ← 0
    ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 4
        sum_στ ← sum_στ + A [i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΡΑΨΕ 'Αθροισμα της', j, 'στήλης', 'είναι:', sum_στ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

sum ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 4
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3
        sum ← sum + A [i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Συνολικό άθροισμα:', sum
  
```

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Αθροίσματα

Παρατήρηση

- Αν τα ζητούμενα σ' ένα δισδιάστατο πίνακα αφορούν γραμμή ή στήλη, τότε οι αρχικές τιμές για άθροισμα, ελάχιστο, μέγιστο, κ.λ.π. δίνονται ανάμεσα στις δύο «ΓΙΑ...».
 - Ενώ αν τα ζητούμενα σ' ένα δισδιάστατο πίνακα αφορούν όλο το πίνακα, τότε οι αρχικές τιμές δίνονται πριν από τις δύο «ΓΙΑ...».
3. Να γραφεί πρόγραμμα στο οποίο να υπολογίζεται και να τυπώνεται ο μεγαλύτερος αριθμός, από 100 ακέραιους αριθμούς που θα διαβαστούν απ' το πληκτρολόγιο.

Λύση

Ανάλυση

1. θα χρησιμοποιηθεί πίνακας 100 θέσεων για να καταχωρήσω τους αριθμούς
2. θα εφαρμοστεί ο αλγόριθμος του μεγαλύτερου, θέτοντας σαν αρχική τιμή την 1η θέση του πίνακα.

Γενική μορφή:

Εύρεση Max

```
i = 1
  Διαβάζω αριθμό A[i]
Μέχρι i = 100

Καταχωρώ στην Max την A[1] σαν αρχική τιμή

i = 2
  Έλεγχος της Max με το A[i]
Μέχρι i = 100

Τυπώνω Max
```

ΠΡΟΓΡΑΜΜΑ Max_Μονοδιάστατο

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], i, Max

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100

ΓΡΑΨΕ 'Δώσε αριθμό:'

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Max ← A[1]

ΓΙΑ i **ΑΠΟ** 2 **ΜΕΧΡΙ** 100

ΑΝ A[i] > Max **ΤΟΤΕ**

 Max ← A[i]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

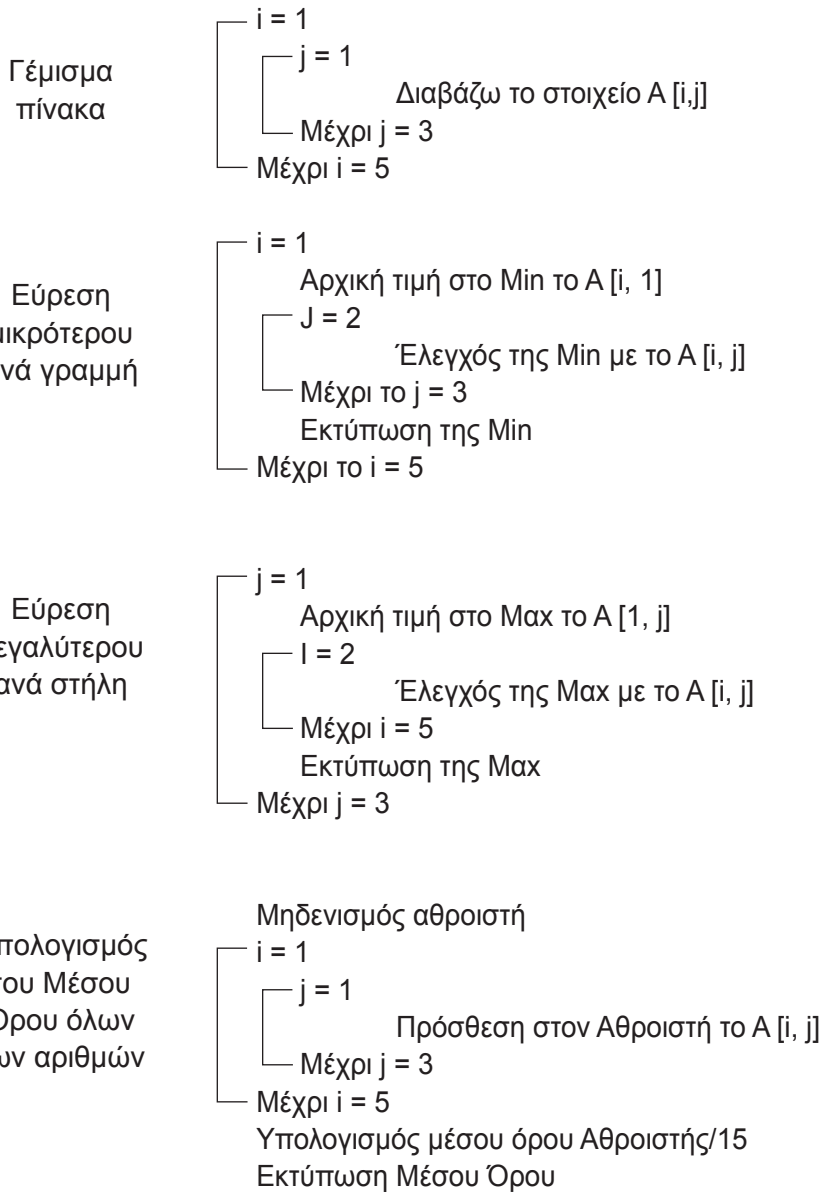
ΓΡΑΨΕ 'Μεγαλύτερος ο', Max

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Max_Μονοδιάστατο

4. Να γραφεί πρόγραμμα που να γεμίζει ένα δισδιάστατο πίνακα 5x3 με ακέραιους αριθμούς, από το πληκτρολόγιο. Στη συνέχεια να υπολογίζει και να τυπώνει:
- α) το μικρότερο στοιχείο κάθε γραμμής
 - β) το μεγαλύτερο στοιχείο κάθε στήλης
 - γ) το μέσο όρο όλων των αριθμών.

Λύση

Γενική μορφή:



ΠΡΟΓΡΑΜΜΑ Min_Max_MO

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[5,3], i, j, Max, Min, sum

ΠΡΑΓΜΑΤΙΚΕΣ: MO

ΑΡΧΗ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3

ΓΡΑΨΕ Δώσε ακέραιο αριθμό

ΔΙΑΒΑΣΕ A [i, j]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5

Min ← A[i, 1]

ΓΙΑ j ΑΠΟ 2 ΜΕΧΡΙ 3

ΑΝ A [i, j] < Min **ΤΟΤΕ**

Min ← A [i, j]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Ο μικρότερος αριθμός της', i, 'γραμμής=', Min

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3

Max ← A[1, j]

ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 5

ΑΝ A [i, j] > Max **ΤΟΤΕ**

Max ← A [i, j]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Ο μεγαλύτερος αριθμός της', j, 'στήλης=', Max

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

sum ← 0

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5

ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 3

sum ← sum + A [i, j]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

MO ← sum/15

ΓΡΑΨΕ 'Μέσος όρος=', MO

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Min_Max_MO

4.6. Τυπικές επεξεργασίες πινάκων

Τα προγράμματα που χρησιμοποιούν πίνακες «κάνουν» συγκεκριμένες επεξεργασίες, οι οποίες είναι:

- Υπολογισμός αθροισμάτων στοιχείων του πίνακα
- Εύρεση του μέγιστου ή ελάχιστου στοιχείου
- Ταξινόμηση των στοιχείων του πίνακα
- Αναζήτηση ενός στοιχείου του πίνακα
- Συγχώνευση δύο πινάκων.

4.6.1. Ταξινόμηση πίνακα

- Ταξινόμηση ονομάζουμε την τακτοποίηση των κόμβων μιας δομής με μια ιδιαίτερη σειρά.
- Η ταξινόμηση μπορεί να γίνει:
 1. κατ' αύξουσα τάξη (δηλ. από το μικρότερο προς το μεγαλύτερο)
 2. κατά φθίνουσα τάξη (δηλ. από το μεγαλύτερο προς το μικρότερο)
- Υπάρχουν διάφοροι αλγόριθμοι ταξινόμησης πίνακα. Εδώ θα αναπτυχθεί ο αλγόριθμος της ταξινόμησης ευθείας ανταλλαγής ή ταξινόμηση φυσαλίδας

4.6.1.1. Ταξινόμηση ευθείας ανταλλαγής ή φυσαλίδας (Bubblesort)

Με τη μέθοδο αυτή της ταξινόμησης συγκρίνουμε και ανταλλάσσουμε ζεύγη γειτονικών στοιχείων μέχρις ότου ταξινομηθούν όλα.

Με τη μέθοδο αυτή κάθε φορά γίνονται διαδοχικές προσπελάσεις στον πίνακα έτσι ώστε το μικρότερο στοιχείο της ακολουθίας να μετακινηθεί προς το αριστερό άκρο του πίνακα.

Περίπτωση 1^η (κατ' αύξουσα τάξη)

ΠΡΟΓΡΑΜΜΑ Ταξινόμηση1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], i, j, K

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Πόσες φορές θα προσπελάσω τον πίνακα

ΓΙΑ j **ΑΠΟ** 2 **ΜΕΧΡΙ** 100

ΓΙΑ j **ΑΠΟ** 100 **ΜΕΧΡΙ** i **ΜΕ_ΒΗΜΑ** -1

ΑΝ A[j-1] > A[j] **ΤΟΤΕ**

K ← A[j-1]

A[j-1] ← A[j]

A[j] ← K

Αντιμετάθεση τιμών

Ποια στοιχεία του πίνακα ελέγχω

Ταξινόμηση του πίνακα A

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
 ΓΡΑΨΕ A[i]
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ταξινόμηση1

- Στην περίπτωση που η ταξινόμηση είναι κατά φθίνουσα τάξη, το μόνο που αλλάζει είναι η σύγκριση των στοιχείων του πίνακα, δηλαδή έχω:

ΑΝ A[j-1] < A[j] ΤΟΤΕ ...

Περίπτωση 2^η

Στην περίπτωση αυτή αναπτύσσεται ο αλγόριθμος ταξινόμησης με τη μέθοδο της φυσαλίδας καλύτερα και εξυπνότερα. Δηλ. όταν ο πίνακας είναι ταξινομημένος, τότε σταματά ο έλεγχος των στοιχείων για ταξινόμηση.

ΠΡΟΓΡΑΜΜΑ Ταξινόμηση1
ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], i, j, K
 ΛΟΓΙΚΕΣ: Flag

ΑΡΧΗ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
 ΔΙΑΒΑΣΕ A[i]
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Η μεταβλητή flag ελέγχει (όταν αλλάξει η τιμή της) από ποιο σημείο και μετά ο πίνακας είναι ταξινομημένος και σταματά η διαδικασία

Ταξινόμηση του πίνακα A

Όταν δεν ισχύει η συνθήκη, σημαίνει ότι το επόμενο στοιχείο είναι μικρότερο ή ίσο του προηγούμενου. Δηλαδή βρέθηκε το σημείο, από το οποίο και μετά, ο πίνακας είναι ταξινομημένος. Η flag από **ΑΛΗΘΗΣ** γίνεται **ΨΕΥΔΗΣ** και τερματίζει η διαδικασία ταξινόμησης.

Flag ← ΑΛΗΘΗΣ
 i ← 2
ΟΣΟ i <= 100 ΚΑΙ flag = ΑΛΗΘΗΣ **ΕΠΑΝΑΛΑΒΕ**
 flag ← ΨΕΥΔΗΣ
 ΓΙΑ j ΑΠΟ 100 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1
ΑΝ A[j-1] > A[j] **ΤΟΤΕ**
 K ← A[j - 1]
 A[j-1] ← A[j]
 A[j] ← K
 flag ← ΑΛΗΘΗΣ
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 i ← i + 1
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ A[j-1] > A[j] **ΤΟΤΕ**
 K ← A[j - 1]
 A[j-1] ← A[j]
 A[j] ← K
 flag ← ΑΛΗΘΗΣ
ΤΕΛΟΣ_ΑΝ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
 ΓΡΑΨΕ A[i]
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ταξινόμηση1

4.6.2. Αναζήτηση στοιχείου πίνακα

Το πρόβλημα της αναζήτησης ενός στοιχείου σε πίνακα είναι ιδιαίτερα χρήσιμο σ' ένα μεγάλο πλήθος εφαρμογών.

Υπάρχουν αρκετοί αλγόριθμοι για την αναζήτηση στοιχείου σε πίνακα που αποσκοπούν στην ταχύτητα εύρεσης του αναζητούμενου στοιχείου. Εμείς θα ασχοληθούμε με δύο αλγορίθμους :

1. Την Σειριακή αναζήτηση, και
2. Την Δυαδική αναζήτηση

4.6.2.1. Σειριακή αναζήτηση

Η πιο απλή μορφή αναζήτησης στοιχείου σε πίνακα είναι η **σειριακή ή γραμμική μέθοδος**.

Με τη μέθοδο αυτή ξεκινάμε από την αρχή του πίνακα και συγκρίνουμε κάθε τιμή του μια προς μια, με το στοιχείο που αναζητάμε.

Παρακάτω θα δοθούν δύο περιπτώσεις αλγόριθμου αναζήτησης για στοιχείο που είναι μοναδικό, δηλ. υπάρχει μόνο μια φορά και για στοιχείο που μπορεί να υπάρχει περισσότερες από μια φορές στον πίνακα.

Περίπτωση 1^η

Αναζήτηση του στοιχείου (key) σε ένα πίνακα A[100]. Το στοιχείο (key) είναι μοναδικό. Σε περίπτωση που βρεθεί να τυπώνεται η αντίστοιχη θέση του και το μήνυμα "Βρέθηκε", αλλιώς να τυπώνεται το μήνυμα "Δεν βρέθηκε".

ΠΡΟΓΡΑΜΜΑ Αναζήτηση1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], I, KEY

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

 i ← 1

ΟΣΟ A[i] <> key **ΚΑΙ** i <= 100 **ΕΠΑΝΑΛΑΒΕ**

 i ← i + 1

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ A[i] = key **ΤΟΤΕ**

ΓΡΑΨΕ 'Βρέθηκε στη θέση:', i

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν βρέθηκε'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Αναζήτηση1

Περίπτωση 2^η

Αναζήτηση του στοιχείου (key) σε ένα πίνακα A[100]. Το στοιχείο key υπάρχει πιθανότητα να βρίσκεται περισσότερες από μια φορές στον πίνακα.

Στην περίπτωση που βρεθεί να τυπώνονται οι θέσεις που βρέθηκε, αλλιώς να τυπώνεται το μήνυμα "Δεν βρέθηκε".

ΠΡΟΓΡΑΜΜΑ Αναζήτηση2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], I, KEY

ΑΡΧΗ

Περίπτωση 1^η: Θεωρούμε ότι τα στοιχεία του πίνακα A είναι ταξινομημένα κατά αύξουσα σειρά.

ΠΡΟΓΡΑΜΜΑ Δυαδική_Αναζήτηση1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], KEY, τέλος, αρχή, μέση

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Δώσε αριθμό για αναζήτηση στον πίνακα A.

ΔΙΑΒΑΣΕ Key

Αρχή \leftarrow 1

Τέλος \leftarrow 100

Μέση \leftarrow (αρχή + τέλος) div 2

ΟΣΟ αρχή \leq τέλος **ΚΑΙ** A[μέση] \neq Key **ΕΠΑΝΑΛΑΒΕ**

ΑΝ Key < A[μέση] **ΤΟΤΕ**

Τέλος \leftarrow μέση-1

ΑΛΛΙΩΣ

Αρχή \leftarrow μέση + 1

ΤΕΛΟΣ_ΑΝ

Μέση \leftarrow (αρχή + τέλος) div 2

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ A[μέση] = Key **ΤΟΤΕ**

ΓΡΑΨΕ 'Βρέθηκε στη θέση ',μέση

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν βρέθηκε'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Δυαδική_Αναζήτηση1

Περίπτωση 2^η: Θεωρούμε ότι τα στοιχεία του πίνακα A είναι ταξινομημένα κατά φθίνουσα σειρά.

ΠΡΟΓΡΑΜΜΑ Δυαδική_Αναζήτηση2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], KEY, τέλος, αρχή, μέση

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Δώσε αριθμό για αναζήτηση στον πίνακα A.

ΔΙΑΒΑΣΕ Key

Αρχή \leftarrow 1

Τέλος \leftarrow 100

μέση \leftarrow (αρχή + τέλος) div 2

ΟΣΟ αρχή \leq τέλος **ΚΑΙ** A[μέση] \neq Key **ΕΠΑΝΑΛΑΒΕ**

ΑΝ Key < A[μέση] **ΤΟΤΕ**

Αρχή \leftarrow μέση + 1

ΑΛΛΙΩΣ

Τέλος \leftarrow μέση - 1

ΤΕΛΟΣ_ΑΝ

Μέση \leftarrow (αρχή + τέλος) div 2

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΑΝ A[μέση] = Key **ΤΟΤΕ**

ΓΡΑΨΕ 'Βρέθηκε στη θέση ',μέση

ΑΛΛΙΩΣ

ΓΡΑΨΕ 'Δεν βρέθηκε'

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Δυαδική_Αναζήτηση2

4.7. Αλγόριθμος συγχώνευσης δύο πινάκων

Η συγχώνευση είναι μια από τις βασικές επεξεργασίες σε πίνακες. Σκοπός της είναι η δημιουργία ενός πίνακα από τα δεδομένα δύο πινάκων. Υπάρχουν δύο περιπτώσεις:

Περίπτωση 1^η

Έστω πίνακας A[25] με τα ονόματα του 1ου τμήματος πληροφορικής της Γ' Λυκείου και B[21] τα ονόματα του 2ου τμήματος πληροφορικής της Γ' Λυκείου. Να δημιουργηθεί πίνακας με τα ονόματα και των δύο τμημάτων.

Λύση

ΠΡΟΓΡΑΜΜΑ Συγχώνευση1

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[25], B[25], i, κ

ΑΡΧΗ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 25

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 21

ΔΙΑΒΑΣΕ B[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

} Είσοδος
δεδομένων



ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 25

 Γ[i] ← A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

κ ← 25

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 21

 Γ[i+κ] ← B[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

} Συγχώνευση
των πινάκων



ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 46

ΓΡΑΨΕ Γ[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Συγχώνευση1

Περίπτωση 2^η

Έστω ότι έχουμε τα δεδομένα της παραπάνω άσκησης και οι πίνακες A,B είναι ταξινομημένοι κατά αύξουσα τάξη. Να δημιουργηθεί ο πίνακας Γ με τα ονόματα και των δύο τμημάτων με απευθείας ταξινόμηση.

Λύση

Ανάλυση

1. Στη περίπτωση αυτή ελέγχουμε ένα στοιχείο του πίνακα A με ένα στοιχείο του πίνακα B (ξεκινώντας από την πρώτη θέση).
2. Το στοιχείο που είναι μικρότερο καταχωρείται στο πίνακα Γ και αυξάνεται κατά μία μονάδα η θέση του πίνακα του οποίου πήρα το στοιχείο του και το καταχώρησα στον πίνακα Γ.
3. Η διαδικασία αυτή σταματά όταν τελειώσουν τα στοιχεία που συγκρίνουμε σε έναν από δύο πίνακες, τον A ή B πίνακα.
4. Στη συνέχεια, από τον πίνακα που περίσσεψαν στοιχεία, τα καταχωρώ στο πίνακα Γ.

ΠΡΟΓΡΑΜΜΑ Συγχώνευση2

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[25], B[25], i, κ

ΑΡΧΗ

Είσοδος
δεδομένων

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 25
 ΔΙΑΒΑΣΕ A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 21
 ΔΙΑΒΑΣΕ B[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

i ← 1
j ← 1
κ ← 1
ΟΣΟ i ≤ 25 **ΚΑΙ** j ≤ 21 **ΕΠΑΝΑΛΑΒΕ**
 ΑΝ A[i] < B[j] **ΤΟΤΕ**
 Γ[κ] ← A[i]
 i ← i + 1
 ΑΛΛΙΩΣ
 Γ[κ] ← B[j]
 j ← j + 1
 ΤΕΛΟΣ_ΑΝ
 κ ← κ + 1

Συνένωση 2
ταξινομημένων
πινάκων με
απευθείας
ταξινόμηση

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ i > 25 **ΤΟΤΕ**
 ΓΙΑ i **ΑΠΟ** j **ΜΕΧΡΙ** 21
 Γ[κ] ← B[i]
 κ ← κ + 1
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΛΛΙΩΣ
 ΓΙΑ i **ΑΠΟ** i **ΜΕΧΡΙ** 25
 Γ[κ] ← A[i]
 κ ← κ + 1
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΑΝ
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 46
 ΓΡΑΨΕ Γ[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Συγχώνευση2

4.8. Παράλληλοι πίνακες

Όταν έχουμε εγγραφές με πεδία διαφορετικού τύπου οι οποίες καταχωρούνται σε πίνακες τότε κάθε πεδίο είναι ένας διαφορετικός πίνακας, λόγω διαφορετικού τύπου. Οι πίνακες αυτοί συνδέονται μεταξύ τους με τέτοιο τρόπο ώστε η κάθε εγγραφή να έχει όλα τα στοιχεία της με την ίδια τιμή δείκτη, δηλ. να βρίσκονται στην ίδια θέση στους πίνακες.

Π.χ. αν έχουμε εγγραφές που αφορούν στοιχεία μισθοδοσίας υπαλλήλων μιας εταιρείας.

ΕΠΩΝΥΜΟ	ΟΝΟΜΑ	ΟΙΚΟΓΕΝΙΑΚΗ ΚΑΤΑΣΤΑΣΗ	ΑΡΙΘΜΟΣ ΠΑΙΔΙΩΝ	ΜΙΣΘΟΣ	ΚΡΑΤΗΣΕΙΣ
ΠΑΠΠΑΣ	ΓΕΩΡΓΙΟΣ	ΕΓΓΑΜΟΣ	2	1,400	10%
ΓΕΩΡΓΙΟΥ	ANNA	ΑΓΑΜΗ	0	1,150	20%
ΝΑΝΟΥ	ΣΟΦΙΑ	ΕΓΓΑΜΗ	1	1,520	16%
...

Τα παραπάνω δεδομένα μπορώ να τα κρατήσω σε 6 μονοδιάστατος πίνακες, γιατί άλλα στοιχεία είναι χαρακτήρες και άλλα αριθμοί. Οπότε είναι αδύνατη η χρήση δισδιάστατου πίνακα. Επίσης, η τοποθέτηση των δεδομένων θα γίνει με τέτοιο τρόπο, ώστε το επώνυμο, όνομα, οικογενειακή κατάσταση, αριθμός παιδιών, μισθός και κρατήσεις ενός υπαλλήλου να βρίσκονται στην ίδια γραμμή σ' όλους τους πίνακες.

Παράδειγμα 1

Έχουμε 50 ονόματα εταιρειών και τις αντίστοιχες εισπράξεις τους. Να τυπωθούν τα ονόματα των εταιρειών με εισπράξεις μεγαλύτερες από το μέσο όρο εισπράξεων και τα ονόματα των εταιρειών με εισπράξεις μικρότερες από το μέσο όρο εισπράξεων.

Λύση

Ανάλυση

1. Το όνομα και η είσπραξη κάθε εταιρείας καταχωρούνται σε διαφορετικούς πίνακες που είναι παράλληλοι. Οπότε τα δεδομένα διαβάζονται ταυτόχρονα.
2. Βρίσκουμε το μέσο όρο από τον πίνακα των εισπράξεων.
3. Ελέγχουμε κάθε είσπραξη αν είναι μεγαλύτερη του μέσου όρου εισπράξης και τυπώνουμε το αντίστοιχο όνομα εταιρείας.
4. Ελέγχουμε κάθε είσπραξη αν είναι μικρότερη του μέσου όρου εισπράξης και τυπώνουμε το αντίστοιχο όνομα εταιρείας.

ΠΡΟΓΡΑΜΜΑ Εταιρεία
ΜΕΤΑΒΛΗΤΕΣ

ΧΑΡΑΚΤΗΡΕΣ: Ετ[50]
ΠΡΑΓΜΑΤΙΚΕΣ: ΕΙΣ[50], ΜΟ, sum
ΑΚΕΡΑΙΕΣ: i

ΑΡΧΗ

Γέμισμα
παράλληλων
πινάκων { **ΓΡΑΨΕ** 'Δώσε 50 ονόματα εταιρειών και τις αντίστοιχες εισπράξεις τους'
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 50
ΔΙΑΒΑΣΕ Ετ[i], ΕΙΣ[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Υπολογισμός
ΜΟ
εισπράξεων { sum ← 0
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 50
sum ← sum + ΕΙΣ[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΜΟ ← sum/50

Τυπώνονται
ονόματα
εταιρειών με
εισπράξεις
>ΜΟ { **ΓΙΑ** i **ΑΠΟ** 1 **ΜΕΧΡΙ** 50
ΑΝ ΕΙΣ[i] > ΜΟ **ΤΟΤΕ**
ΓΡΑΨΕ Ετ[i]
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Τυπώνονται
ονόματα
εταιρειών με
εισπράξεις
<ΜΟ { **ΓΙΑ** i **ΑΠΟ** 1 **ΜΕΧΡΙ** 50
ΑΝ ΕΙΣ[i] < ΜΟ **ΤΟΤΕ**
ΓΡΑΨΕ Ετ[i]
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Εταιρεία

Παράδειγμα 2 (ταξινόμηση σε παράλληλους πίνακες)

Όταν ταξινομούμε παράλληλους πίνακες, κάνουμε τον έλεγχο ως προς τον πίνακα που ζητείται η ταξινόμηση, αλλά στην αντιμετάθεση των στοιχείων αντιμεταθέτουμε τις αντίστοιχες θέσεις όλων των παράλληλων πινάκων ως προς αυτόν που γίνεται η ταξινόμηση.

Π.χ. Να γίνει πρόγραμμα που δίνω τα ονόματα και τους μέσους όρους 50 μαθητών της Α' Λυκείου ενός σχολείου. Να τυπωθεί μια κατάσταση ταξινομημένη ως προς το μέσο όρο κατά φθίνουσα τάξη, δηλαδή από τον μεγαλύτερο βαθμό προς τον μικρότερο.

Λύση

Ανάλυση

1. Καταχωρώ σε δύο παράλληλους πίνακες τα ονόματα και τους μέσους όρους 50 μαθητών
2. Εφαρμόζω τον αλγόριθμο της ταξινόμησης στον πίνακα των μέσων όρων.

ΠΡΟΓΡΑΜΜΑ Μαθητές
ΜΕΤΑΒΛΗΤΕΣ

ΧΑΡΑΚΤΗΡΕΣ: Ον[50], t2
ΠΡΑΓΜΑΤΙΚΟΙ: ΜΟ[50], t1
ΑΚΕΡΑΙΕΣ: i, j

ΑΡΧΗ

Γέμισμα
παράλληλων
πινάκων

```
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 50
  ΓΡΑΨΕ 'Δώσε όνομα και Μέσο Όρο Μαθητή:'
  ΔΙΑΒΑΣΕ Ον[i], ΜΟ[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Ταξινόμηση
ως προς
τους ΜΟ

```
ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 50
  ΓΙΑ j ΑΠΟ 50 ΜΕΧΡΙ i ΜΕ ΒΗΜΑ -1
    ΑΝ ΜΟ [j] > ΜΟ [j - 1] ΤΟΤΕ
      Αντιμετάθεση
      ΜΟ { t1 ← ΜΟ [j]
          ΜΟ [j] ← ΜΟ[j-1]
          ΜΟ[j-1] ← t1
      Αντιμετάθεση
      Ονομάτων { t2 ← ΟΝ[j]
                ΟΝ[j] ← ΟΝ[j - 1]
                ΟΝ[j-1] ← t2
    ΤΕΛΟΣ_ΑΝ
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Εκτύπωση
πινάκων

```
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 50
  ΓΡΑΨΕ ΟΝ[i], ΜΟ [i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μαθητές
```

Παράδειγμα 3 (Διπλή ταξινόμηση σε παράλληλους πίνακες)

Χρησιμοποιώντας το προηγούμενο παράδειγμα να εκτυπωθεί μια κατάσταση μαθητών ταξινομημένη ως προς το Μέσο Όρο τους κατά φθίνουσα τάξη και σε περίπτωση ισοβαθμίας η ταξινόμηση να γίνεται ως προς το όνομα κατά αύξουσα τάξη (αλφαβητικά).

Λύση

Ανάλυση

1. Καταχωρώ σε δύο παράλληλους πίνακες τα ονόματα και τους Μέσους Όρους 50 μαθητών.
2. Ταξινομώ τους πίνακες ως προς τον Μέσο Όρο.
3. Ελέγχω μέσα στην ταξινόμηση αν οι Μέσοι Όροι είναι ίσοι και ταξινομώ ως προς το Όνομα του μαθητή.

ΠΡΟΓΡΑΜΜΑ Διπλή_Ταξινόμηση ΜΕΤΑΒΛΗΤΕΣ

ΧΑΡΑΚΤΗΡΕΣ: Ον[50], t2
ΠΡΑΓΜΑΤΙΚΟΙ: ΜΟ[50], t1
ΑΚΕΡΑΙΕΣ: i, j

ΑΡΧΗ

```
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 50
  ΓΡΑΨΕ 'Δώσε όνομα και Μέσο Όρο Μαθητή:'
  ΔΙΑΒΑΣΕ Ον[i], ΜΟ[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

```
ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 50
  ΓΙΑ j ΑΠΟ 50 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1
```

ΑΝ $MO[j] > MO[j - 1]$ **ΤΟΤΕ**

$t1 \leftarrow MO[j]$

$MO[j] \leftarrow MO[j-1]$

$MO[j-1] \leftarrow t1$

$t2 \leftarrow ON[j]$

$ON[j] \leftarrow ON[j - 1]$

$ON[j-1] \leftarrow t2$

ΤΕΛΟΣ_ΑΝ

ΑΝ $MO[j]=MO[j-1]$ **ΤΟΤΕ**

ΑΝ $ON[j] < ON[j-1]$ **ΤΟΤΕ**

$t2 \leftarrow ON[j]$

$ON[j] \leftarrow ON[j - 1]$

$ON[j - 1] \leftarrow t2$

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 50

ΓΡΑΨΕ $ON[i]$, $MO[i]$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Διπλή_Ταξινόμηση

4.9. Λυμένες ασκήσεις

4.9.1. Άσκηση 1. Μονοδιάστατος πίνακας

Έστω μονοδιάστατος πίνακας ακέραιων A , 100 θέσεων. Να γραφτεί πρόγραμμα που να πραγματοποιεί τα παρακάτω:

- i. Γέμισμα πίνακα με ακέραιους αριθμούς.
- ii. Εκτύπωση πίνακα από την 1η θέση μέχρι την 100η θέση.
- iii. Εκτύπωση πίνακα από την 100η θέση μέχρι την 1η θέση.
- iv. Εκτύπωση του αθροίσματος των στοιχείων του πίνακα.
- v. Εκτύπωση του μέσου όρου των στοιχείων του πίνακα.
- vi. Εκτύπωση του αθροίσματος των στοιχείων του πίνακα που βρίσκονται στις ζυγές θέσεις του πίνακα και του αθροίσματος που βρίσκονται στις μονές θέσεις.
- vii. Εκτύπωση του γινόμενου των 10 πρώτων στοιχείων του πίνακα A .
- viii. Αντιμέταθεση των συμμετρικών στοιχείων του πίνακα A και εκτύπωση του πίνακα A .
- ix. Υπολογισμό του αθροίσματος των 50 πρώτων στοιχείων του πίνακα A και του αθροίσματος των 50 τελευταίων στοιχείων του πίνακα A , αν τα δύο αθροίσματα είναι ίσα να τυπώνεται το μήνυμα "Ίσα αθροίσματα".
- x. Εμφάνιση το μηνύματος "Πίνακας συμμετρικός", αν τα συμμετρικά στοιχεία του πίνακα είναι ίσα.
- xi. Εκτύπωση του πλήθους των στοιχείων που είναι μικρότερα από το Μέσο Όρο τους.
- xii. Αναζήτηση της τιμής «42» και αν βρεθεί να τυπωθεί η θέση της.
- xiii. Ταξινόμηση κατά φθίνουσα τάξη του πίνακα A .
- xiv. Έλεγχο του πίνακα A αν είναι ταξινομημένος και να βγάλει κατάλληλο μήνυμα.

Λύση

ΠΡΟΓΡΑΜΜΑ Μονοδιάστατος_πίνακας

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], i, j, sum, sum1, sum2, suma, sumμ, K, μ, GIN

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ

ΛΟΓΙΚΕΣ: flag

ΑΡΧΗ

- i. Γέμισμα πίνακα
- ii.
- iii.
- iv.
- v.
- vi. Ελέγχουμε την κάθε θέση πίνακα A[i] αν έχει άρτια θέση ($i \bmod 2 = 0$) ή όχι και προσθέτουμε το στοιχείο στον κατάλληλο αθροιστή.
- vii. Η μεταβλητή **GIN** είναι ο πολλαπλός με αρχική τιμή το 1, το οποίο είναι το ουδέτερο στοιχείο του πολλαπλασιασμού
- ΓΡΑΨΕ** 'Δώσε 100 ακέραιους αριθμούς:'
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100
 ΔΙΑΒΑΣΕ A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- ΓΙΑ** i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100
 ΓΡΑΨΕ A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- ΓΙΑ** i **ΑΠΟ** 100 **ΜΕΧΡΙ** 1 **ΜΕ_ΒΗΜΑ** -1
 ΓΡΑΨΕ A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- sum ← 0
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100
 sum ← sum + A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Αθροισμα στοιχείων:', sum
- ΜΟ ← sum/100
ΓΡΑΨΕ 'Μέσος Όρος στοιχείων =', ΜΟ
- suma ← 0
sumμ ← 0
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 100
 ΑΝ i mod 2 = 0 **ΤΟΤΕ**
 suma ← suma + A[i]
 ΑΛΛΙΩΣ
 sumμ ← sumμ + A[i]
 ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Αθροισμα αριθμών σε άρτιες θέσεις:', suma
ΓΡΑΨΕ 'Αθροισμα αριθμών σε μονές θέσεις:', sumμ
- GIN ← 1
ΓΙΑ i **ΑΠΟ** 1 **ΜΕΧΡΙ** 10
 GIN ← GIN * A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Το γινόμενο των 10 πρώτων αριθμών :', GIN

viii. Αντιμεταθέτει τα:

i	100-i+1
1	100-1+1=100
2	100-2+1=99
...
50	100-50+1=49

```

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 50
    K ← A[i]
    A[i] ← A[100 - i + 1]
    A[100 - i + 1] ← K
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
    ΓΡΑΨΕ A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  
```

} Αντιμετάθεση

ix.

```

sum1 ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 50
    sum1 ← sum1 + A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
sum2 ← 0
ΓΙΑ i ΑΠΟ 51 ΜΕΧΡΙ 100
    sum2 ← sum2 + A[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ sum1 = sum2 ΤΟΤΕ
    ΓΡΑΨΕ 'Ίσα αθροίσματα'
ΤΕΛΟΣ_ΑΝ
  
```

} Υπολογισμός αθροίσματος των 50 πρώτων αριθμών

} Υπολογισμός αθροίσματος των 50 επόμενων αριθμών

x. Έλεγχος των
A[1], A[100]
A[2], A[99]
A[3], A[98]
...
A[49], A[50]

```

flag ← ΨΕΥΔΗΣ
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 50
    ΑΝ A[i] <> A[100 - i + 1] ΤΟΤΕ
        flag ← ΑΛΗΘΗΣ
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ flag = ΨΕΥΔΗΣ ΤΟΤΕ
    ΓΡΑΨΕ 'Πίνακας συμμετρικός'
ΤΕΛΟΣ_ΑΝ
  
```

xi.

```

μ ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
    ΑΝ A[i] < ΜΟ ΤΟΤΕ
        μ ← μ + 1
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Πλήθος:', μ
  
```

xi.
Αναζήτηση

```

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100
    ΑΝ A[i] = 42 ΤΟΤΕ
        ΓΡΑΨΕ i
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
  
```

```

ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 100
    ΓΙΑ j ΑΠΟ 100 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1
        ΑΝ A[j] > A[j - 1] ΤΟΤΕ
            K ← A[j]
            A[j] ← A[j-1]
            A[j-1] ← K
        ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

xiii. Ταξινόμηση

xiv. Έλεγχος αν ο πίνακας είναι ταξινομημένος

```

flag ← ΑΛΗΘΗΣ
ΓΙΑ i ΑΠΟ 100 ΜΕΧΡΙ 2 ΜΕ_ΒΗΜΑ -1
    ΑΝ A[i] > A[i - 1] ΤΟΤΕ
        flag ← ΨΕΥΔΗΣ
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΑΝ flag = ΑΛΗΘΗΣ ΤΟΤΕ
    ΓΡΑΨΕ 'Πίνακας ταξινομημένος κατά φθίνουσα τάξη'
ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Μονοδιάστατος_πίνακας

```

4.9.2. Άσκηση 2. Δισδιάστατος πίνακας

Έστω δισδιάστατος πίνακας A, 10 x 20 με ακέραιους αριθμούς. Να γραφτεί πρόγραμμα που να:

- i. Γεμίζει τον πίνακα A κατά γραμμή.
- ii. Γεμίζει τον πίνακα A κατά στήλη.
- iii. Να τυπώνει τα στοιχεία του πίνακα A, εμφανίζοντάς τα κατά γραμμή.
- iv. Να τυπώνει τα στοιχεία του πίνακα A, εμφανίζοντάς τα κατά στήλη.
- v. Να υπολογίζει τα αθροίσματα των γραμμών του πίνακα A, να τα τυπώνει και να τα καταχωρεί στον πίνακα ΑΘΡ_ΓΡ[20].
- vi. Να υπολογίζει τα αθροίσματα των στηλών του πίνακα A, να τα τυπώνει και να τα καταχωρεί στον πίνακα ΑΘΡ_ΣΤ[20].
- vii. Να υπολογίζει το άθροισμα όλων των στοιχείων του πίνακα A.
- viii. Να εμφανίζει το μέσο όρο των στοιχείων του πίνακα A.
- ix. Να τυπώνει τον μεγαλύτερο αριθμό της 12ης στήλης.
- x. Να τυπώνει τον μικρότερο αριθμό της 7ης γραμμής.
- xi. Να τυπώνει τον μεγαλύτερο και τον μικρότερο αριθμό όλου του πίνακα.
- xii. Να τυπώνει πόσα στοιχεία είναι μεγαλύτερα του μέσου όρου.
- xiii. Να διαβάζει έναν ακέραιο και να τον αναζητά στον πίνακα, όταν τον βρει να τυπώνεται η θέση του.
- xiv. Να ταξινομεί τα στοιχεία κάθε γραμμής κατά φθίνουσα τάξη.
- xv. Να ταξινομεί τα στοιχεία κάθε στήλης κατά αύξουσα τάξη.

Λύση

ΠΡΟΓΡΑΜΜΑ Δισδιάστατος πίνακας ΜΕΤΑΒΛΗΤΕΣ

sum

ΑΚΕΡΑΙΕΣ: A[10,20], ΑΘΡ_ΓΡ[10],ΑΘΡ_ΣΤ[20],i, j, sum

ΠΡΑΓΜΑΤΙΚΕΣ: ΜΟ

ΑΡΧΗ

- i. Διάβασμα ανά γραμμή
 - ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10 → γραμμή
 - ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20 → στήλη
 - ΓΡΑΨΕ 'Δώσε ακέραιο αριθμό'
 - ΔΙΑΒΑΣΕ A[i, j]
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- ii. Διάβασμα ανά στήλη
 - ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20 → στήλη
 - ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10 → γραμμή
 - ΓΡΑΨΕ 'Δώσε ακέραιο αριθμό'
 - ΔΙΑΒΑΣΕ A[i, j]
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- iii. Εκτύπωση ανά γραμμή
 - ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
 - ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
 - ΓΡΑΨΕ A[i]
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- iv. Εκτύπωση ανά στήλη
 - ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
 - ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
 - ΓΡΑΨΕ A[i, j]
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- v. Άθροισμα ανά γραμμή.
 - ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
 - sum ← 0
 - ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
 - sum ← sum + A[i, j]
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 - ΓΡΑΨΕ 'Άθροισμα', i, 'γραμμής=', sum
 - ΑΘΡ_ΓΡ[i] ← sum
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
- vi. Άθροισμα ανά στήλη.
 - ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
 - sum ← 0
 - ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
 - sum ← sum + A[i, j]
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
 - ΓΡΑΨΕ 'Άθροισμα', j, 'στήλης=', sum
 - ΑΘΡ_ΣΤ[j] ← sum
 - ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

- vii. Άθροισμα όλων των στοιχείων.
- ```

sum ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
 ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
 sum ← sum + A[i, j]
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Άθροισμα όλων των στοιχείων=', sum

```
- viii. Ο Μέσος Όρος
- ```

ΜΟ ← sum/200
ΓΡΑΨΕ 'Μέσος Όρος =', ΜΟ

```
- ix. Εφαρμόζουμε τον αλγόριθμο του μεγαλύτερου για την στήλη 12.
- ```

Μαx ← A[1,12]
ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 10
 ΑΝ A[i,12] > Μαx ΤΟΤΕ
 Μαx ← A[i,12]
 ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Ο μεγαλύτερος της 12ης στήλης =', Μαx

```
- x. Εφαρμόζουμε τον αλγόριθμο του μικρότερου για την γραμμή 7.
- ```

Μin ← A[7, 1]
ΓΙΑ j ΑΠΟ 2 ΜΕΧΡΙ 20
    ΑΝ A[7, j] < Μin ΤΟΤΕ
        Μin ← A[7, j]
    ΤΕΛΟΣ_ΑΝ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Ο μικρότερος της 7ης γραμμής =', Μin

```
- xi. Εφαρμόζουμε τον αλγόριθμο του μεγαλύτερου και του μικρότερου για τον πίνακα Α.
- ```

Μin ← A[1, 1]
Μαx ← A[1, 1]
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
 ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
 ΑΝ A[i, j] < Μin ΤΟΤΕ
 Μin ← A[i, j]
 ΤΕΛΟΣ_ΑΝ
 ΑΝ A[i, j] > Μαx ΤΟΤΕ
 Μαx ← A[i, j]
 ΤΕΛΟΣ_ΑΝ
 ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```
- xii. Μετράμε πόσα στοιχεία είναι μεγαλύτερα του μέσου όρου.
- ```

μ ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
        ΑΝ A[i, j] > ΜΟ ΤΟΤΕ
            μ ← μ + 1
        ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Πλήθος:' μ

```

xiii. Αναζήτηση στοιχείου στον πίνακα **A**.

```

ΓΡΑΨΕ 'Δώσε ακέραιο για αναζήτηση'
ΔΙΑΒΑΣΕ x
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 10
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 20
        ΑΝ x=A[i, j] ΤΟΤΕ
            ΓΡΑΨΕ 'Βρέθηκε στη θέση=', i, j
        ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

xiv. Εφαρμογή της ταξινόμησης για κάθε γραμμή του πίνακα **A**. Δηλαδή κάθε γραμμή την θεωρώ σαν μονοδιάστατο πίνακα και την ταξινομώ

Επανάληψη γι' όλες τις γραμμές

```

ΓΙΑ gr ΑΠΟ 1 ΜΕΧΡΙ 10
    Ταξινόμηση μιας γραμμής
    ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 20
        ΓΙΑ j ΑΠΟ 20 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1
            ΑΝ A[gr, j] > A[gr, j - 1] ΤΟΤΕ
                K ← A[gr, j]
                A[gr,j] ← A[gr, j - 1]
                A[gr, j - 1] ← K
            ΤΕΛΟΣ_ΑΝ
        ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

xv. Εφαρμογή της ταξινόμησης για κάθε στήλη του πίνακα **A**. Δηλαδή κάθε στήλη την θεωρώ σαν μονοδιάστατο πίνακα και την ταξινομώ

Επανάληψη γι' όλες τις στήλες

```

ΓΙΑ st ΑΠΟ 1 ΜΕΧΡΙ 20
    Ταξινόμηση μιας στήλης
    ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 10
        ΓΙΑ j ΑΠΟ 10 ΜΕΧΡΙ i ΜΕ_ΒΗΜΑ -1
            ΑΝ A[j,st]<A[j-1, st] ΤΟΤΕ
                K ← A[j, st]
                A[j,st] ← A[j - 1, st]
                A[j - 1, st] ← K
            ΤΕΛΟΣ_ΑΝ
        ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Δισδιάστατος_πίνακας

```

4.9.3. Άσκηση 3

Ένα κτίριο με γραφεία έχει 5 ορόφους με 12 γραφεία ανά όροφο. Να δημιουργηθεί ένας δισδιάστατος πίνακας 5x12, όπου η κάθε γραμμή θα αντιπροσωπεύει τον όροφο και σε κάθε θέση του πίνακα θα αποθηκεύουμε το πλήθος των υπαλλήλων, που έχει κάθε γραφείο. Να γραφτεί πρόγραμμα που:

- α. Να διαβάζει το πλήθος των υπαλλήλων κάθε γραφείου ανά όροφο και να τους καταχωρεί στον πίνακα $KT_ΓΡ[5,12]$
- β. Να υπολογίζει και να εμφανίζει τον αριθμό και τον όροφο του γραφείου με τους περισσότερους και τους λιγότερους υπαλλήλους.
- γ. Να υπολογίζει και να τυπώνει το πλήθος των υπαλλήλων ανά όροφο.
- δ. Να υπολογίζει και να τυπώνει το σύνολο των υπαλλήλων όλων των γραφείων.

Λύση

Αν δίνουμε μια γραφική παράσταση των πινάκων που θα δημιουργήσουμε και τη σχέση που έχουν μεταξύ τους, θα είχαμε:

Σχεδιασμός Πινάκων

		Γραφεία															
		1	2	3	4	5	6	7	8	9	10	11	12				
$KT_ΓΡ =$	Οροφοί	1	3	5	10	2	4	7	11	9	2	1	3	5	$(=3+5+10+...+5)$	1	62
	2	2	6	3	7	9	3	5	6	2	9	10	15	$(=2+6+3+...+15)$	2	77	
	3	24	32	9	10	3	6	5	3	9	8	5	6	←→	3	120	
	4	7	4	9	3	20	2	4	5	2	6	5	9	$AΘ_ΟΡ =$	4	76	
	5	5	2	11	12	21	5	9	3	2	4	3	10		5	87	

Ανάλυση

1. Γεμίζουμε το πίνακα ανά γραμμή, αφού ο κάθε όροφος εκφράζει γραμμή.
2. Εφαρμόζουμε τον αλγόριθμο του ελάχιστου και μέγιστου ενώ ταυτόχρονα κρατάμε σε 2 μεταβλητές τη στήλη και τη γραμμή που βρέθηκε αντίστοιχα ο μικρότερος και ο μεγαλύτερος.
3. Εφαρμόζουμε άθροισμα ανά γραμμή δημιουργώντας ένα μονοδιάστατο και παράλληλο πίνακα, ως προς τις γραμμές, γιατί ο κάθε όροφος είναι μια γραμμή πίνακα.
4. Αθροίζουμε σ' ένα αθροιστή όλων τα στοιχεία του μονοδιάστατου πίνακα που δημιουργήσαμε παραπάνω.

ΠΡΟΓΡΑΜΜΑ Κτήριο_Γραφείων

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: ΚΤ_ΓΡ[5, 12], ΑΘ_ΟΡ[5], i, j, sum, Min, Min_i, Min_j, Max, Max_i, Max_j

ΑΡΧΗ

Γέμισμα πίνακα
ανά γραμμή

```
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5
    ΓΡΑΨΕ 'Όροφος', i
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 12
        ΓΡΑΨΕ 'Δώσε υπαλλήλους γραφείου',j
        ΔΙΑΒΑΣΕ ΚΤ_ΓΡ[i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Υπολογισμός Μικρότερου και Μεγαλύτερου σε Δισδιάστατο
πίνακα

```
Min ← ΚΤ_ΓΡ[1, 1]
Min_i ← 1
Min_j ← 1
Max ← ΚΤ_ΓΡ [1, 1]
Max_i ← 1
Max_j ← 1
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 12
        ΑΝ ΚΤ_ΓΡ[i, j] < Min ΤΟΤΕ
            Min ← ΚΤ_ΓΡ[i, j]
            Min_i ← i
            Min_j ← j
        ΤΕΛΟΣ_ΑΝ
        ΑΝ ΚΤ_ΓΡ[i, j] > Max ΤΟΤΕ
            Max ← ΚΤ_ΓΡ [i, j]
            Max_i ← i
            Max_j ← j
        ΤΕΛΟΣ_ΑΝ
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Το γραφείο',Min_j, 'στον όροφο',Min_i, 'έχει τους λιγότερους υπαλλήλους'
ΓΡΑΨΕ 'Το γραφείο',Max_j, 'στον όροφο',Max_i, 'έχει τους περισσότερους υπαλλήλους'
```

Άθροισμα ανά γραμμή
& καταχώρηση
σε παράλληλο
ως προς τις γραμμές
πίνακα

```
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5
    sum ← 0
    ΓΙΑ j ΑΠΟ 1 ΜΕΧΡΙ 12
        sum ← sum + ΚΤ_ΓΡ[i, j]
    ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΡΑΨΕ 'Το σύνολο των υπαλλήλων του',i,'ορόφου είναι:',sum
    ΑΘΡ_ΟΡ[i] ← sum
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
```

Άθροισμα
μονοδιάστατου
πίνακα

```
sum ← 0
ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 5
    sum ← sum + ΑΘ_ΟΡ[i]
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ 'Το σύνολο όλων των υπαλλήλων, είναι:',sum
```

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Κτήριο_Γραφείων

4.10. Μη λυμένες ασκήσεις

1. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να διαβάζει τις ηλικίες και τα ονόματα 100 ανθρώπων. Τα δεδομένα να καταχωρούνται σε πίνακες, κάνοντας έλεγχο εγκυρότητας για την ηλικία που πρέπει να είναι θετικός αριθμός. Κατόπιν να υπολογίζει και να εμφανίζει:
 - 1) Το μέσο όρο των ηλικιών.
 - 2) Τη μέγιστη ηλικία καθώς πόσοι και ποιοί την έχουν.
 - 3) Το πλήθος και τα ονόματα των ανθρώπων που είναι άνω των 50 ετών.
2. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να διαβάζει για τους 176 υπαλλήλους μιας επιχείρησης τα ονόματα, το έτος πρόσληψης, τα έτη προϋπηρεσίας σε άλλες επιχειρήσεις και να αποθηκεύονται στους πίνακες ΟΝΟΜΑΤΑ, ΕΤΟΣ_ΠΡΟΣΛΗΨΗΣ και ΕΤΗ_ΥΠΗΡΕΣΙΑΣ αντίστοιχα.. Στη συνέχεια να:
 - 1) Εμφανίζει τα ονόματα των υπαλλήλων που θα συνταξιοδοτηθούν την επόμενη πενταετία καθώς και το πλήθος τους. Σύνταξη δίνεται στα 40 έτη συνολικής υπηρεσίας και άνω.
 - 2) Δημιουργηθεί πίνακας με το όνομα ΝΕΟΙ_ΕΡΓΑΖΟΜΕΝΟΙ, με τα ονόματα των εργαζομένων που έχουν συνολικά έτη υπηρεσίας (μαζί και η προϋπηρεσία τους) λιγότερα από 5 και στη συνέχεια να τα εμφανίζει.
3. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο τους αποθηκεύει σε μονοδιάστατο πίνακα. Στη συνέχεια να κάνει αντιστροφή των στοιχείων του πίνακα:
 - 1) Με τη χρήση 2^{ου} πίνακα
 - 2) Χωρίς τη χρήση 2^{ου} πίνακα.
4. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο διαβάζει την βαθμολογία ενός τμήματος 26 μαθητών στο μάθημα του ΑΕΠΠ και την αποθηκεύει σε μονοδιάστατο πίνακα. Στη συνέχεια να υπολογίζει και να εμφανίζει:
 - 1) Το μέσο όρο του τμήματος.
 - 2) Την απόκλιση από το μέσο όρο για κάθε μαθητή
 - 4) Το πλήθος των μαθητών με βαθμό μεγαλύτερο του μέσου όρου.
5. Στο κεντρικό υπολογιστή του τμήματος μηχανογράφησης ενός ασφαλιστικού ταμείου καταχωρούνται οι ασφαλισμένοι με όλα τα απαραίτητα στοιχεία τους. Για το σκοπό αυτό χρησιμοποιούνται διαφορετικοί πίνακες, μεταξύ των οποίων ο πίνακας ΟΝΟΜΑ και ο πίνακας ΕΤΟΣ, περιέχουν για κάθε ασφαλισμένο το όνομα και το έτος γέννησής του, αντίστοιχα. Γνωρίζοντας ότι κάποιος ασφαλισμένος συνταξιοδοτείται μόλις συμπληρώσει το 67ο έτος της ηλικίας του, να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να:
 - 1) Διαβάζει το τρέχον έτος
 - 2) Διαβάζει τα ονόματα και τις χρονολογίες γέννησης 100 ασφαλισμένων.
 - 3) Εμφανίζει τα ονόματα και το πλήθος εκείνων που θα συνταξιοδοτηθούν σε λιγότερο από 10 χρόνια.
6. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να διαβάζει 100 ακέραιους αριθμούς και να τους καταχωρεί στον πίνακα Α[100]. Στη συνέχεια να εμφανίζει όλες τις τριάδες αριθμών όπου ο μεσαίος αριθμός ισούται με το άθροισμα των άλλων δύο.
7. Σε μία πολιτιστική λέσχη τα μέλη της μπορούν να επιλέξουν για να παίξουν Σκάκι (Σ), Μπριτζ (Μ) ή Τάβλι (Τ). Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο διαβάζει τις προτιμήσεις 100 μελών του. Στη συνέχεια να υπολογίζει και να εμφανίζει ποιο παιχνίδι είναι η προτίμηση των περισσότερων. Να γίνεται έλεγχος εγκυρότητας των τιμών που δίνονται από το πληκτρολόγιο.
8. Μια τράπεζα διαχειρίζεται τους λογαριασμούς πελατών της χρησιμοποιώντας 2 πίνακες. Τον πίνακα ΟΝΟΜΑ που περιέχει τα ονοματεπώνυμα των πελατών της και τον πίνακα ΛΟΓΑΡΙΑΣΜΟΣ που περιέχει τα υπόλοιπα των λογαριασμών τους. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να:
 - 1) Διαβάζει τον αριθμό των πελατών της τράπεζας, ελέγχοντας ότι είναι θετικός αριθμός με ανώτατο όριο 10.000.000. Στη περίπτωση που δοθεί λάθος νούμερο, να τυπώνεται το μήνυμα " Μη απο-

δεκτός αριθμός... Ξαναπροσπαθήστε...." και να επαναλαμβάνεται η διαδικασία μέχρι αποδεκτής τιμής.

- 2) Διαβάζει τα Ονοματεπώνυμο και υπόλοιπο λογαριασμού για κάθε πελάτη της τράπεζας, καταχωρώντας τα αντίστοιχα στους πίνακες ΟΝΟΜΑ και ΛΟΓΑΡΙΑΣΜΟΣ.
 - 3) Διαβάζει το ονοματεπώνυμο, την προτίμηση ενός πελάτη για κατάθεση ή ανάληψη χρημάτων (K= κατάθεση, A= ανάληψη) καθώς και το ποσό. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να δοθεί ως ονοματεπώνυμο πελάτη η λέξη "ΤΕΛΟΣ".
 - 4) Για κάθε πελάτη που διαβάζει να κάνει τα εξής:
 - a. Να ελέγχει, στην περίπτωση ανάληψης, αν αυτή μπορεί να γίνει και να εμφανίζει το μήνυμα "Δεν επαρκεί το υπόλοιπό σας...." στην περίπτωση που δεν είναι εφικτή.
 - b. Σε κάθε περίπτωση (κατάθεση ή ανάληψη) να ενημερώνεται ο πίνακας με τον λογαριασμό του πελάτη ανάλογα σε κάθε περίπτωση.
 - c. Να εμφανίζει το νέο υπόλοιπο λογαριασμού.
 - 5) Να εμφανίζει το σύνολο των καταθέσεων που έγιναν.
 - 6) Να εμφανίζει το σύνολο των αναλήψεων που έγιναν.
9. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να δημιουργεί και να εμφανίζει τους παρακάτω πίνακες:

Πίνακας 1

1	6	11	16	21
2	7	12	17	22
3	8	13	18	23
4	9	14	19	24
5	10	15	20	25

Πίνακας 2

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

10. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να:
 - 1) Γεμίζει με ακέραιες τιμές από το πληκτρολόγιο ένα δισδιάστατο πίνακα 20x7
 - 2) Υπολογίζει και να τυπώνει τη γραμμή με το μεγαλύτερο άθροισμα.
 - 3) Υπολογίζει και να τυπώνει τη στήλη με το μικρότερο άθροισμα.
11. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να:
 - 1) Γεμίζει έναν πίνακα 5x5 με ακέραιους αριθμούς, που δίνονται από το πληκτρολόγιο.
 - 2) Υπολογίζει και να εμφανίζει:
 - i. Το μέγιστο στοιχείο της κύριας και της δευτερεύουσας διαγωνίου.
 - ii. Τον μέσο όρο των στοιχείων της κύριας και της δευτερεύουσας διαγωνίου.
12. Ένα σχολείο έχει 9 τμήματα με 26 μαθητές το κάθε τμήμα. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να:
 - 1) Καταχωρεί τους μέσους όρους των μαθητών ανά τμήμα στο πίνακα A[9,26].
 - 2) Υπολογίζει και να εμφανίζει το μέσο όρο κάθε τμήματος.
 - 3) Υπολογίζει και να εμφανίζει το πλήθος και το ποσοστό των αριστούχων ανά τμήμα. Αριστούχος θεωρείται αυτός που έχει μέσο όρο μεγαλύτερο από 18.5.
 - 4) Υπολογίζει και να εμφανίζει το μεγαλύτερο μέσο όρο σε όλο το σχολείο.
13. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο σε ένα μονοδιάστατο πίνακα ΒΑΘΜΟΙ[52] καταχωρεί τους βαθμούς του μαθήματος ΑΕΠΠ, των μαθητών της Γ' Λυκείου ενός σχολείου. Οι βαθμοί θεωρούνται θετικοί και ακέραιοι. Το πρόγραμμα υπολογίζει και τυπώνει τη συχνότητα που εμφανίζεται ο κάθε βαθμός, αν θεωρήσουμε ότι όλοι οι βαθμοί είναι από το 1 μέχρι το 20.
14. Ένα εμπορικό κατάστημα εμπορεύεται 25 μάρκες και από κάθε μάρκα 10 διαφορετικά είδη. Σε ένα δισδιάστατο πίνακα 25x10, το κατάστημα αποθηκεύει τις τιμές του κάθε είδους από την κάθε μάρκα. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο :

- 1) Να διαβάξει και να αποθηκεύει σ' ένα μονοδιάστατο πίνακα ΜΑΡΚΑ[10], τα ονόματα των μαρκών.
 - 2) Να διαβάξει και να αποθηκεύει τα ονόματα των ειδών που εμπορεύεται στον πίνακα ΕΙΔΗ[25].
 - 3) Να διαβάξει τη τιμή κάθε είδους ανά μάρκα και να τη καταχωρεί στον πίνακα ΤΙΜΗ[25,10].
 - 4) Να δίνεται από το πληκτρολόγιο ένα είδος και η μάρκα του και να τυπώνεται η αντίστοιχη τιμή του.
- 15.** Σ' ένα διαγωνισμό ταλέντων διαγωνίστηκαν 250 υποψήφιοι. Η βαθμολογία τους είναι στο διάστημα [1,100]. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο :
- 1) Να καταχωρεί σε πίνακες τα ονόματα και οι βαθμοί των υποψηφίων.
 - 2) Να εμφανίζει μια λίστα με τους 10 πρώτους, ταξινομημένους κατά φθίνουσα σειρά ως προς την βαθμολογία τους.
- 16.** Η Γ' τάξη έχει 4 τμήματα με 25 μαθητές η κάθε τάξη. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο :
- 1) Να διαβάξει τους βαθμούς ΑΕΠΠ των μαθητών και να τους καταχωρεί σ' ένα πίνακα Β[25,4].
 - 2) Να υπολογίζει και να εμφανίζει το Μέσο όρο ανά τμήμα.
 - 3) Να υπολογίζει και να εμφανίζει το τμήμα με τον μεγαλύτερο Μέσο όρο.
 - 4) Να εμφανίζει τον μεγαλύτερο Βαθμό και σε ποιο τμήμα βρίσκεται.
 - 5) Να υπολογίζει και να εμφανίζει το Μέσο Όρο του μαθήματος σε όλης την Γ' Λυκείου.
- 17.** Σε πίνακα ακεραίων ΛΟΤΤΟ[1000000,6] βρίσκονται οι 6 αριθμοί του ΛΟΤΤΟ που έχουν παίξει 1.000.000 παίκτες για το παιχνίδι του ΛΟΤΤΟ και στον πίνακα ΠΑΙΧΤΗΣ[1000000] είναι τα ονόματα των παικτών. Στο πίνακα ΚΕΡΔΙΣΕ[6] είναι οι 6 τυχεροί αριθμοί του ΛΟΤΤΟ που κληρώθηκαν. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο :
- 1) Να γεμίζει τους παραπάνω πίνακες με τιμές από το πληκτρολόγιο.
 - 2) Να κάνει την τελική διαλογή- πόσοι παίκτες είχαν 6 επιτυχίες, πόσοι είχαν 5 επιτυχίες, πόσοι είχαν 4 επιτυχίες, πόσοι είχαν 3 επιτυχίες, πόσοι είχαν 2 επιτυχίες και πόσοι είχαν 1 επιτυχία - και θα τις αποθηκεύει στον πίνακα ΔΙΑΛΟΓΗ[6].
 - 3) Να εμφανίζει τα ονόματα των παικτών που είχαν 6 επιτυχίες.
- 18.** Για τις ανάγκες μιας έρευνας δημιουργήθηκαν 3 μοναδιάστατοι πίνακες όπου ο 1ος περιέχει τι ονοματεπώνυμο, ο 2ος το φύλλο και ο 3ος τον μισθό για τους 245 υπαλλήλους μιας εταιρείας. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να υπολογίζει και να εμφανίζει:
- 1) Πόσοι είναι οι υπάλληλοι με μισθό μεγαλύτερο από 560€.
 - 2) Ποιο είναι το % ποσοστό των ανδρών με μισθό μεγαλύτερο του μέσου όρου του μισθού των ανδρών.
 - 3) Ποιες είναι οι 10 πιο καλοπληρωμένες γυναίκες.
- 19.** Στο αγώνισμα του άλματος εις μήκος συμμετέχουν 12 αθλητές οι οποίοι κάνουν 6 προσπάθειες ο καθένας. Να γραφεί πρόγραμμα σε "ΓΛΩΣΣΑ", το οποίο να:
- 1) Διαβάξει και να αποθηκεύει τα ονόματα 12 αθλητών σε μονοδιάστατο πίνακα.
 - 2) Διαβάξει και να αποθηκεύει σε διδιάστατο πίνακα τις επιδόσεις του κάθε αθλητή στις 6 προσπάθειες που έκανε. Η άκυρη προσπάθεια δίνεται το μηδέν.
 - 3) Εμφανίζει για κάθε αθλητή το όνομά του και τις επιδόσεις του ταξινομημένες, από την καλύτερη προς την χειρότερη, εκτός από τις άκυρες.
 - 4) Εμφανίζει το όνομα του νικητή του αγωνίσματος άλματος εις μήκος. Νικητής θεωρείται αυτός με την μεγαλύτερη επίδοση. Στην περίπτωση ισοβαθμίας, νικητής θεωρείται αυτός που έχει την μεγαλύτερη 2η επίδοση.

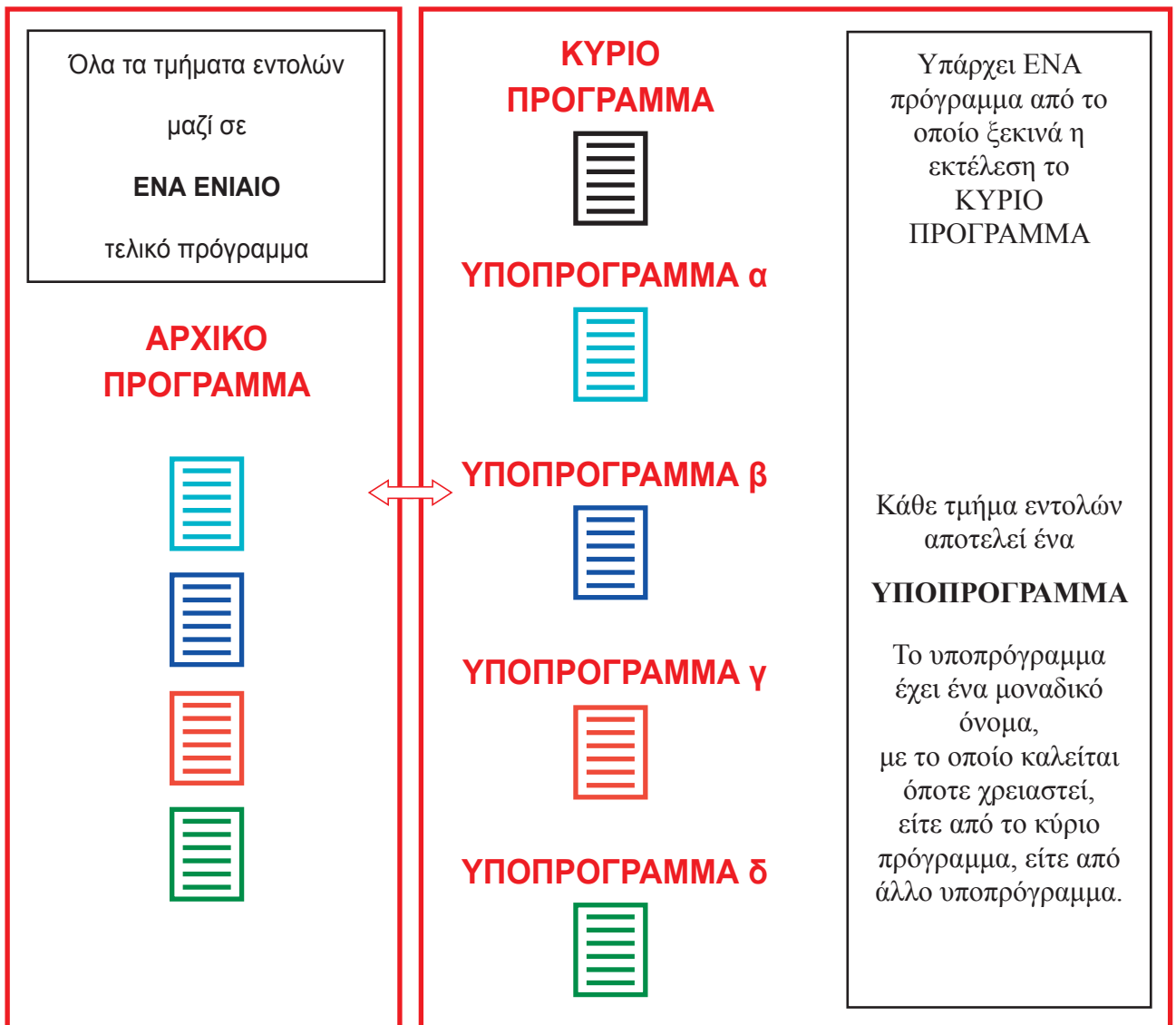
ΤΜΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ - ΥΠΟΠΡΟΓΡΑΜΜΑΤΑ (ΔΙΑΔΙΚΑΣΙΕΣ ΣΥΝΑΡΤΗΣΕΙΣ)

Περιεχόμενα

- Διαδικασίες
- Συναρτήσεις

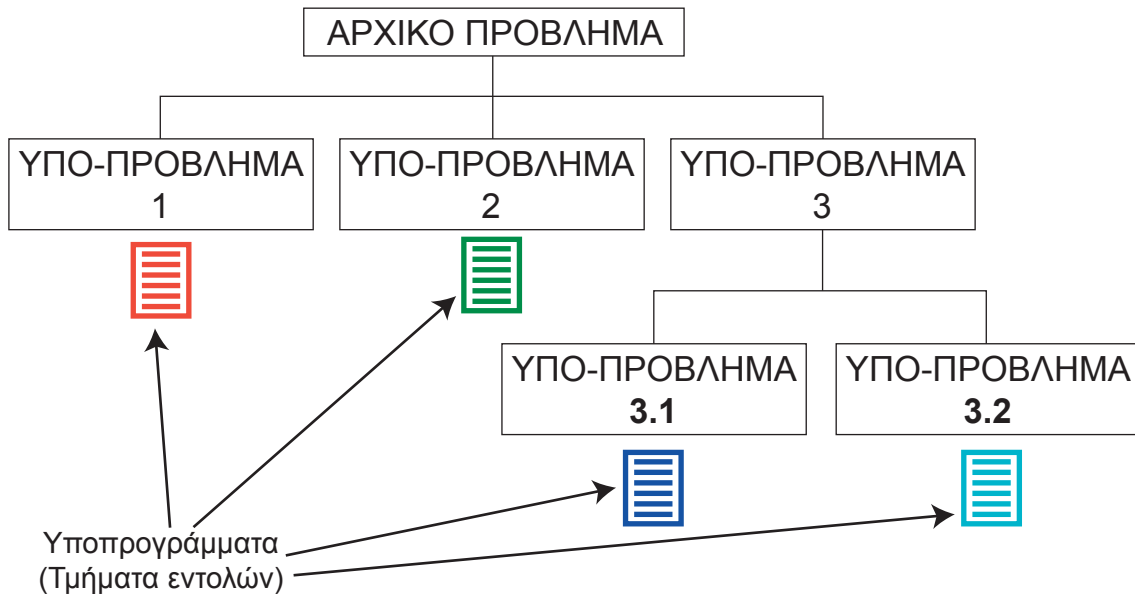
5. ΤΜΗΜΑΤΙΚΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ-ΥΠΟΠΡΟΓΡΑΜΜΑΤΑ (ΔΙΑΔΙΚΑΣΙΕΣ-ΣΥΝΑΡΤΗΣΕΙΣ)

Τμηματικός προγραμματισμός είναι η τεχνική σχεδίασης των προγραμμάτων κατά την οποία ένα πρόγραμμα αντικαθίσταται από ένα σύνολο απλούστερων μικρότερων και αυτόνομων τμημάτων προγραμμάτων. Και το αρχικό πρόγραμμα και το σύνολο των απλούστερων, μικρότερων και αυτόνομων τμημάτων προγραμμάτων όταν εκτελούνται, παράγουν ακριβώς τα ίδια αποτελέσματα. Στο σύνολο των απλούστερων μικρότερων και αυτόνομων τμημάτων προγραμμάτων, υπάρχει ένα από το οποίο ξεκινά η εκτέλεση των υποπρογραμμάτων, το οποίο καλείται ΚΥΡΙΟ ΠΡΟΓΡΑΜΜΑ. Τα άλλα απλούστερα μικρότερα και αυτόνομα τμήματα-προγράμματα ονομάζονται υποπρογράμματα.



5.1. Που χρησιμοποιούνται τα υποπρογράμματα;

Χρησιμοποιούνται στην περίπτωση όπου έχουμε να επιλύσουμε ένα σύνθετο και πολύπλοκο πρόβλημα με την χρήση Η/Υ. Το αντίστοιχο πρόγραμμα που πρέπει να δημιουργήσουμε είναι σύνθετο, πολύπλοκο και επομένως αρκετά δύσκολο. Ο καλύτερος τρόπος για την επίλυση του σύνθετου προβλήματος είναι να διαιρεθεί σε μικρότερα προβλήματα και αυτά σε άλλα μικρότερα κ.ο.κ. (ιεραρχική σχεδίαση) και να σχεδιάσουμε υποπρογράμματα (απλά μικρά προγράμματα) για τα μικρότερα προβλήματα. Έτσι είναι δυνατό να αντιμετωπισθεί πιο εύκολα η επίλυση του αρχικού σύνθετου προβλήματος.



Παράδειγμα 1: Ένα πρόγραμμα εύρεσης και εμφάνισης του μέσου όρου βαθμολογίας της τάξης, μπορεί να διαιρεθεί στο κύριο πρόγραμμα, που θα ξεκινάει την εκτέλεση των υποπρογραμμάτων:

1. Υποπρόγραμμα για την εισαγωγή των δεδομένων (βαθμών).
2. Υποπρόγραμμα για την επεξεργασία των δεδομένων (εύρεση του μέσου όρου της βαθμολογίας).
3. Υποπρόγραμμα για την εξαγωγή των αποτελεσμάτων (εμφάνιση του μέσου όρου).

5.2. Τι πρέπει να προσεχθεί;

1. Τα υποπρογράμματα χαρακτηρίζονται από τις παρακάτω τρεις ιδιότητες:
 - Κάθε υποπρόγραμμα έχει μια μόνο είσοδο και μια έξοδο
 - Κάθε υποπρόγραμμα πρέπει να είναι ανεξάρτητο από τα άλλα
 - Όταν σχεδιάζεται ένα υποπρόγραμμα πρέπει να είναι ανεξάρτητο από άλλα υποπρογράμματα, ώστε να συντηρείται αυτόνομα.
 - Το υποπρόγραμμα είναι αυτόνομο και ανεξάρτητο, δηλώνονται όλες οι μεταβλητές που χρησιμοποιεί στις εντολές του και στο κύριο σώμα του γράφονται οι εντολές του.
 - Κάθε υποπρόγραμμα πρέπει να μην είναι μεγάλο. Πρέπει να εκτελεί μόνο μια λειτουργία έτσι, ώστε να είναι εύκολα κατανοητό και να ελέγχεται.

2. Παράμετροι

- Ένα πρόγραμμα ενεργοποιείται (δηλ. εκτελεί τις εντολές του), όταν καλεστεί από άλλο υποπρόγραμμα ή πρόγραμμα (κύριο πρόγραμμα).
- Για να επικοινωνεί το κύριο πρόγραμμα με το υπόλοιπο πρόγραμμα χρησιμοποιεί μεταβλητές που μεταφέρουν τιμές μεταξύ υποπρογράμματος και κύριου προγράμματος, δηλ. χρησιμοποιούνται σαν κανάλια επικοινωνίας, και λέγονται παράμετροι. Δηλαδή παράμετρος είναι μια μεταβλητή που επιτρέπει τη μεταφορά τιμών από ένα υποπρόγραμμα (τυπικές παράμετροι ή ορίσματα), στο κύριο πρόγραμμα (πραγματικές παράμετροι ή παράμετροι) ή σε άλλο υποπρόγραμμα. Δεν έχουν σημασία τα ονόματα των πραγματικών και τυπικών παραμέτρων (μπορεί και να είναι διαφορετικά), παρά μόνο το πλήθος τους, η σειρά τους (η αντιστοιχία τους γίνεται με την σειρά που είναι γραμμένες) και να είναι του ίδιου τύπου.

3. Μεταβλητές

- Το κύριο πρόγραμμα και τα υποπρογράμματα χρησιμοποιούν και μεταβλητές εκτός από τις παραμέτρους. Μετά το τέλος του υποπρογράμματος ο χώρος μνήμης που είχε δεσμευθεί για τις δηλωθείσες μεταβλητές του, στο τμήμα δηλώσεών του, αποδεσμεύεται.

4. Διαφορά μεταξύ παραμέτρων και μεταβλητών

- Η παράμετροι χρησιμοποιούνται για μεταφορά τιμών από τα υποπρογράμματα στο κύριο πρόγραμμα και αντίστροφα.
- Η μεταβλητές χρησιμοποιούνται για καταχώρηση τιμών ξεχωριστά για το κύριο πρόγραμμα και τα υποπρογράμματα.

5. Είδη υποπρογραμμάτων

Υπάρχουν δύο είδη υποπρογραμμάτων

- Διαδικασία
- Συνάρτηση

5.3. Διαδικασίες (Παράδειγμα διαδικασίας)

Εκφώνηση προβλήματος

Να σχεδιασθεί κύριο πρόγραμμα και διαδικασία, έτσι ώστε όταν εκτελείται το κύριο πρόγραμμα (και με κλήση της αντίστοιχης διαδικασίας), θα ανταλλάσσει τις τιμές μεταξύ δύο μεταβλητών, για δύο διαφορετικά ζεύγη τιμών που θα δίνονται από το πληκτρολόγιο. Συγκεκριμένα θα εισάγονται σε δύο μεταβλητές δύο τιμές από το πληκτρολόγιο και το πρόγραμμα θα ανταλλάσσει τις τιμές τους. Στη συνέχεια θα εισάγονται για δύο άλλες μεταβλητές δύο νέες τιμές από το πληκτρολόγιο και το πρόγραμμα θα ανταλλάσσει πάλι τις τιμές τους.

Ανάλυση προβλήματος

Σύμφωνα με το Παράδειγμα 1 (σελίδα 2), πρέπει να αναλυθεί το πρόβλημα σε υποπροβλήματα. Ποια όμως είναι τα υποπροβλήματα στο δοθέν πρόβλημα;

Η εμπειρία, που θα αποκτηθεί μετά την ολοκλήρωση αυτής της ενότητας βοηθά στον επιτυχή χωρισμό ενός προβλήματος σε υποπροβλήματα.

Για διευκόλυνση της κατανόησης της έννοιας των διαδικασιών (υποπρογραμμάτων) για αρχή, ακολουθείται η παρακάτω τεχνική, βασιζόμενοι σε πράγματα που έχουν ήδη διδαχθεί.

Χωρίς διαδικασίες η λύση του προβλήματος θα ήταν:

ΠΡΟΓΡΑΜΜΑ Ανταλλαγή

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: α, β, ι, κ, temp

ΑΡΧΗ

ΓΡΑΨΕ 'Δώσε τιμές για α και β'

ΔΙΑΒΑΣΕ α, β

temp <-- α

α <-- β

β <-- temp

ΓΡΑΨΕ 'α=', α, 'β=', β

ΓΡΑΨΕ 'Δώσε τιμές για ι και κ'

ΔΙΑΒΑΣΕ ι, κ

temp <-- ι

ι <-- κ

κ <-- temp

ΓΡΑΨΕ 'ι=', ι, 'κ=', κ

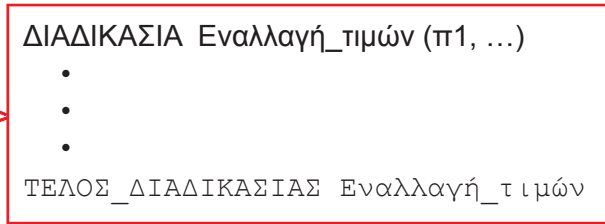
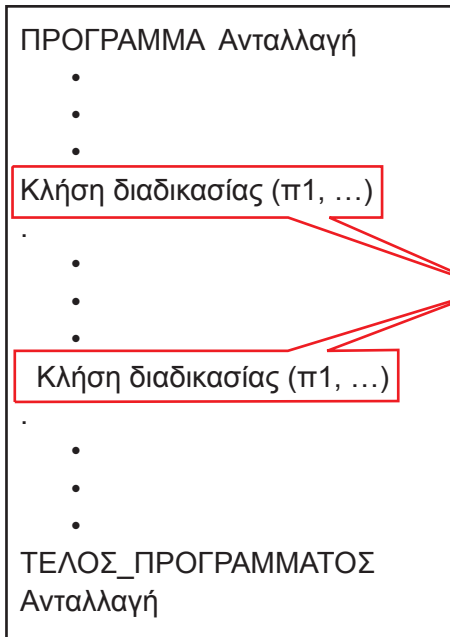
ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ανταλλαγή

Τμήμα προγράμματος που μπορεί να εκτελεστεί αυτοτελώς.

Μάλιστα επαναλαμβάνεται δύο φορές σε διαφορετικά σημεία στο πρόγραμμα.

Στην περίπτωση χρήσης διαδικασιών και κύριου προγράμματος ποια θα μπορούσε να είναι η λύση;

Παρατηρείται ότι στο πρόγραμμα του παραδείγματος χωρίς διαδικασίες, υπάρχει ένα τμήμα προγράμματος που μπορεί να εκτελεστεί αυτοτελώς και μάλιστα αυτό το τμήμα επαναλαμβάνεται δύο φορές. Η σκέψη που δημιουργείται, για την επίλυση το ίδιου προβλήματος με διαδικασίες, είναι να σχεδιασθεί ένα κύριο πρόγραμμα (Ανταλλαγή) που θα χρησιμοποιεί κατάλληλα και όπου χρειάζεται, μια διαδικασία (Εναλλαγή_τιμών), η οποία θα υλοποιεί ακριβώς το παραπάνω επαναλαμβανόμενο τμήμα προγράμματος. Η λύση σχηματικά θα μπορούσε να αναπαρασταθεί (Οπτικοποίηση της λύσης), ως εξής;



Λύση

<pre> ΠΡΟΓΡΑΜΜΑ Ανταλλαγή ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ:α, β, ι, κ ΑΡΧΗ ΔΙΑΒΑΣΕ α, β ΚΑΛΕΣΕ Εναλλαγή_τιμών(α β) ΓΡΑΨΕ α, β ΔΙΑΒΑΣΕ ι, κ ΚΑΛΕΣΕ Εναλλαγή_τιμών(ι,κ) ΓΡΑΨΕ ι, κ ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ανταλλαγή </pre>	<p>Κύριο πρόγραμμα, που εκτελεί δύο φορές κατά σειρά τις εξής λειτουργίες:</p> <ul style="list-style-type: none"> • Διαβάζει σε δύο μεταβλητές, δύο ακέραιους αριθμούς. • Εναλλάσσει τις τιμές μεταξύ των μεταβλητών (διαδοχική κλήση διαδικασίας). • Εμφανίζει το περιεχόμενο των δύο μεταβλητών.
--	---

<pre> ΔΙΑΔΙΚΑΣΙΑ Εναλλαγή_τιμών(κ, λ) ΜΕΤΑΒΛΗΤΕΣ ΑΚΕΡΑΙΕΣ:κ, λ,temp ΑΡΧΗ temp <- κ κ <- λ λ <- temp ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ Εναλλαγή_τιμών </pre>	<p>Διαδικασία που εναλλάσσει τις τιμές δύο μεταβλητών (κ,λ) μέσω της βοηθητικής μεταβλητής temp.</p>
--	--

Ανάλυση της λύσης του προβλήματος

5.3.1 Πώς συντάσσεται (ορίζεται) μια διαδικασία;

Σύνταξη:

ΔΙΑΔΙΚΑΣΙΑ <όνομα> (<λίστα παραμέτρων>)

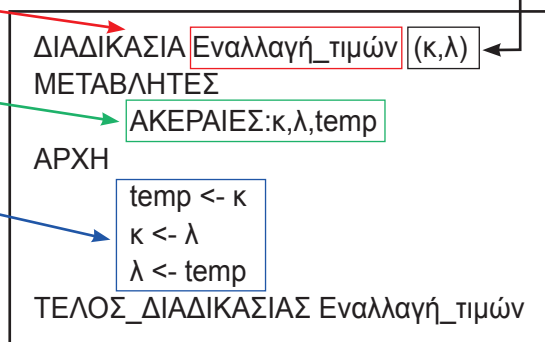
ΜΕΤΑΒΛΗΤΕΣ

<δήλωση μεταβλητών>

ΑΡΧΗ

<εντολές>

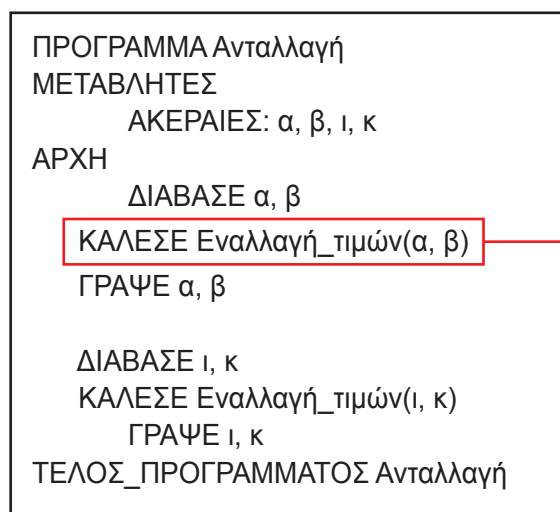
ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ



5.3.2. Πώς το κύριο πρόγραμμα καλεί μια διαδικασία;

Πάντοτε έχουμε μόνο ένα ΚΥΡΙΟ ΠΡΟΓΡΑΜΜΑ ενώ μπορούμε να έχουμε όσες ΔΙΑΔΙΑΚΑΣΙΕΣ χρειαζόμαστε

Μια διαδικασία καλείται από το κύριο πρόγραμμα ή άλλη διαδικασία με την εντολή ΚΑΛΕΣΕ, που έχει ως όρισμα το όνομα της καλούμενης διαδικασίας και τις αντίστοιχες πραγματικές παραμέτρους.



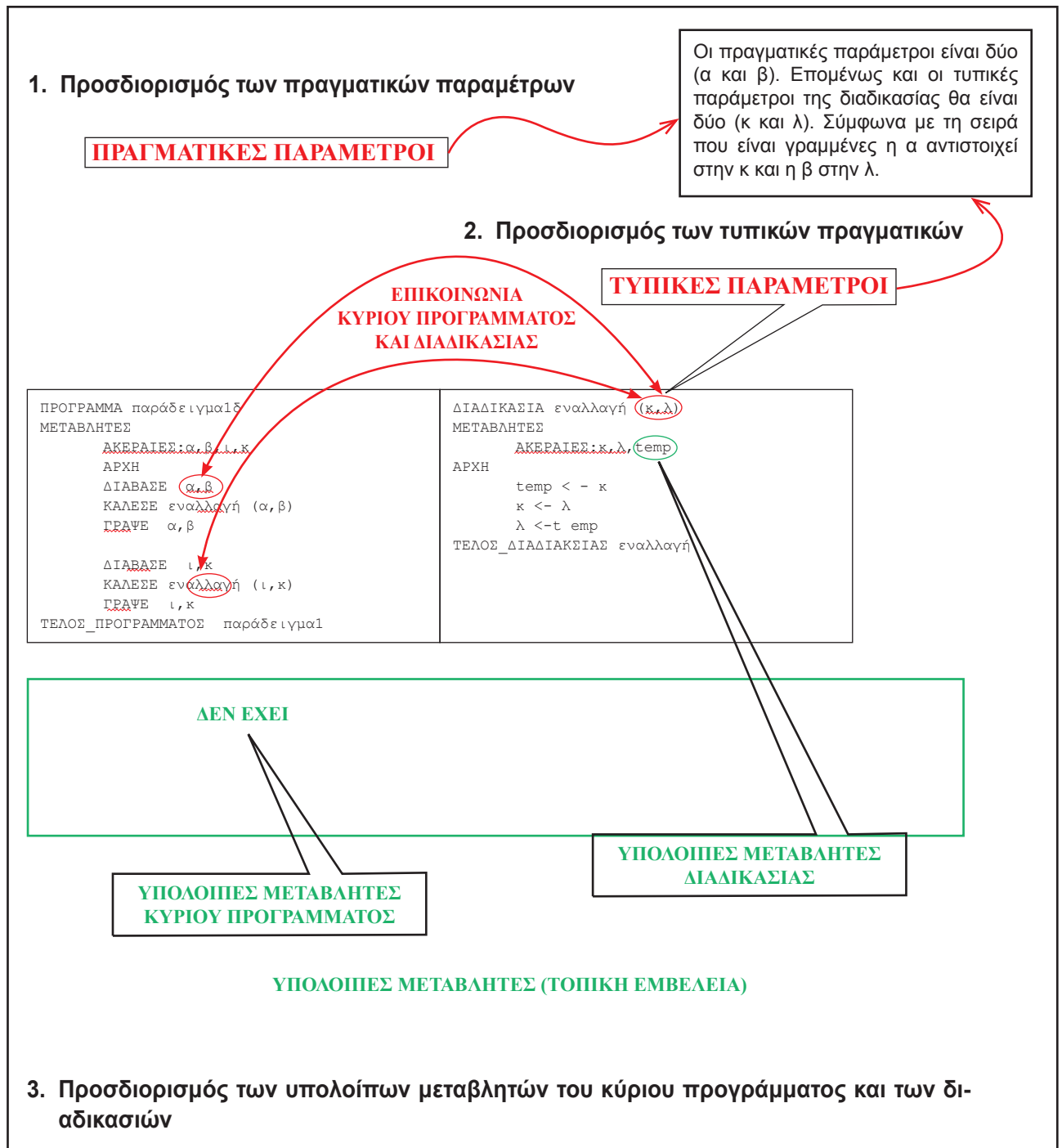
Κλήση της διαδικασίας Εναλλαγή_τιμών, με δύο πραγματικές παραμέτρους την α και την β

5.3.3. Πώς εκτελείται το πρόγραμμα και πώς λειτουργεί η επικοινωνία του κύριου προγράμματος με τη διαδικασία;

Για την κατανόηση της λειτουργίας θα εκτελεστεί το πρόγραμμα για με δεδομένα εισόδου 3, 7, 2, 9 και θα αναλυθούν η επικοινωνία του κύριου προγράμματος και της διαδικασίας με:

1. Προσδιορισμό των πραγματικών παραμέτρων
2. Προσδιορισμό των τυπικών πραγματικών
3. Προσδιορισμό των υπολοίπων μεταβλητών του κύριου προγράμματος και των διαδικασιών

4. Συμπλήρωση των πινάκων τιμών όλων των μεταβλητών (πραγματικές παράμετροι, τυπικές παράμετροι και υπόλοιπες μεταβλητές) του κύριου προγράμματος και των διαδικασιών με συμπλήρωση των τιμών τους για κάθε εντολή που εκτελείται, του κύριου προγράμματος και των διαδικασιών.



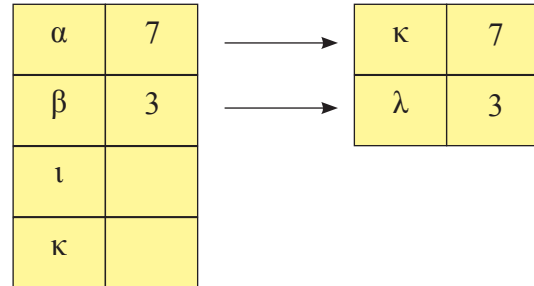
4. Εκτέλεση του προγράμματος και συμπλήρωση των πινάκων τιμών όλων των μεταβλητών (πραγματικές παράμετροι, τυπικές παράμετροι και υπόλοιπες μεταβλητές) του κύριου προγράμματος και των διαδικασιών με συμπλήρωση των τιμών τους για κάθε εντολή που εκτελείται, του κύριου προγράμματος και των διαδικασιών και επηρεάζουν τις τιμές των μεταβλητών.

- Η εκτέλεση του προγράμματος ξεκινά πάντα από το κύριο πρόγραμμα Ανταλλαγή και εκτελούνται οι εντολές του κυρίου προγράμματος (εκτελείται δηλαδή η πρώτη εντολή του κυρίου προγράμματος, η **ΔΙΑΒΑΣΕ**) μέχρι την εντολή **ΚΑΛΕΣΕ Εναλλαγή_τιμών**. Η εκτέλεση της εντολής αυτής προκαλεί "πάγωμα" του κυρίου προγράμματος. Η συνέχεια της εκτέλεσης του προγράμματος μεταφέρεται στη διαδικασία, **Εναλλαγή_τιμών** που καλείται. Η τιμές των μεταβλητών και παραμέτρων πριν την πρώτη κλήση της διαδικασίας, φαίνεται στο παρακάτω σχήμα. Κατά την κλήση της διαδικασίας, γίνεται αντιγραφή (μεταβίβαση) των τιμών των πραγματικών παραμέτρων στις αντίστοιχες τυπικές

ΠΕΡΑΣΜΑ ΠΑΡΑΜΕΤΡΩΝ ΚΑΤΑ ΤΗΝ ΚΛΗΣΗ ΔΙΑΔΙΚΑΣΙΩΝ

Πρώτη κλήση διαδικασίας

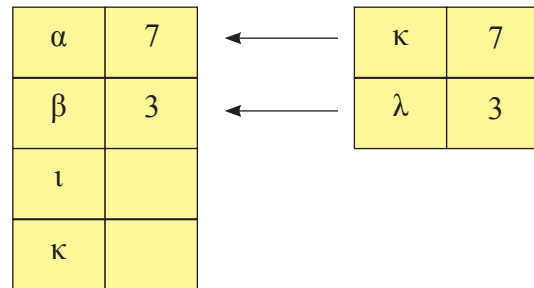
α	7
β	3
ι	
κ	



Κατάσταση πριν την πρώτη κλήση

Μεταβίβαση των μεταβλητών α και β στις κ και λ αντίστοιχα

κ	9
λ	2

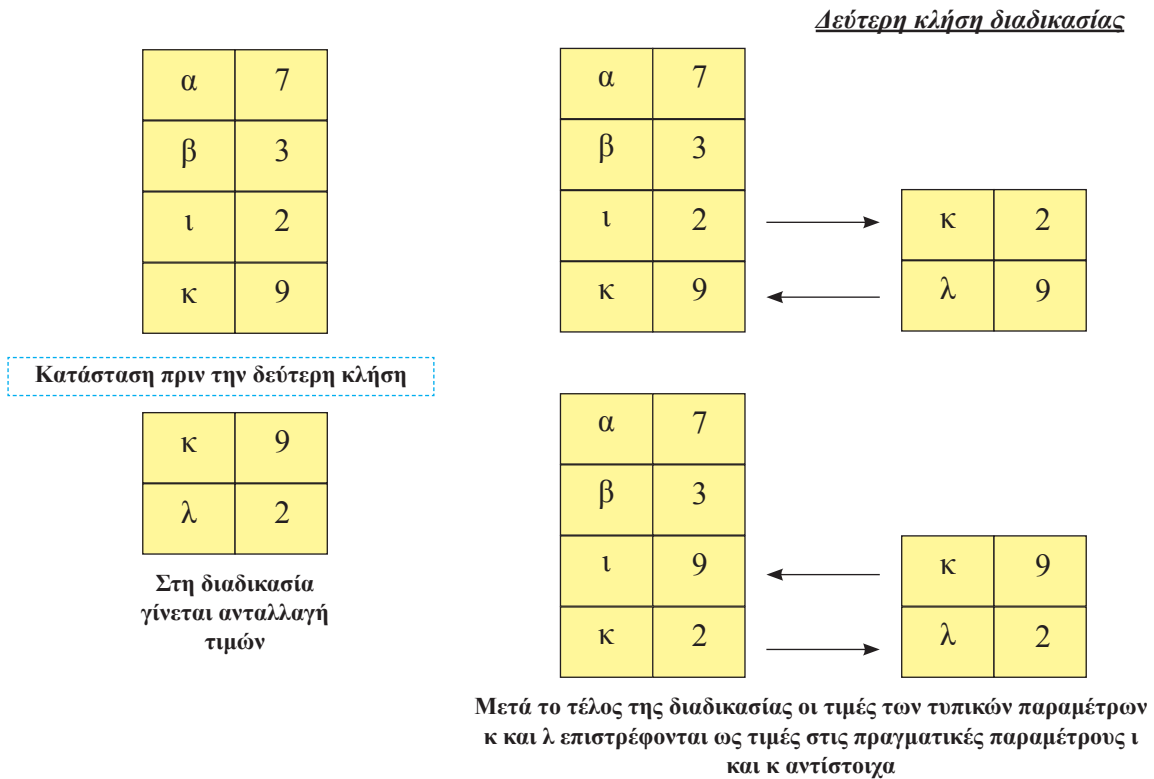


Στη διαδικασία
γίνεται ανταλλαγή
τιμών

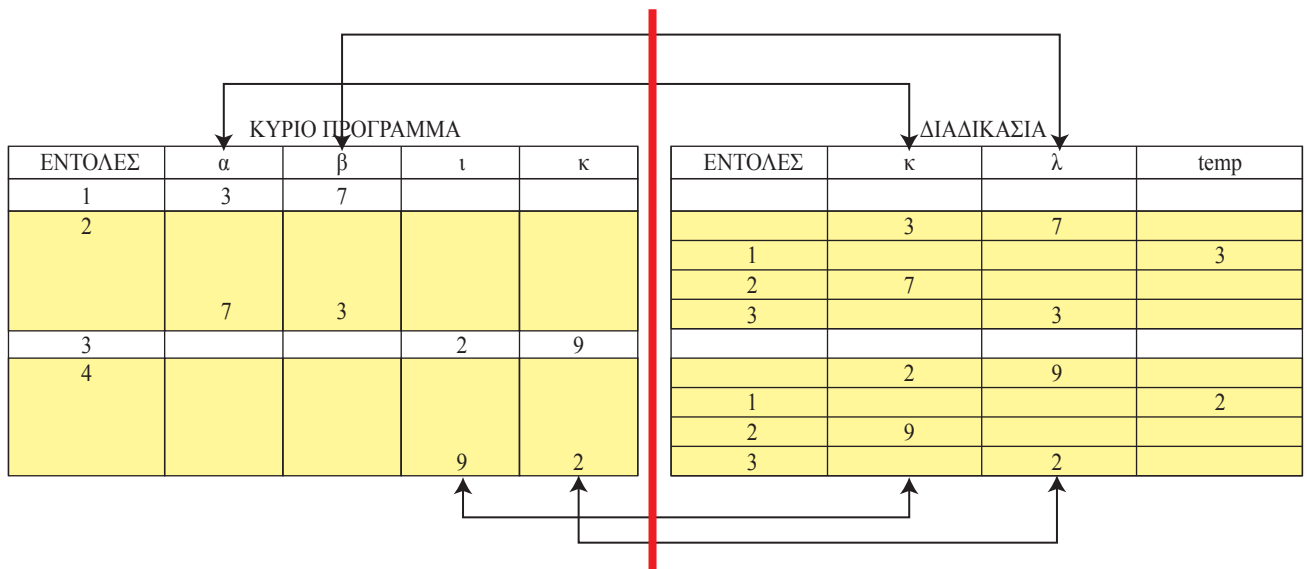
Μετά το τέλος της διαδικασίας οι τιμές των τυπικών παραμέτρων κ και λ επιστρέφονται ως τιμές στις πραγματικές παραμέτρους α και β αντίστοιχα

- Οι μεταβλητές του κυρίου προγράμματος, είναι γνωστές μόνο στο πρόγραμμα. Έξω από το πρόγραμμα, κανένα υποπρόγραμμα - διαδικασία δεν τις γνωρίζει και δεν μπορεί να τις χρησιμοποιήσει. Το ίδιο ισχύει και για τις μεταβλητές των υποπρογραμμάτων διαδικασιών. Για αυτό λέμε, ότι οι μεταβλητές έχουν ισχύ τοπική (εντός του προγράμματος-υποπρογράμματος που δηλώνονται). ο Η/Υ διαθέτει διαφορετικό χώρο μνήμης για τις μεταβλητές της διαδικασίας από αυτές του κυρίου προγράμματος)

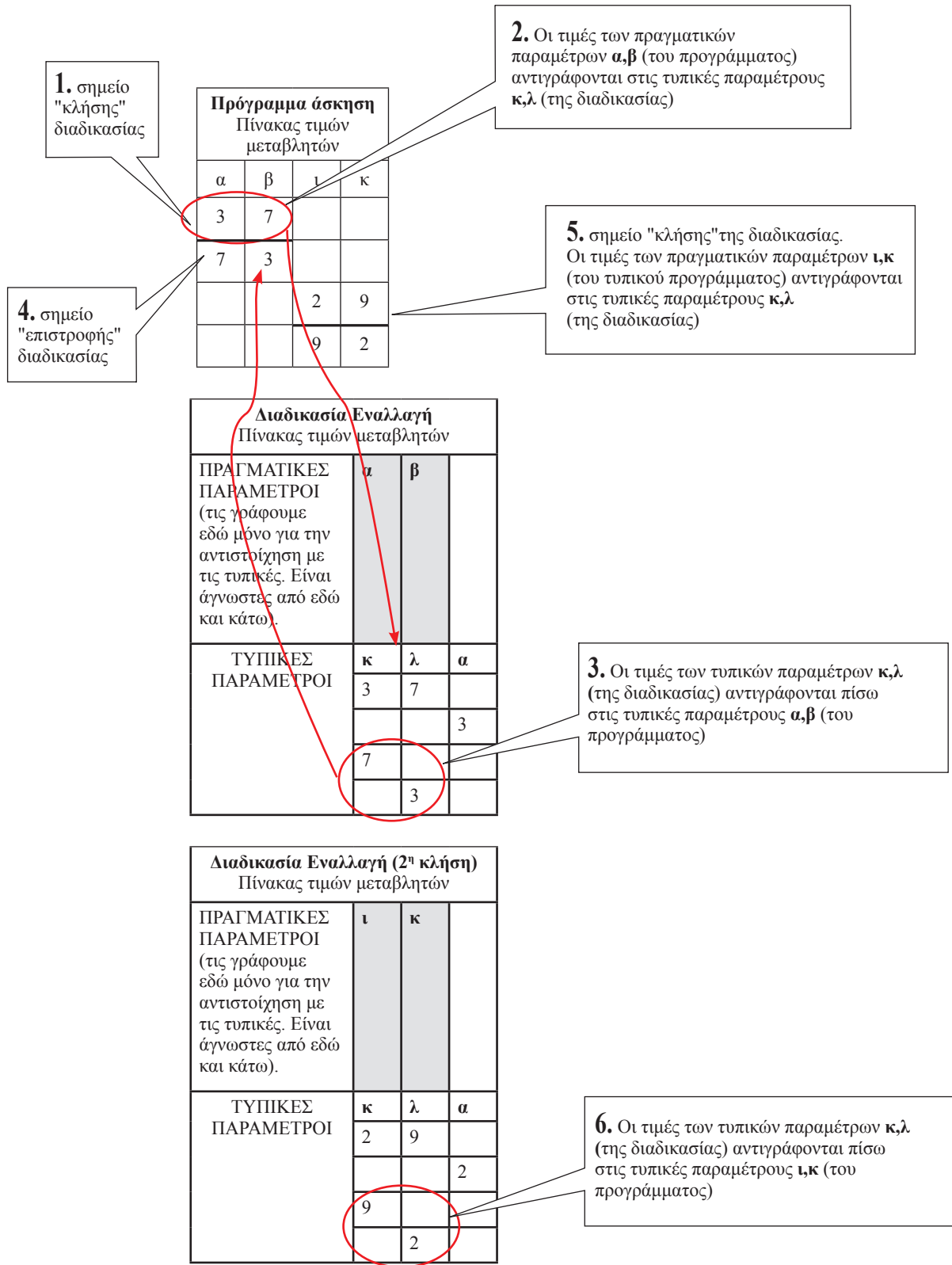
- Να σημειωθεί ιδιαίτερα ότι μετά το τέλος της διαδικασίας οι τιμές των τυπικών παραμέτρων της διαδικασίας επιστρέφονται ως τιμές στις πραγματικές παραμέτρους του κυρίου προγράμματος (όπως φαίνεται στο προηγούμενο σχήμα) και ο έλεγχος εκτέλεσης του προγράμματος επανέρχεται στο κύριο πρόγραμμα και συνεχίζει η εκτέλεση από την επόμενη εντολή, από την εντολή που κάλεσε την διαδικασία (σημείο κλήσης της διαδικασίας. Μέχρι εδώ έχουν ανταλλαγεί οι τιμές των μεταβλητών α και β. Στη συνέχεια εκτελείται η εντολή ΓΡΑΨΕ. Και τυπώνονται οι τιμές των α, και β. Εκτελείται το δεύτερο ΔΙΑΒΑΣΕ, του κυρίου προγράμματος και φθάνουμε στην εντολή κλήσης της διαδικασίας για δεύτερη φορά, και οι τιμές των μεταβλητών και παραμέτρων διαμορφώνονται όπως ακριβώς δείχνει το παρακάτω σχήμα.



Ο συγκεντρωτικός πίνακας τιμών για το κύριο πρόγραμμα και τις κλήσεις των διαδικασιών φαίνεται στο παρακάτω σχήμα.



Ακολουθώντας τα αριθμημένα βήματα στο παρακάτω σχήμα, γίνεται καλύτερα αντιληπτή η εκτέλεση του προγράμματος.



Στο παρακάτω σχήμα φαίνεται τι θα φανεί στην οθόνη του Η/Υ.

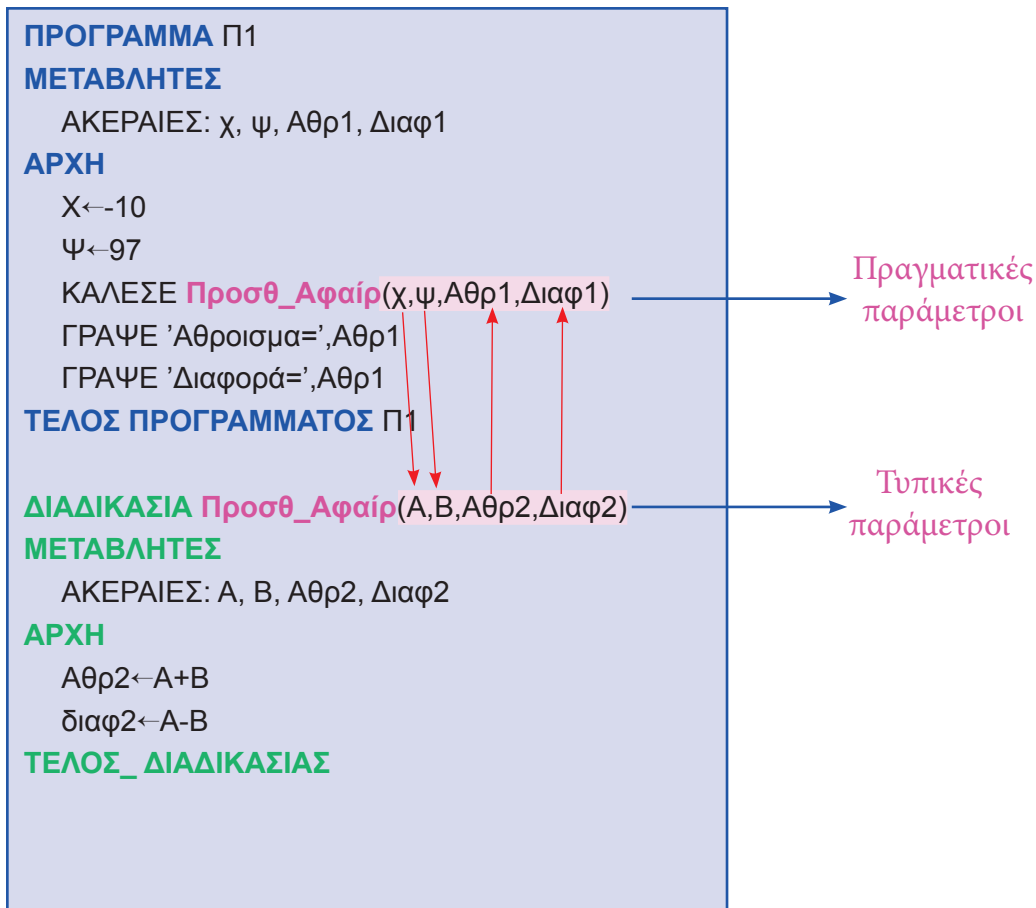
ΜΟΝΑΔΑ ΕΞΟΔΟΥ-ΟΘΟΝΗ

7 3
9 2

5.3.3.1. Σημεία στα οποία πρέπει να δοθεί ιδιαίτερη σημασία.

- **Τοπικότητα μεταβλητών.** Οι μεταβλητές του κυρίου προγράμματος, είναι γνωστές μόνο στο πρόγραμμα. Έξω από το πρόγραμμα, κανένα υποπρόγραμμα - διαδικασία δεν τις γνωρίζει και δεν μπορεί να τις χρησιμοποιήσει. Το ίδιο ισχύει και για τις μεταβλητές των υποπρογραμμάτων διαδικασιών. Για αυτό λέμε, ότι οι μεταβλητές έχουν ισχύ τοπική (εντός του προγράμματος-υποπρογράμματος που δηλώνονται).
- **Επικοινωνία υποπρογραμμάτων.** Ο μοναδικός τρόπος επικοινωνίας είναι μέσω των παραμέτρων (πραγματικών-τυπικών). Οι παράμετροι είναι μεταβλητές του κυρίου προγράμματος και των υποπρογραμμάτων.
- **Αντιστοιχία παραμέτρων-κανόνες τυπικών και πραγματικών παραμέτρων.**
 1. Ο αριθμός των πραγματικών και τυπικών παραμέτρων πρέπει να είναι ο ίδιος.
 2. Κάθε πραγματική παράμετρος αντιστοιχεί στην τυπική παράμετρο που βρίσκεται στην αντίστοιχη θέση.
 3. Η τυπική παράμετρος και η αντίστοιχη πραγματική παράμετρος πρέπει να είναι του ίδιου τύπου.Θα μπορούσαμε να το παρομοιάσουμε με ένα βύσμα (π.χ. οθόνης που συνδέεται στην κατάλληλη θύρα του υπολογιστή).
- **Χωροθέτηση.** Πρώτα πάντα είναι γραμμένο το κύριο πρόγραμμα. Έπειτα ακολουθούν τα υποπρογράμματα.
- **Εκτέλεση.** Η εκτέλεση ξεκινά πάντα από το κύριο πρόγραμμα, και προφανώς τελειώνει στο κύριο πρόγραμμα.

Τα σημεία προσοχής που αναφέρονται φαίνονται στο παρακάτω παράδειγμα το οποίο υπολογίζεται και τυπώνεται το άθροισμα και η διαφορά δύο δοθέντων αριθμών με διαδικασία που υπολογίζει το άθροισμα και τη διαφορά τους.

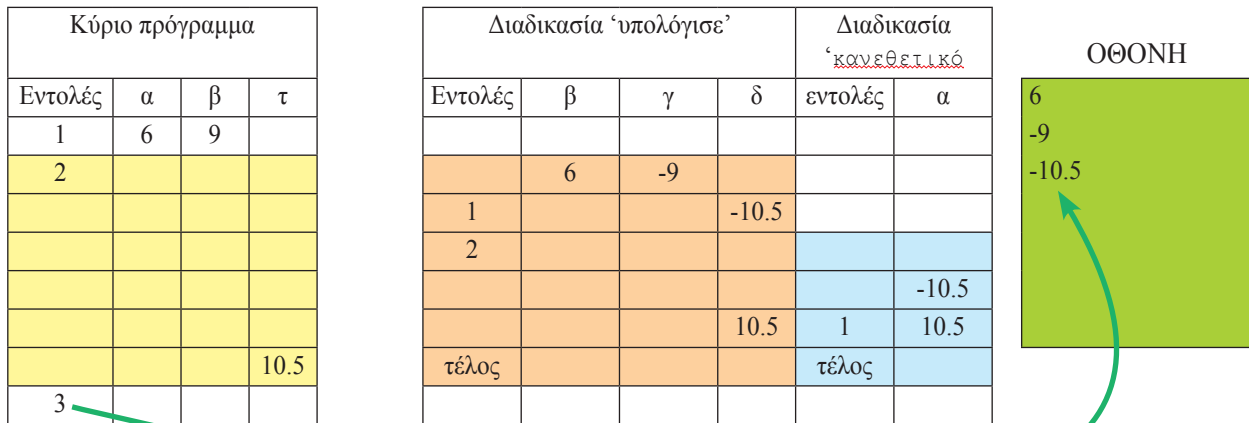


Σύμφωνα με το Πρόγραμμα Π1 του παραδείγματος:

- Οι τιμές των μεταβλητών χ, ψ, Αθρ1 και Διαφ1 δίνονται στις μεταβλητές Α, Β, Αθρ2 και Διαφ2 του υποπρογράμματος Προσθ_Αφαίρ, κατά την κλήση του.
- Έτσι: η μεταβλητή Α παίρνει την τιμή -10, η μεταβλητή Β παίρνει τη τιμή 97 ενώ οι μεταβλητές Αθρ2 και Διαφ2 δεν παίρνουν καμία τιμή αφού η Αθρ1 και η Διαφ1 δεν έχουν τιμές.
- Μετά την εκτέλεση των εντολών της Διαδικασίας, όταν εκτελεστεί η εντολή ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ, οι τυπικές παράμετροι της διαδικασίας δίνουν τις τιμές που έχουν στις αντίστοιχες πραγματικές παραμέτρους. Έτσι :
- Έτσι:
 Η μεταβλητή χ παίρνει τη τιμή της Α(=-10)
 Η μεταβλητή ψ παίρνει τη τιμή της Β(=97)
 Η μεταβλητή Αθρ1 παίρνει τη τιμή της Αθρ2(=87)
 Η μεταβλητή Διαφ1 παίρνει τη τιμή της Διαφ2(=-107)
- Με την επιστροφή στο κύριο πρόγραμμα όλες οι θέσεις μνήμης που είχαν δοθεί στη διαδικασία απελευθερώνονται

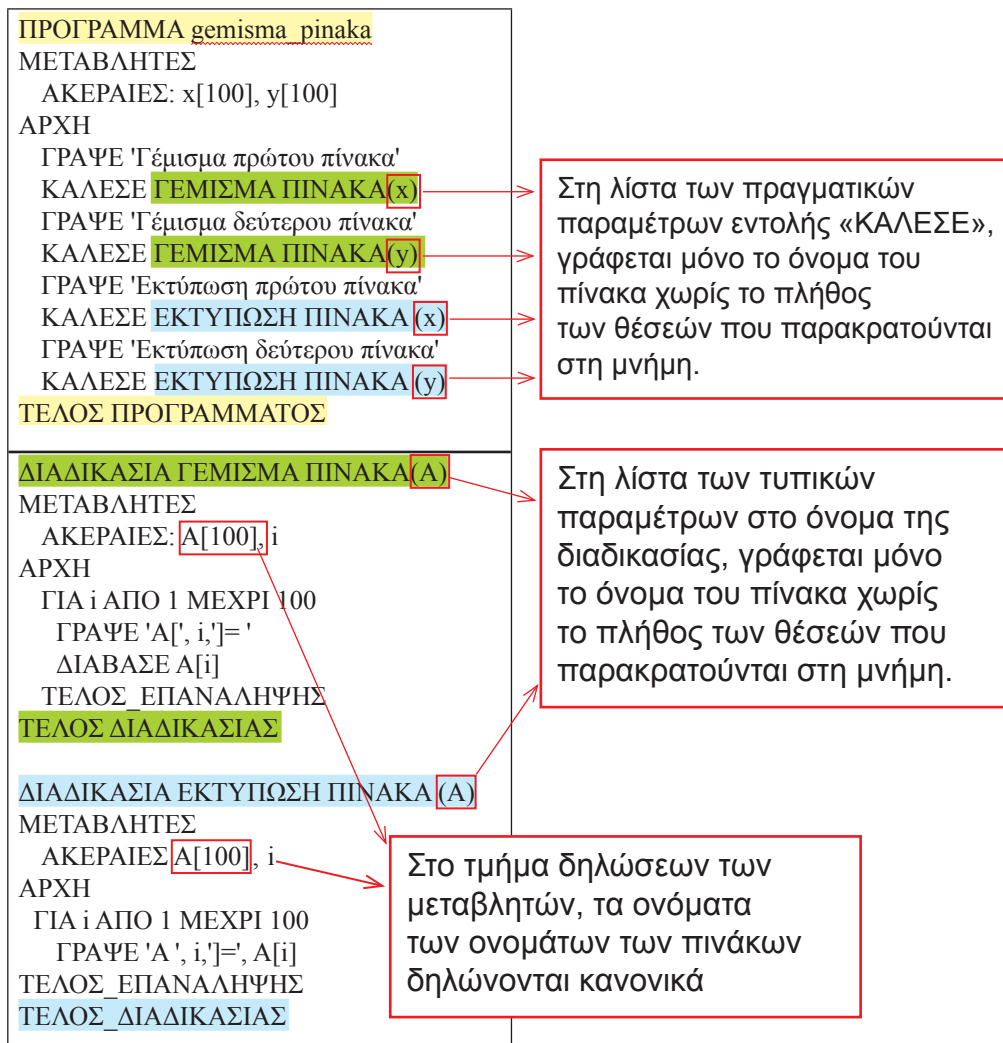
<p>ΔΙΑΔΙΚΑΣΙΑ υπολόγισε(β, γ, δ)</p> <p>ΜΕΤΑΒΛΗΤΕΣ</p> <p>ΑΚΕΡΑΙΕΣ: β, γ</p> <p>ΠΡΑΓΜΑΤΙΚΕΣ: δ</p> <p>ΑΡΧΗ</p> <p>$\delta \leftarrow (\beta + 3 * \gamma) / 2$</p> <p>ΚΑΛΕΣΕ κανεΘετικό(δ)</p> <p>ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ υπολόγισε</p>	<p>Η διαδικασία «υπολογισε», υπολογίζει την τιμή μιας παράστασης.</p> <p>Αυτή η διαδικασία καλεί τη διαδικασία «κανεΘετικο», που της μεταβιβάζει την τιμή της παράστασης που υπολόγισε.</p> <p>Μετά την επιστροφή της απόλυτης τιμής της παράστασης, από τη διαδικασία «κανεΘετικο», την επιστρέφει στο κύριο πρόγραμμα.</p>
<p>ΔΙΑΔΙΚΑΣΙΑ κανεΘετικό(α)</p> <p>ΜΕΤΑΒΛΗΤΕΣ</p> <p>ΠΡΑΓΜΑΤΙΚΕΣ: α</p> <p>ΑΡΧΗ</p> <p>ΑΝ $\alpha < 0$ ΤΟΤΕ $\alpha \leftarrow (-1) * \alpha$</p> <p>ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ κανεΘετικο</p>	<p>Η διαδικασία «κανεΘετικο» υπολογίζει την απόλυτη τιμή της παράστασης και την επιστρέφει στην διαδικασία «υπολογισε».</p>

Ο πίνακας τιμών του παραδείγματος φαίνεται στο παρακάτω σχήμα.



5.3.5. Διαδικασίες και Πίνακες

Στο παρακάτω παράδειγμα φαίνεται το γέμισμα και η εκτύπωση δύο πινάκων με κλήση των αντίστοιχων διαδικασιών ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ(x) και ΕΚΤΥΠΩΣΗ_ΠΙΝΑΚΑ(x)



Προσοχή: στη σύνταξη της εντολής «ΚΑΛΕΣΕ», σε σχέση με τα ονόματα των παραμέτρων (πραγματική και τυπική), όταν αφορά πίνακα. Στη λίστα των παραμέτρων γράφεται μόνο το όνομα του πίνακα χωρίς το πλήθος των θέσεων που παρακρατούνται στη μνήμη. Π.χ.

ΚΑΛΕΣΕ ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ(χ) και **όχι ΚΑΛΕΣΕ ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ(χ[100])**

Ο ίδιος κανόνας χρησιμοποιείται και στην τυπική παράμετρο των διαδικασιών

Π.χ. ΔΙΑΔΙΚΑΣΙΑ ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ(χ) και

όχι ΔΙΑΔΙΚΑΣΙΑ ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ(χ[100]).

Στο τμήμα δηλώσεων των μεταβλητών, τα ονόματα των ονομάτων των πινάκων δηλώνονται κανονικά Π.χ.

ΑΚΕΡΑΙΕΣ:Α[100]

Στο παρακάτω παράδειγμα ζητείται να γραφεί διαδικασία που να επιστρέφει τον μικρότερο αριθμό και τη θέση του από ένα πίνακα χ[100] με ακέραιους αριθμούς.

Ανάλυση

- Η διαδικασία θα έχει τρεις παραμέτρους, τον πίνακα χ σαν δεδομένο, τον Min που θα επιστρέφει τον μικρότερο αριθμό και τον Min_θέση που θα επιστρέφει τη θέση της μικρότερης τιμής.
- Θα αναπτυχθεί ο αλγόριθμος του μικρότερου που θα κρατά και την αντίστοιχη θέση του και τις τιμές αυτές θα τις καταχωρεί στις αντίστοιχες παραμέτρους για να μεταφέρουν τα αποτελέσματα στο υποπρόγραμμα που καλεί τη διαδικασία.

Λύση

ΔΙΑΔΙΚΑΣΙΑ Μικρότερο (χ,Min, Min_θέση)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: χ[100], i,Min,Min_θέση

ΑΡΧΗ

Min ← χ[1]

Min_θέση ← 1

ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 100

ΑΝ χ[i]< Min ΤΟΤΕ

Min ← χ[i]

Min_θέση←i

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

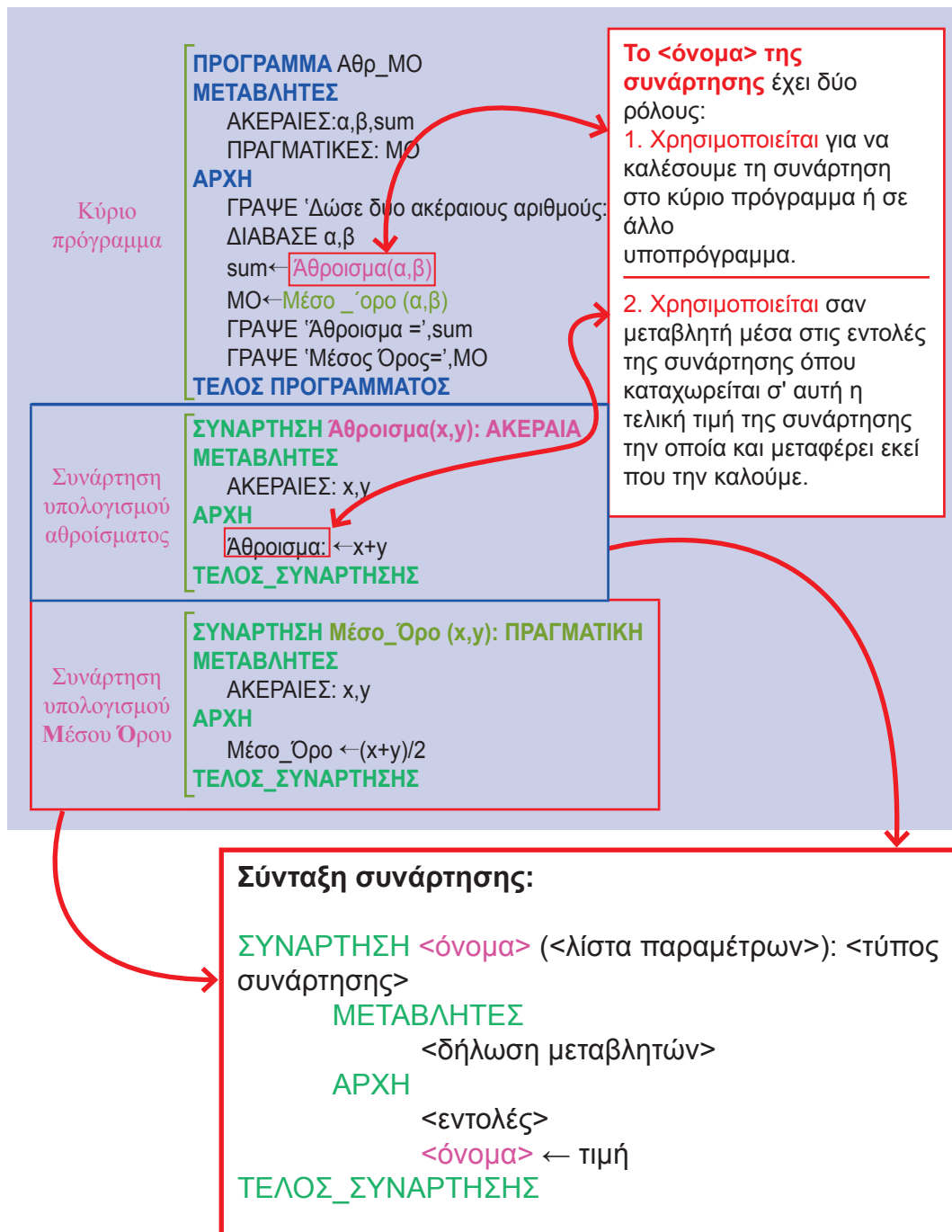
ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

5.4. Συναρτήσεις (Παράδειγμα συνάρτησης)

Να γραφτεί πρόγραμμα που να διαβάζει δύο ακέραιους αριθμούς και να τυπώνει το άθροισμα τους και το μέσο όρο τους.

Ο υπολογισμός του αθροίσματος και του μέσου όρου γίνεται από δύο συναρτήσεις.

Λύση



5.4.1. Πώς συντάσσεται και πώς καλείται μια συνάρτηση;

5.4.1.1. Σύνταξη - ορισμός συνάρτησης

ΣΥΝΑΡΤΗΣΗ <όνομα> (<λίστα παραμέτρων>): <τύπος συνάρτησης>

ΜΕΤΑΒΛΗΤΕΣ

<δήλωση μεταβλητών>

ΑΡΧΗ

<εντολές>

<όνομα> ← τιμή

ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ

Ο τύπος της συνάρτησης, που δηλώνεται στον ορισμό της (<τύπος της συνάρτησης>) μετά το όνομα της συνάρτησης, εκφράζει τον τύπο της τιμής που μεταφέρει η συνάρτηση.

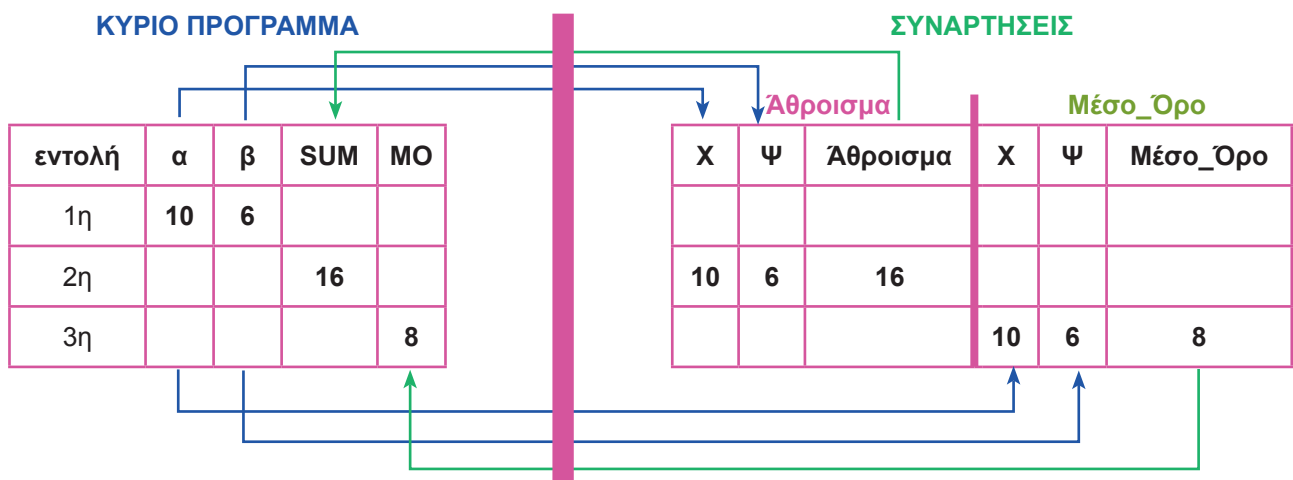
Αν δηλαδή η τιμή που παίρνει είναι ακέραια, πραγματική, λογική, ανεξάρτητα από τον τύπο των δικών της ορισμάτων. Π. χ. η συνάρτηση «αθροισμα» οι τιμές που παίρνει είναι ακέραιες (η συνάρτησης πρέπει να είναι ακέραιου τύπου). Οι παράμετροί της (χ,ψ) είναι ακέραιες. Αν οι παράμετροί της ήταν πραγματικού τύπου και η συνάρτηση υπολόγιζε το ακέραιο μέρος του αθροίσματός τους, πάλι η συνάρτηση, αφού παίρνει ακέραιες τιμές θα πρέπει να είναι ακεραίου τύπου και όχι πραγματικού τύπου που είναι οι παράμετροί της.

5.4.1.2. Κλήση συνάρτησης

- Η συνάρτηση καλείται από το κύριο πρόγραμμα με το όνομά της, μέσα σε μια εντολή. Π.χ ως τιμή μιας μεταβλητής σε μια εντολή εκχώρησης: $\text{sum} \leftarrow \text{Αθροισμα}(α,β)$
- Το <όνομα> της συνάρτησης έχει δύο ρόλους:
 - Χρησιμοποιείται για να καλέσουμε τη συνάρτηση στο κύριο πρόγραμμα ή σε άλλο υποπρόγραμμα.
 - Χρησιμοποιείται σαν μεταβλητή μέσα στις εντολές της συνάρτησης όπου καταχωρείται σ' αυτή η τελική τιμή της συνάρτησης την οποία και μεταφέρει εκεί που την καλούμε.
- Οι παράμετροι στη συνάρτηση δέχονται τιμές και δεν τις μεταφέρουν από τη συνάρτηση σε άλλο πρόγραμμα ή υποπρόγραμμα.
- Οι τιμές που επιστρέφουν οι συναρτήσεις μπορεί να είναι ΠΡΑΓΜΑΤΙΚΕΣ, ΑΚΕΡΑΙΕΣ, ΧΑΡΑΚΤΗΡΕΣ, ΛΟΓΙΚΕΣ.

Επεξήγηση της λύσης του παραπάνω παραδείγματος:

- Ο πίνακας τιμών κατά την εκτέλεση του προγράμματος είναι:



Παρατηρούμε, ότι:

- Το κάθε υποπρόγραμμα, εδώ η κάθε συνάρτηση, δουλεύει αυτόνομα στη μνήμη στη δική του περιοχή μνήμης και έχει επαφή με το πρόγραμμα μόνο μέσω των παραμέτρων.
- Όταν τελειώνει η εκτέλεση των εντολών της ΣΥΝΑΡΤΗΣΗΣ τότε όλες οι μεταβλητές της χάνονται και ελευθερώνεται η αντίστοιχη περιοχή μνήμης.
- Οι μεταβλητές α και β δίνουν τη τιμή τους στις μεταβλητές x και y αντίστοιχα όταν καλείται η συνάρτηση άθροισμα, ενώ δίνουν τη τιμή τους αντίστοιχα στις x και y όταν καλείται η συνάρτηση Μέσος Όρος.
- Η μεταβλητή Άθροισμα δίνει τη τιμή της στη μεταβλητή sum του κύριου προγράμματος όταν τελειώσει η συνάρτηση Άθροισμα και ομοίως η μεταβλητή Μέσο_Όρο στην MO όταν τελειώσει η συνάρτηση Μέσο_Όρο.

5.4.2. Σε τι διαφέρουν οι διαδικασίες από τις συναρτήσεις;

Διαδικασίες

Οι διαδικασίες μπορούν να εκτελέσουν μια οποιαδήποτε λειτουργία, π.χ. να εισάγουν δεδομένα, να εκτελούν υπολογισμούς, να τυπώνουν αποτελέσματα και να αλλάζουν τις τιμές των μεταβλητών

Οι διαδικασίες μεταφέρουν τα αποτελέσματα στα άλλα υποπρογράμματα με τη χρήση παραμέτρων

Οι διαδικασίες εκτελούνται αν γράψουμε την εντολή ΚΑΛΕΣΕ και μετά το όνομα της διαδικασίας.

Συναρτήσεις

Οι συναρτήσεις υπολογίζουν μόνο μια τιμή και μόνο αυτή επιστρέφουν στο κύριο πρόγραμμα ή στο υποπρόγραμμα που τις κάλεσε. Η τιμή αυτή μπορεί να είναι ΑΡΙΘΜΗΤΙΚΗ, ΧΑΡΑΚΤΗΡΑΣ ή ΛΟΓΙΚΗ.

Οι συναρτήσεις μεταφέρουν το αποτέλεσμα στο κύριο πρόγραμμα ή στο υποπρόγραμμα που τις κάλεσε με το όνομά τους και όχι με τη χρήση παραμέτρων. Μοιάζουν με τις μαθηματικές συναρτήσεις.

Οι συναρτήσεις εκτελούνται με τη χρήση του ονόματος τους μέσα σε οποιαδήποτε εντολή.

Μια συνάρτηση μπορεί να αντικατασταθεί με αντίστοιχη διαδικασία

5.4.3. Διαδικασίες και Συναρτήσεις Μαζί

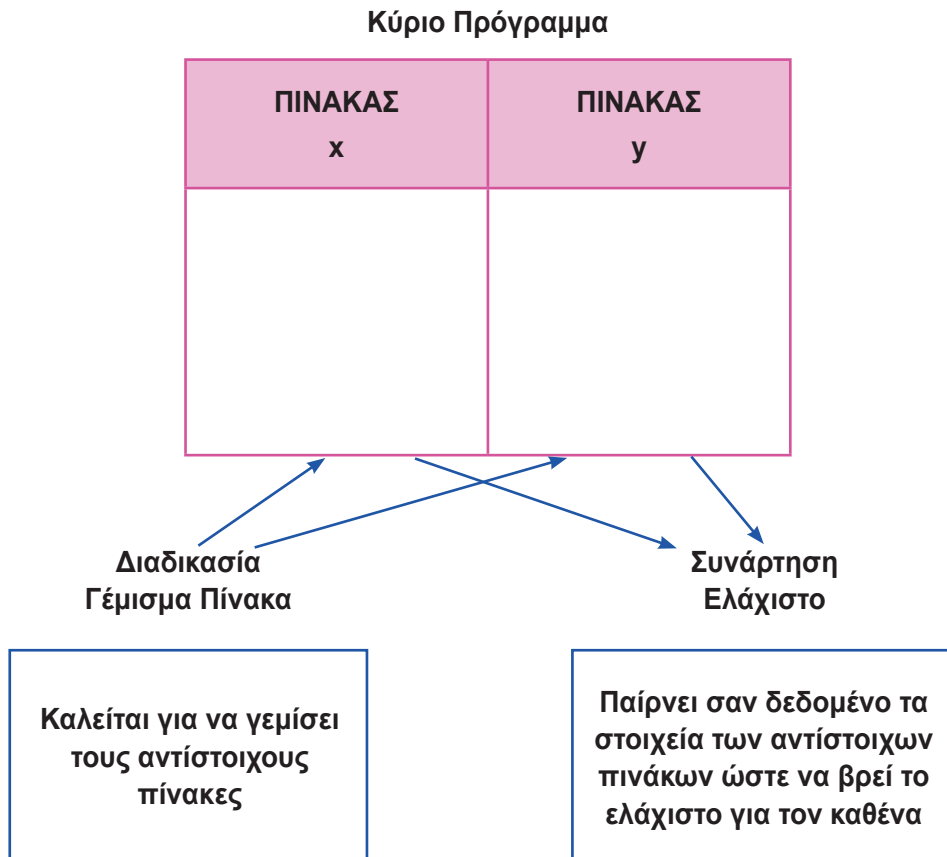
Ένα πρόγραμμα είναι δυνατόν να περιλαμβάνει και διαδικασίες και συναρτήσεις. Π.Χ.

Να γραφτεί πρόγραμμα που να διαβάσει δύο πίνακες $x[100]$ και $y[100]$ με ακέραιους αριθμούς και στη συνέχεια να τυπώνει τον ελάχιστο για κάθε πίνακα.

Το γέμισμα των πινάκων να γίνει με χρήση της διαδικασίας ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ και η εύρεση του μικρότερου με χρήση της συνάρτησης ΕΛΑΧΙΣΤΟ.

Ανάλυση προβλήματος

Αυτό που ζητείται σχηματικά αναπαρίσταται με το παρακάτω σχήμα.



1. Πρέπει να γεμίσουν τους δύο πίνακες με κάλεσμα της διαδικασίας ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ (x) και ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ (y). Καλείται η ίδια διαδικασία για να γεμίσουν δύο πίνακες. Αυτό γίνεται με αλλαγή της παραμέτρου, όπου την κάθε φορά παίρνει το όνομα του πίνακα που αφορά. Αυτό είναι και το βασικό πλεονέκτημα των υποπρογραμμάτων, ότι μπορούν να χρησιμοποιηθούν όσες φορές χρειάζεται, χρησιμοποιώντας κάθε φορά τις κατάλληλες παραμέτρους.

2. Επίσης θα πρέπει να κληθεί δυο φορές η συνάρτηση ΕΛΑΧΙΣΤΟ (x) και ΕΛΑΧΙΣΤΟ (y), με τη λογική που αναφέρθηκε παραπάνω. Επειδή η συνάρτηση καλείται μέσα από εντολή ή θα χρησιμοποιηθεί η εντολή ΓΡΑΨΕ οπότε και τυπώνεται απευθείας ή θα χρησιμοποιηθεί η εντολή καταχώρησης για να αποθηκευθεί τη τιμή της σε μια μεταβλητή και στη συνέχεια να τυπωθεί η μεταβλητή.
3. Μετά το τέλος του προγράμματος γράφονται τα υποπρογράμματα. Η Διαδικασία ΓΕΜΙΣΜΑ_ΠΙΝΑΚΑ και η Συνάρτηση ΕΛΑΧΙΣΤΟ.

Ο κώδικας του προγράμματος είναι:

ΠΡΟΓΡΑΜΜΑ Ελάχιστα_Πινάκων

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: x[100], y[100]

ΑΡΧΗ

ΓΡΑΨΕ 'Καταχώρηση δεδομένων 1ου πίνακα'

ΚΑΛΕΣΕ Γέμισμα_Πίνακα (x)

ΓΡΑΨΕ 'Καταχώρηση δεδομένων 2ου πίνακα'

ΚΑΛΕΣΕ Γέμισμα_Πίνακα (y)

ΓΡΑΨΕ 'Ελάχιστος αριθμός του 1ου πίνακα:', Ελάχιστο (x)

ΓΡΑΨΕ 'Ελάχιστος αριθμός του 2ου πίνακα:', Ελάχιστο (y)

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ Ελάχιστα_Πινάκων

ΔΙΑΔΙΚΑΣΙΑ Γέμισμα_Πίνακα (A)

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A[100], i

ΑΡΧΗ

ΓΙΑ i ΑΠΟ 1 ΜΕΧΡΙ 100

ΓΡΑΨΕ 'Δώσε ', i, ' στοιχείο'

ΔΙΑΒΑΣΕ A[i]

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΤΕΛΟΣ_ΔΙΑΔΙΚΑΣΙΑΣ

ΣΥΝΑΡΤΗΣΗ Ελάχιστο (A): ΑΚΕΡΑΙΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: Min, i, A[100]

ΑΡΧΗ

Min ← A[1]

ΓΙΑ i ΑΠΟ 2 ΜΕΧΡΙ 100

ΑΝ A[i] < Min ΤΟΤΕ

Min ← A[i]

ΤΕΛΟΣ_ΑΝ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Ελάχιστο ← Min

ΤΕΛΟΣ_ΣΥΝΑΡΤΗΣΗΣ

5.4.4. Μη λυμένες ασκήσεις

1. Να γραφεί συνάρτηση η οποία να επιστρέφει την τιμή ΑΛΗΘΗΣ, αν ο αριθμός είναι πολλαπλάσιο του 2 και στην αντίθετη περίπτωση, να επιστρέφει την τιμή ΨΕΥΔΗΣ.
2. Να γραφεί διαδικασία η οποία να διαβάζει ακέραιους αριθμούς και να επιστρέφει το πλήθος των θετικών, το πλήθος των αρνητικών και το πλήθος των μηδέν. Το πλήθος των ακεραίων που θα διαβαστούν θα ορίζεται σαν παράμετρος της διαδικασίας.
3. Να γραφεί υποπρόγραμμα το οποίο να αναζητεί ένα ακέραιο αριθμό σ' ένα πίνακα χ[1000] ακεραίων αριθμών. Όταν τον βρίσκει να επιστρέφει τη θέση που βρέθηκε, διαφορετικά να επιστρέφει μηδέν. Ο πίνακας και ο ακέραιος αριθμός να ορίζονται ως παράμετροι του υποπρογράμματος. (οι αριθμοί στον πίνακα είναι μοναδικοί)
4. Να γραφεί υποπρόγραμμα το οποίο να ταξινομεί τον πίνακα χ[100] με ονόματα, αλφαβητικά. Στην αντιμετάθεση των τιμών να καλείται το υποπρόγραμμα ΑΝΤΙΜΕΤΑΘΕΣΗ (x,y), όπου και θα αναπτυχθεί κατάλληλα. Ο πίνακας χ να είναι παράμετρος του υποπρογράμματος.
5. Να γραφτεί υποπρόγραμμα, που να εντοπίζει τους δύο μικρότερους αριθμούς από ένα πίνακα Χ[50] με ακέραιους αριθμούς και να τους επιστρέφει.
6. Να γραφεί συνάρτηση που να υπολογίζει την παρακάτω μαθηματική παράσταση:

$$g(x) = x^2 + x^3 - \frac{\sqrt{2 + x^3}}{x + 3 * \sqrt{x}}$$

7. Να γραφεί υποπρόγραμμα, το οποίο να ελέγχει αν ο αριθμός που πληκτρολογείται είναι θετικός και να τον επιστρέφει στο υποπρόγραμμα που το καλεί. Σε περίπτωση που ο αριθμός δεν είναι θετικός να τυπώνεται μήνυμα λάθους και να επαναλαμβάνεται η πληκτρολόγηση μέχρι να πληκτρολογηθεί ο σωστός αριθμός.
8. Να γραφεί υποπρόγραμμα που να ελέγχει αν ένας πίνακας Α[100] με ακέραιους αριθμούς είναι ταξινομημένος κατ' αύξουσα τάξη ή όχι.
9. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ» το οποίο να:
 - 1) Καταχωρεί στους πίνακες Ημέρα[500] και Μήνα[500] την ημέρα και τον μήνα αντίστοιχα μιας δαπάνης, στον πίνακα Περιγραφή[500] την περιγραφή της δαπάνης και στον πίνακα Κόστος[500] το συνολικό ποσό της αντίστοιχης δαπάνης.
 - 2) Ταξινομεί τους πίνακες πρώτα ως προς το μήνα κατά αύξουσα τάξη και στην περίπτωση ίδιου μήνα η ταξινόμηση να γίνει ως προς την ημέρα, κατά αύξουσα τάξη.
 - 3) Δίνονται από το πληκτρολόγιο 2 ημερομηνίες (εκφράζουν μια χρονική περίοδο) να εμφανίζεται η περιγραφή κάθε δαπάνης με τα αντίστοιχα ποσά δαπάνης της χρονικής περιόδου που πληκτρολογήσαμε. Επίσης στο τέλος να εμφανίζεται το συνολικό ποσό που έχει ξοδευτεί την αντίστοιχη χρονική περίοδο.
 - 4) Να γίνεται χρήση του υποπρογράμματος ΕΛΕΓΧΟΣ_ΗΜΕΡΟΜΗΝΙΑΣ, που να ελέγχει ότι η 1^η ημ/νια είναι μικρότερη από την 2^η ημ/νια. Στην αντίθετη περίπτωση να τις αντιμεταθέτει. Να χρησιμοποιηθεί η διαδικασία ΑΝΤΙΜΕΤΑΘΕΣΗ όπου χρειάζεται.
10. Να γραφεί πρόγραμμα σε «ΓΛΩΣΣΑ» το οποίο να διαχειρίζεται πίνακα Τηλεφωνικού Καταλόγου 500 γραμμών, με 1^η στήλη το Ονοματεπώνυμο, 2^η στήλη την Διεύθυνση και 3^η στήλη το Τηλέφωνο. Η διαχείριση γίνεται βάση του παρακάτω Μενού επιλογών:
 1. Εισαγωγή νέου συνδρομητή
 2. Διαγραφή συνδρομητή
 3. Διόρθωση δεδομένων συνδρομητή
 4. Αναζήτηση συνδρομητή (Με Όνομα ή Τηλέφωνο)
 5. Εμφάνιση Τηλεφωνικού Καταλόγου
 6. ΈξοδοςΕΠΙΛΟΓΗ: ____

Η κάθε επιλογή αποτελεί ξεχωριστό υποπρόγραμμα.

Στην περίπτωση που έχει γεμίσει ο πίνακας να εμφανίζεται κατάλληλο μήνυμα

Βάσει του ν. 3966/2011 τα διδακτικά βιβλία του Δημοτικού, του Γυμνασίου, του Λυκείου, των ΕΠΑ.Λ. και των ΕΠΑ.Σ. τυπώνονται από το ΙΤΥΕ - ΔΙΟΦΑΝΤΟΣ και διανέμονται δωρεάν στα Δημόσια Σχολεία. Τα βιβλία μπορεί να διατίθενται προς πώληση, όταν φέρουν στη δεξιά κάτω γωνία του εμπροσθόφυλλου ένδειξη «ΔΙΑΤΙΘΕΤΑΙ ΜΕ ΤΙΜΗ ΠΩΛΗΣΗΣ». Κάθε αντίτυπο που διατίθεται προς πώληση και δεν φέρει την παραπάνω ένδειξη θεωρείται κλεψίτυπο και ο παραβάτης διώκεται σύμφωνα με τις διατάξεις του άρθρου 7 του νόμου 1129 της 15/21 Μαρτίου 1946 (ΦΕΚ 1946,108, Α').

Απαγορεύεται η αναπαραγωγή οποιουδήποτε τμήματος αυτού του βιβλίου, που καλύπτεται από δικαιώματα (copyright), ή η χρήση του σε οποιαδήποτε μορφή, χωρίς τη γραπτή άδεια του Υπουργείου Παιδείας, Έρευνας και Θρησκευμάτων / ΙΤΥΕ - ΔΙΟΦΑΝΤΟΣ.

11010011 11110101 11100011 11110111 11100001
11110001 11100111 11110100 11011110 11110001
11101001 11100001 00101110 00100000 11000010
11110001 11011110 11101010 11100101 11110010
00100000 11110100 11101111 00100000 11101010
11110001 11110101 11101100 11011101 11101101
11101111 00100000 11101100 11011110 11101101
11110101 11101100 11100001 00101110 00100000
11000011 11101001 11100001 00100000 11100010
11101111 11011110 11101000 11100101 11101001
11100001 00100000 11110011 11110100 11100001
00100000 11101100 11100001 11101000 11011110
11101100 11100001 11110100 11011100 00100000
11110011 11101111 11110101 00100000 11110000
11011110 11100011 11100001 11101001 11101101
11100101 00100000 11110011 11110100 11101111
00100000 01110111 01110111 01110111 00101110
01110011 01110100 01110101 01100100 01111001
00110100 01100101 01111000 01100001 01101101
01110011 00101110 01100111 01110010

Κωδικός Βιβλίου: 0-22-0238

ISBN 978-960-06-5209-3

ITYE
"ΔΙΟΦΑΝΤΟΣ"
ΙΝΣΤΙΤΟΥΤΟ
ΤΕΧΝΟΛΟΓΙΑΣ
ΥΠΟΛΟΓΙΣΤΩΝ & ΕΚΔΟΣΕΩΝ



(01) 000000 0 22 0238 9