

ΝΙΚΟΛΑΟΣ Φ. ΚΑΡΥΔΗΣ

Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον

Βασικές έννοιες – Μεθοδολογία
Αποθετήριο Θεμάτων



Στην οικογένειά μου

«Το παρόν έργο πνευματικής ιδιοκτησίας προστατεύεται κατά τις διατάξεις της ελληνικής νομοθεσίας (Ν. 2121/1993 όπως έχει τροποποιηθεί και ισχύει σήμερα) και τις διεθνείς συμβάσεις περί πνευματικής ιδιοκτησίας. Απαγορεύεται απολύτως η άνευ γραπτής άδειας του εκδότη κατά οποιοδήποτε τρόπο ή μέσο (ηλεκτρονικό, μηχανικό ή άλλο) αντιγραφή, φωτοανατύπωση και εν γένει αναπαραγωγή, εκμίσθωση ή δανεισμός, μετάφραση, διασκευή, αναμετάδοση στο κοινό σε οποιαδήποτε μορφή και η εν γένει εκμετάλλευση του συνόλου ή μέρους του έργου».

Εκδόσεις Πατάκη – Εκπαίδευση

Νικόλαος Φ. Καρύδης, *Ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον* – Αποθετήριο θεμάτων

Υπεύθυνος έκδοσης: Βαγγέλης Μπακλαβάς

Διορθώσεις: Μαριάνθη Κιουρτσόγλου

Σελιδοποίηση: Αλέξιος Μάστορης

Φιλμ-Μοντάζ: Μαρία Ποινιού-Ρένεση

Copyright© Σ. Πατάκης Α.Ε.Ε.Δ.Ε. (Εκδόσεις Πατάκη), Αθήνα, 2016

Πρώτη έκδοση από τις Εκδόσεις Πατάκη, Αθήνα, Αύγουστος 2017

Κ.Ε.Τ. Α895 – Κ.Ε.Π. 584/17

ISBN 978-960-16-7169-7



ΠΑΝΑΓΗ ΤΣΑΛΔΑΡΗ (ΠΡΩΗΝ ΠΕΙΡΑΙΩΣ) 38, 104 37 ΑΘΗΝΑ,
ΤΗΛ.: 210.36.50.000, 210.52.05.600, 801.100.2665, ΦΑΞ: 210.36.50.069
ΚΕΝΤΡΙΚΗ ΔΙΑΘΕΣΗ: ΕΜΜ. ΜΠΕΝΑΚΗ 16, 106 78 ΑΘΗΝΑ, ΤΗΛ.: 210.38.31.078
ΥΠΟΚΑΤΑΣΤΗΜΑ: ΚΟΡΥΤΣΑΣ (ΤΕΡΜΑ ΠΟΝΤΟΥ - ΠΕΡΙΟΧΗ Β΄ ΚΤΕΟ),
570 09 ΚΑΛΟΧΩΡΙ ΘΕΣΣΑΛΟΝΙΚΗΣ, ΤΗΛ.: 2310.70.63.54, 2310.70.67.15, ΦΑΞ: 2310.70.63.55
Web site: <http://www.patakis.gr> • e-mail: info@patakis.gr, sales@patakis.gr

ΠΕΡΙΕΧΟΜΕΝΑ

Βασικές έννοιες	5
Μεθοδολογία	29
Υποερωτήματα Θέματος Α	201
Λυμένα υποερωτήματα	203
Άλυτα υποερωτήματα	222
Υποερωτήματα Θέματος Β	329
Λυμένες ασκήσεις	331
Άλυτες ασκήσεις	342
Θέμα Γ	433
Λυμένες ασκήσεις	435
Άλυτες ασκήσεις	450
Θέμα Δ	499
Λυμένες ασκήσεις	501
Άλυτες ασκήσεις	519
Απαντήσεις	571
Υποερωτήματα Θέματος Α	573
Υποερωτήματα Θέματος Β	611
Θέμα Γ	686
Θέμα Δ	771

**ΒΑΣΙΚΕΣ
ΕΝΝΟΙΕΣ**

Τύπος δεδομένων

Είναι ένα σύνολο τιμών. Οι τύποι δεδομένων που υποστηρίζει η ΓΛΩΣΣΑ είναι:

ΑΚΕΡΑΙΕΣ, για τους ακέραιους αριθμούς (π.χ. -7)

ΠΡΑΓΜΑΤΙΚΕΣ, για τους πραγματικούς αριθμούς (π.χ. 12.3)

ΧΑΡΑΚΤΗΡΕΣ, για κείμενο (αλφαριθμητικά δεδομένα) (π.χ. 'τεστ')

ΛΟΓΙΚΕΣ, για τις λογικές τιμές (ΑΛΗΘΗΣ, ΨΕΥΔΗΣ)

Μεταβλητή

Είναι το μέγεθος του οποίου η τιμή μπορεί να αλλάξει κατά τη διάρκεια εκτέλεσης ενός προγράμματος. Κάθε μεταβλητή καταλαμβάνει μια συγκεκριμένη θέση της κύριας μνήμης του υπολογιστή, έχει ένα όνομα μοναδικό που τη χαρακτηρίζει και έναν μοναδικό τύπο δεδομένων που προσδιορίζει το είδος του περιεχομένου της. Η γενική μορφή του τμήματος δηλώσεων μεταβλητών ενός προγράμματος είναι:

ΜΕΤΑΒΛΗΤΕΣ

Τύπος_1: Λίστα_μεταβλητών_1

Τύπος_2: Λίστα_μεταβλητών_2

.....

Τύπος_κ: Λίστα_μεταβλητών_κ

όπου *Τύπος_1*, *Τύπος_2*, ..., *Τύπος_κ* ένας από τους τύπους δεδομένων που υποστηρίζει η ΓΛΩΣΣΑ.

ΠΑΡΑΔΕΙΓΜΑ

ΜΕΤΑΒΛΗΤΕΣ

ΑΚΕΡΑΙΕΣ: A

ΠΡΑΓΜΑΤΙΚΕΣ: B, Γ

ΧΑΡΑΚΤΗΡΕΣ: Δ, ΣΤ

ΛΟΓΙΚΕΣ: X, Y, Z

Σταθερά

Είναι το μέγεθος του οποίου η τιμή παραμένει αμετάβλητη κατά τη διάρκεια εκτέλεσης ενός προγράμματος. Οι σταθερές μπορεί να είναι αριθμητικές (π.χ. 17), αλφαριθμητικές (π.χ. 'δοκιμή') ή λογικές (π.χ. ΑΛΗΘΗΣ)

Συμβολική σταθερά

Είναι ονόματα που αντιπροσωπεύουν σταθερές ποσότητες και δηλώνονται στην αρχή του προγράμματος. Κάθε συμβολική σταθερά έχει ένα όνομα μοναδικό που τη χαρακτηρίζει. Να σημειωθεί ότι απαγορεύεται αυστηρά η μεταβολή του περιεχομένου μιας συμβολικής σταθεράς. Η γενική μορφή του τμήματος δηλώσεων συμβολικών σταθερών ενός προγράμματος είναι:

ΣΤΑΘΕΡΕΣ

Όνομα_Σταθεράς_1 = Τιμή_σταθεράς_1

Όνομα_Σταθεράς_2 = Τιμή_σταθεράς_2

.....

Όνομα_Σταθεράς_κ = Τιμή_σταθεράς_κ

ΠΑΡΑΔΕΙΓΜΑ**ΣΤΑΘΕΡΕΣ**

ΑΛΦΑ = 12

ΒΗΤΑ = 3.1

ΓΑΜΜΑ = 'ΤΕΣΤ'

Κανόνες ονοματολογίας

Κάθε όνομα (π.χ. μεταβλητής, συμβολικής σταθεράς, συνάρτησης) πρέπει να ξεκινάει με γράμμα και να συνεχίζεται με γράμμα ή ψηφίο. Ο μόνος ειδικός χαρακτήρας που επιτρέπεται σε ένα όνομα είναι η υπογράμμιση. Απαγορεύεται η χρήση δεσμευμένων λέξεων (π.χ. ΜΕΤΑΒΛΗΤΕΣ), η ύπαρξη κενών διαστημάτων ή αλλαγών γραμμής.

Γενική μορφή προγράμματος

Ένα πρόγραμμα στη ΓΛΩΣΣΑ έχει την ακόλουθη μορφή:

ΠΡΟΓΡΑΜΜΑ *όνομα_προγράμματος*

Τμήμα δήλωσης συμβολικών σταθερών

Τμήμα δήλωσης μεταβλητών

ΑΡΧΗ

Εντολές

ΤΕΛΟΣ_ΠΡΟΓΡΑΜΜΑΤΟΣ

Τα δύο τμήματα δηλώσεων είναι προαιρετικά, δηλαδή ένα πρόγραμμα μπορεί να μην έχει καμία συμβολική σταθερά ή μεταβλητή. Σε περίπτωση που το πρόγραμμα περιλαμβάνει και υποπρογράμματα, η συγγραφή τους γίνεται μετά το τέλος του προγράμματος.

Τελεστές

Είναι σύμβολα που αντιπροσωπεύουν πράξεις ή συγκρίσεις ανάμεσα σε δεδομένα. Διακρίνονται σε τρεις κατηγορίες: αριθμητικούς, συγκριτικούς, λογικούς.

α) Αριθμητικοί τελεστές

Υλοποιούν τις ακόλουθες αριθμητικές πράξεις:

Τελεστής	Πράξη
+	Πρόσθεση
-	Αφαίρεση
*	Πολλαπλασιασμός
/	Διαίρεση
^	Ύψωση σε δύναμη
DIV	Πηλίκιο ακέραιας διαίρεσης
MOD	Υπόλοιπο ακέραιας διαίρεσης

Η ιεραρχία (προτεραιότητά) τους είναι:

1. Ύψωση σε δύναμη
2. Πολλαπλασιασμός, διαίρεση (*, /, **DIV**, **MOD**)
3. Πρόσθεση, αφαίρεση (+, -)

Αν υπάρχουν παρενθέσεις, προηγούνται οι πράξεις μέσα στις παρενθέσεις. Πράξεις της ίδιας προτεραιότητας εκτελούνται από τα αριστερά προς τα δεξιά.

ΠΑΡΑΔΕΙΓΜΑ

$$(7+4^2 \text{ DIV } 3) \text{ MOD } 2017 = (7+16 \text{ DIV } 3) \text{ MOD } 2017 = (7+5) \text{ MOD } 2017 = 12 \text{ MOD } 2017 = 12$$

β) Λογικοί τελεστές

Υλοποιούν τις λογικές πράξεις της άρνησης (**ΟΧΙ**), της σύζευξης (**ΚΑΙ**) και της διάζευξης (**Ή**). Αν A, B λογικές μεταβλητές, το αποτέλεσμα των παραπάνω πράξεων αποτυπώνεται στον πίνακα αλήθειας που ακολουθεί:

A	B	A ΚΑΙ B	A Ή B	ΟΧΙ A	ΟΧΙ B
ΑΛΗΘΗΣ	ΑΛΗΘΗΣ	ΑΛΗΘΗΣ	ΑΛΗΘΗΣ	ΨΕΥΔΗΣ	ΨΕΥΔΗΣ
ΑΛΗΘΗΣ	ΨΕΥΔΗΣ	ΨΕΥΔΗΣ	ΑΛΗΘΗΣ	ΨΕΥΔΗΣ	ΑΛΗΘΗΣ
ΨΕΥΔΗΣ	ΑΛΗΘΗΣ	ΨΕΥΔΗΣ	ΑΛΗΘΗΣ	ΑΛΗΘΗΣ	ΨΕΥΔΗΣ
ΨΕΥΔΗΣ	ΨΕΥΔΗΣ	ΨΕΥΔΗΣ	ΨΕΥΔΗΣ	ΑΛΗΘΗΣ	ΑΛΗΘΗΣ

Η ιεραρχία (προτεραιότητά) τους είναι:

1. **ΟΧΙ**
2. **ΚΑΙ**
3. **Ή**

Αν υπάρχουν παρενθέσεις, προηγούνται οι πράξεις μέσα στις παρενθέσεις. Πράξεις της ίδιας προτεραιότητας εκτελούνται από τα αριστερά προς τα δεξιά.

ΠΑΡΑΔΕΙΓΜΑ

$$\begin{aligned} & \text{ΑΛΗΘΗΣ ΚΑΙ ΨΕΥΔΗΣ Ή ΟΧΙ(ΑΛΗΘΗΣ ΚΑΙ ΨΕΥΔΗΣ)} = \\ & \text{ΑΛΗΘΗΣ ΚΑΙ ΨΕΥΔΗΣ Ή ΟΧΙ(ΨΕΥΔΗΣ)} = \\ & \text{ΑΛΗΘΗΣ ΚΑΙ ΨΕΥΔΗΣ Ή ΑΛΗΘΗΣ} = \text{ΨΕΥΔΗΣ Ή ΑΛΗΘΗΣ} = \text{ΑΛΗΘΗΣ} \end{aligned}$$

γ) Συγκριτικοί τελεστές

Επιτρέπουν τη σύγκριση ανάμεσα σε δεδομένα και είναι οι ακόλουθοι:

Τελεστής	Λειτουργία
>	μεγαλύτερο από
<	μικρότερο από
=	ίσο με
<>	διάφορο από
>=	μεγαλύτερο ή ίσο από
<=	μικρότερο ή ίσο από

Το αποτέλεσμά τους είναι λογικού τύπου. Η σύγκριση ανάμεσα σε αλφαριθμητικά δεδομένα ακολουθεί την αλφαβητική σειρά. Σε λογικά δεδομένα μπορούν να εφαρμοστούν μόνο οι συγκριτικοί τελεστές = και <>.

ΠΑΡΑΔΕΙΓΜΑ

‘ΒΕΛΟΣ’ < ‘ΒΕΛΑΚΙ’ = ΨΕΥΔΗΣ

Έκφραση

Είναι μια παράσταση αριθμητικού ή λογικού τύπου. Διαμορφώνεται από τους τελεστές (που μπορεί να είναι μεταβλητές ή σταθερές) και από τους τελεστές. Για να υπολογίσουμε την τιμή της, αποδίδουμε τιμές στις μεταβλητές της και εκτελούμε τις πράξεις σύμφωνα με την ιεραρχία τους.

Στην περίπτωση ωστόσο μιας λογικής έκφρασης που απαρτίζεται μόνο από λογικές μεταβλητές, μπορεί να μας ζητηθεί να κατασκευάσουμε τον πίνακα αλήθειάς της. Ο πίνακας αυτός καταγράφει τις τιμές της λογικής παράστασης για κάθε δυνατό συνδυασμό τιμών των λογικών μεταβλητών που την απαρτίζουν. Σημειώνεται ότι αν η λογική παράσταση περιλαμβάνει N λογικές μεταβλητές τότε οι δυνατοί συνδυασμοί είναι 2^N (π.χ. για $N = 3$ έχουμε $2^3 = 8$ δυνατούς συνδυασμούς).

ΠΑΡΑΔΕΙΓΜΑ

Αν M, N, Λ λογικές μεταβλητές ο πίνακας αλήθειας της παράστασης

M ΚΑΙ $OXI\ N$ Ή $OXI\ \Lambda$

είναι ο ακόλουθος (συμβολίζουμε με A την τιμή $ΑΛΗΘΗΣ$ και Ψ την Τιμή $\PsiΕΥΔΗΣ$):

M	N	Λ	$OXI\ N$	$OXI\ \Lambda$	M ΚΑΙ $OXI\ N$	M ΚΑΙ $OXI\ N$ Ή $OXI\ \Lambda$
A	A	A	Ψ	Ψ	Ψ	Ψ
A	A	Ψ	Ψ	A	Ψ	A
A	Ψ	A	A	Ψ	A	A
A	Ψ	Ψ	A	A	A	A
Ψ	A	A	Ψ	Ψ	Ψ	Ψ
Ψ	A	Ψ	Ψ	A	Ψ	A
Ψ	Ψ	A	A	Ψ	Ψ	Ψ
Ψ	Ψ	Ψ	A	A	Ψ	A

Ιεραρχία τελεστών

Αν μια έκφραση περιλαμβάνει και τα τρία είδη τελεστών, η ιεραρχία είναι η ακόλουθη:

1. Αριθμητικοί
2. Συγκριτικοί
3. Λογικοί

ΠΑΡΑΔΕΙΓΜΑ

$$\mathbf{OXI} ((4+3) = 7) = \mathbf{OXI} (7 = 7) = \mathbf{OXI} \text{ ΑΛΗΘΗΣ} = \Psi\text{ΕΥΔΗΣ}$$

Εντολή εκχώρησης

Με την εντολή αυτή αποδίδεται μια συγκεκριμένη τιμή σε μια μεταβλητή. Μετά την ολοκλήρωση της εντολής το παλιό περιεχόμενο της μεταβλητής χάνεται. Η γενική μορφή της εντολής είναι:

$$\text{Μεταβλητή} \leftarrow \text{Νέο_Περιεχόμενο}$$

Θα πρέπει φυσικά ο τύπος της μεταβλητής να συμφωνεί με τον τύπο του νέου περιεχομένου. Το *Νέο_Περιεχόμενο* εξαρτάται λοιπόν από τον τύπο της μεταβλητής και μπορεί να είναι:

Τύπος μεταβλητής	Νέο_Περιεχόμενο
ΑΚΕΡΑΙΕΣ	ακέραια σταθερά ή ακέραια έκφραση ή άλλη ακέραια μεταβλητή
ΠΡΑΓΜΑΤΙΚΕΣ	πραγματική σταθερά ή ακέραια σταθερά ή έκφραση πραγματικού τύπου ή έκφραση ακέραιου τύπου ή άλλη πραγματική μεταβλητή ή ακέραια μεταβλητή
ΧΑΡΑΚΤΗΡΕΣ	αλφαριθμητικά δεδομένα (κείμενο) ή άλλη μεταβλητή τύπου ΧΑΡΑΚΤΗΡΕΣ
ΛΟΓΙΚΕΣ	λογική σταθερά (ΑΛΗΘΗΣ, ΨΕΥΔΗΣ) ή λογική έκφραση ή άλλη μεταβλητή λογικού τύπου

Όταν το *Νέο_Περιεχόμενο* είναι έκφραση, υπολογίζεται πρώτα η τιμή της και στη συνέχεια εκχωρείται στην μεταβλητή. Τέλος αν η *Μεταβλητή* είναι πραγματικού τύπου και το *Νέο_Περιεχόμενο* ακέραιου τύπου, τελικά στην μεταβλητή θα αποθηκευθεί τιμή πραγματικού τύπου (π.χ. το 12 θα αποθηκευθεί σαν 12.0)

ΠΑΡΑΔΕΙΓΜΑ

Αν X λογικού τύπου η εντολή $X \leftarrow (12 \text{ DIV } 2) > 5^2$ αποδίδει στη μεταβλητή X την τιμή ΨΕΥΔΗΣ, γιατί:

$$X = (12 \text{ DIV } 2) > 5^2 = 5 > 25 = \Psi\text{ΕΥΔΗΣ}$$

Εντολή ΔΙΑΒΑΣΕ

Με την εντολή αυτή πραγματοποιείται η εισαγωγή δεδομένων (είσοδος) σ' ένα πρόγραμμα. Η γενική μορφή της εντολής είναι:

ΔΙΑΒΑΣΕ μεταβλητή1, μεταβλητή2, ..., μεταβλητήN

όπου μεταβλητή1, μεταβλητή2, ..., μεταβλητήN μεταβλητές οποιουδήποτε τύπου.

Όταν εκτελείται η εντολή, η ροή εκτέλεσης των εντολών του προγράμματος διακόπτεται προσωρινά μέχρι τη στιγμή που θα ολοκληρωθεί η εισαγωγή των δεδομένων από το πληκτρολόγιο. Το πλήθος των εισαγόμενων δεδομένων πρέπει να ισούται με το πλήθος των μεταβλητών της εντολής. Επίσης πρέπει να υπάρχει συμφωνία τύπων ανάμεσα στα εισαγόμενα δεδομένα και στις αντίστοιχες μεταβλητές.

Εντολή ΓΡΑΨΕ

Με την εντολή αυτή πραγματοποιείται η έξοδος των αποτελεσμάτων στην οθόνη του υπολογιστή. Η γενική μορφή της εντολής είναι:

ΓΡΑΨΕ αποτέλεσμα1, αποτέλεσμα2, ..., αποτέλεσμαN

Καθένα από τα αποτελέσματα μπορεί να είναι:

- ✓ Αριθμητική σταθερά (ακέραια ή πραγματική)
- ✓ Λογική σταθερά
- ✓ Αλφαριθμητικό δεδομένο (κείμενο)
- ✓ Μεταβλητή οποιουδήποτε τύπου
- ✓ Αριθμητική έκφραση (ακέραιου ή πραγματικού τύπου)
- ✓ Λογική παράσταση

Η λειτουργία της εντολής αποτυπώνεται στον πίνακα που ακολουθεί:

Αποτέλεσμα	Έξοδος
Σταθερά	Εμφανίζεται ως έχει
Αλφαριθμητικά δεδομένα	Εμφανίζονται ως έχουν χωρίς τις αποστρόφους
Μεταβλητή	Εμφανίζεται το περιεχόμενό της
Έκφραση	Υπολογίζεται πρώτα η τιμή της και στη συνέχεια εμφανίζεται στην οθόνη

Αν τα αποτελέσματα που εμφανίζονται στην οθόνη είναι τουλάχιστον 2, χωρίζονται μεταξύ τους με κενά διαστήματα.

ΠΑΡΑΔΕΙΓΜΑ

Αν $X = 5$, η εντολή **ΓΡΑΨΕ** 'X =', X, X^2 , $X < 0$ εμφανίζει στην οθόνη:

X = 5 25 ΨΕΥΔΗΣ

Η εντολή ΑΝΤΙΜΕΤΑΘΕΣΕ

Με την εντολή αυτή μπορούμε να αντιμεταθέσουμε το περιεχόμενο δύο μεταβλητών. Η γενική μορφή της εντολής είναι:

ΑΝΤΙΜΕΤΑΘΕΣΕ μεταβλητή1 μεταβλητή2

όπου μεταβλητή1, μεταβλητή2 είναι μεταβλητές οποιουδήποτε τύπου.

ΠΑΡΑΔΕΙΓΜΑ

Αν $X = 10$, $Y = 20$, η εντολή **ΑΝΤΙΜΕΤΑΘΕΣΕ** X, Y έχει ως αποτέλεσμα η μεταβλητή X να αποκτήσει την τιμή 20 και η μεταβλητή Y την τιμή 10.

Ενσωματωμένες συναρτήσεις της ΓΛΩΣΣΑΣ

Μια αριθμητική έκφραση μπορεί να περιλαμβάνει και αναφορές συναρτήσεων. Οι ενσωματωμένες συναρτήσεις που παρέχει η ΓΛΩΣΣΑ και η λειτουργία τους απεικονίζονται στον πίνακα που ακολουθεί:

Συνάρτηση	Λειτουργία
HM	Υπολογισμός ημίτονου
ΣΥΝ	Υπολογισμός συνημίτονου
ΕΦ	Υπολογισμός εφαπτομένης
T_P	Υπολογισμός τετραγωνικής ρίζας
ΛΟΓ	Υπολογισμός φυσικού λογαρίθμου
E	Υπολογισμός της εκθετικής δύναμης (βάση το e)
A_M	Υπολογισμός του ακέραιου μέρους
A_T	Υπολογισμός της απόλυτης τιμής

ΠΑΡΑΔΕΙΓΜΑ

Η μαθηματική παράσταση $(\text{Συν}X + \eta\mu Y - 7| + e^{\text{ΕΦ}\omega} - 2)^3$ γράφεται στη ΓΛΩΣΣΑ:
 $(A_T(\Sigma YN(X) + HM(Y) - 7) + E(E\Phi(\omega)) - 2)^{\wedge}3$

Αλγοριθμική δομή (συνιστώσα)

Είναι ο τρόπος οργάνωσης και δόμησης των εντολών ενός προγράμματος. Υπάρχουν τρεις τέτοιες δομές: η δομή ακολουθίας, η δομή επιλογής και η δομή επανάληψης.

α) Δομή ακολουθίας

Είναι μια ομάδα εντολών που εκτελούνται ακριβώς μία φορά σύμφωνα με τη σειρά που έχουν γραφεί.

β) Δομή επιλογής

Είναι μια αλγοριθμική δομή στην οποία ελέγχουμε μια λογική έκφραση (συνθήκη) και ανάλογα αποφασίζουμε για τη συνέχεια των ενεργειών μας. Υπάρχουν τρεις διαφορετικές μορφές: η απλή (**AN..TOTE**), η σύνθετη (**AN..TOTE..ΑΛΛΙΩΣ**) και η πολλαπλή (**AN..TOTE..ΑΛΛΙΩΣ_ΑΝ**).

Εντολή **AN..TOTE** (απλή επιλογή)

Η γενική μορφή της εντολής είναι:

```

AN συνθήκη TOTE
    Εντολή_1
    Εντολή_2
    ...
    Εντολή_N
ΤΕΛΟΣ_ΑΝ
    
```

όπου *συνθήκη* μια λογική έκφραση και *Εντολή_1*, *Εντολή_2*, ..., *Εντολή_N* εντολές της ΓΛΩΣΣΑΣ.

Όταν εκτελείται μια **AN..TOTE**, αποτιμάται η τιμή της συνθήκης, και μόνο αν αυτή

έχει την τιμή **ΑΛΗΘΗΣ** εκτελείται και η περιεχόμενη ομάδα εντολών. Σε κάθε περίπτωση, η ροή των εντολών συνεχίζεται από την εντολή που βρίσκεται μετά το **ΤΕΛΟΣ_ΑΝ**. Ας σημειωθεί ότι σε περίπτωση που η περιεχόμενη ομάδα εντολών περιλαμβάνει μία μόνο εντολή, η δεσμευμένη λέξη **ΤΕΛΟΣ_ΑΝ** μπορεί να παραληφθεί.

ΠΑΡΑΔΕΙΓΜΑ

ΑΝ ΑΠ = 'ΝΑΙ' **ΤΟΤΕ**
 ΑΠΟΔ ← ΑΠΟΔ+ΕΠΙΔΟΜΑ
ΤΕΛΟΣ_ΑΝ

Εντολή **ΑΝ..ΤΟΤΕ..ΑΛΛΙΩΣ** (σύνθετη επιλογή)

Η γενική μορφή της εντολής είναι:

ΑΝ συνθήκη **ΤΟΤΕ**
 Ομάδα_Εντολών_1
ΑΛΛΙΩΣ
 Ομάδα_Εντολών_2
ΤΕΛΟΣ_ΑΝ

όπου *συνθήκη* μια λογική έκφραση και *Ομάδα_Εντολών_1*, *Ομάδα_Εντολών_2* εντολές της ΓΛΩΣΣΑΣ.

Όταν εκτελείται μια **ΑΝ..ΤΟΤΕ..ΑΛΛΙΩΣ**, αποτιμάται η τιμή της συνθήκης, και αν αυτή έχει την τιμή **ΑΛΗΘΗΣ** εκτελείται *Ομάδα_Εντολών_1*. Αν όμως η συνθήκη είναι **ΨΕΥΔΗΣ** εκτελείται η *Ομάδα_Εντολών_2*. Σε κάθε περίπτωση, η ροή των εντολών συνεχίζεται από την εντολή που βρίσκεται μετά το **ΤΕΛΟΣ_ΑΝ**.

ΠΑΡΑΔΕΙΓΜΑ

ΑΝ X > Y **ΤΟΤΕ**
 M ← X
ΑΛΛΙΩΣ
 M ← Y
ΤΕΛΟΣ_ΑΝ

Εντολή **ΑΝ..ΤΟΤΕ..ΑΛΛΙΩΣ_ΑΝ** (πολλαπλή επιλογή)

Η γενική μορφή της εντολής είναι:

ΑΝ συνθήκη1 **ΤΟΤΕ**
 Ομάδα_Εντολών_1
ΑΛΛΙΩΣ_ΑΝ συνθήκη2 **ΤΟΤΕ**
 Ομάδα_Εντολών_2
 ...
ΑΛΛΙΩΣ_ΑΝ συνθήκηN **ΤΟΤΕ**
 Ομάδα_Εντολών_N
ΑΛΛΙΩΣ
 Ομάδα_Εντολών_ΑΛΛΙΩΣ
ΤΕΛΟΣ_ΑΝ

όπου *συνθήκη1*, *συνθήκη2*, ..., *συνθήκηN*, λογικές εκφράσεις και *Ομάδα_Εντολών_1*, *Ομάδα_Εντολών_2*, ..., *Ομάδα_Εντολών_ΑΛΛΙΩΣ* εντολές της ΓΛΩΣΣΑΣ.

Όταν εκτελείται μια **ΑΝ..ΤΟΤΕ..ΑΛΛΙΩΣ_ΑΝ**, αποτιμάται η τιμή της *συνθήκης1*, και αν αυτή έχει την τιμή **ΑΛΗΘΗΣ** εκτελείται η *Ομάδα_Εντολών_1*. Αν όμως η *συνθήκη1* είναι **ΨΕΥΔΗΣ** αποτιμάται η *συνθήκη2*, και αν είναι **ΑΛΗΘΗΣ** εκτελείται η *Ομάδα_Εντολών_2* κτλ. Η πρώτη συνθήκη που θα αποτιμηθεί ως **ΑΛΗΘΗΣ** (από πάνω προς τα κάτω) θα επιφέρει την εκτέλεση των εντολών της αντίστοιχης ομάδας και μόνο αυτής. Αν όλες οι επιμέρους συνθήκες είναι **ΨΕΥΔΗΣ** εκτελείται η *Ομάδα_Εντολών_ΑΛΛΙΩΣ*. Σε κάθε περίπτωση, η ροή των εντολών συνεχίζεται από την εντολή που βρίσκεται μετά το **ΤΕΛΟΣ_ΑΝ**.

ΠΑΡΑΔΕΙΓΜΑ

```

ΑΝ X > 0 ΤΟΤΕ
    ΓΡΑΨΕ 'ΘΕΤΙΚΟΣ'
ΑΛΛΙΩΣ_ΑΝ X < 0 ΤΟΤΕ
    ΓΡΑΨΕ 'ΑΡΝΗΤΙΚΟΣ'
ΑΛΛΙΩΣ
    ΓΡΑΨΕ 'ΜΗΔΕΝ'
ΤΕΛΟΣ_ΑΝ

```

γ) Δομή επανάληψης

Είναι μια αλγοριθμική δομή που μας επιτρέπει να επαναλαμβάνουμε μια ομάδα εντολών πολλές φορές. Υπάρχουν τρεις διαφορετικές μορφές: το επαναληπτικό σχήμα με έλεγχο επανάληψης στην αρχή (**ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**), το επαναληπτικό σχήμα με έλεγχο επανάληψης στο τέλος (**ΜΕΧΡΙΣ_ΟΤΟΥ**) και το επαναληπτικό σχήμα ορισμένων φορών επανάληψης (**ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ**).

Εντολή **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**

Η γενική μορφή της εντολής είναι:

```

ΟΣΟ συνθήκη ΕΠΑΝΑΛΑΒΕ
    Ομάδα_Εντολών
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

όπου *συνθήκη* μια λογική έκφραση και *Ομάδα_Εντολών* είναι εντολές της **ΓΛΩΣΣΑΣ**.

Όταν εκτελείται μια **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**, επαναλαμβάνεται η περιεχόμενη ομάδα εντολών όσο η *συνθήκη* είναι **ΑΛΗΘΗΣ**. Μόλις η *συνθήκη* γίνει **ΨΕΥΔΗΣ**, η **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ** τερματίζεται. Αν η *συνθήκη* είναι ευθύς εξ αρχής **ΨΕΥΔΗΣ**, η περιεχόμενη ομάδα εντολών δεν εκτελείται καμία φορά.

ΠΑΡΑΔΕΙΓΜΑ

```

ΟΣΟ X > 0 ΕΠΑΝΑΛΑΒΕ
    ΓΡΑΨΕ X
    ΔΙΑΒΑΣΕ X
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

Εντολή ΜΕΧΡΙΣ_ΟΤΟΥ

Η γενική μορφή της εντολής είναι:

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

Ομάδα_Εντολών

ΜΕΧΡΙΣ_ΟΤΟΥ *συνθήκη*

όπου *συνθήκη* μια λογική έκφραση και *Ομάδα_Εντολών* είναι εντολές της ΓΛΩΣΣΑΣ.

Όταν εκτελείται μια **ΜΕΧΡΙΣ_ΟΤΟΥ**, επαναλαμβάνεται η περιεχόμενη ομάδα εντολών όσο η *συνθήκη* είναι ΨΕΥΔΗΣ. Μόλις η *συνθήκη* γίνει ΑΛΗΘΗΣ, η **ΜΕΧΡΙΣ_ΟΤΟΥ** τερματίζεται. Επειδή η αποτίμηση της *συνθήκης* γίνεται στο τέλος, η περιεχόμενη ομάδα εντολών εκτελείται τουλάχιστον μία φορά.

ΠΑΡΑΔΕΙΓΜΑ

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'ΔΩΣΕ ΑΡΙΘΜΟ:'

ΔΙΑΒΑΣΕ X

ΜΕΧΡΙΣ_ΟΤΟΥ X > 0

Εντολή ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ

Η γενική μορφή της εντολής είναι:

ΓΙΑ *μεταβλητή* **ΑΠΟ** *τιμή_1* **ΜΕΧΡΙ** *τιμή_2* **ΜΕ_ΒΗΜΑ** β

Ομάδα_εντολών

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Οι *τιμή_1* και *τιμή_2* είναι αριθμητικές σταθερές ή εκφράσεις ή μεταβλητές. Το β είναι και αυτό μια αριθμητική τιμή ή μεταβλητή ή έκφραση και ονομάζεται βήμα μεταβολής (αν ισούται με την τιμή 1 παραλείπεται). Ο τύπος της *μεταβλητής* (μετροητής) πρέπει να συμφωνεί με τον τύπο των άλλων αριθμητικών μεγεθών. Η *Ομάδα_εντολών* είναι εντολές της ΓΛΩΣΣΑΣ.

Όταν εκτελείται μια **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ**, η περιεχόμενη *Ομάδα_εντολών* επαναλαμβάνεται συνεχώς για όλες τις τιμές της μεταβλητής από *τιμή_1* μέχρι *τιμή_2* με ρυθμό μεταβολής β. Αν το βήμα β είναι θετικό, η *μεταβλητή* αυξάνει το περιεχόμενό της σε κάθε επανάληψη και ο βρόχος ολοκληρώνεται μόλις η τιμή της υπερβεί την *τιμή_2*. Αν το βήμα β είναι αρνητικό, η *μεταβλητή* μειώνει το περιεχόμενό της σε κάθε επανάληψη και ο βρόχος ολοκληρώνεται μόλις η τιμή της γίνει μικρότερη από την *τιμή_2*. Ειδικότερα στην περίπτωση που η *μεταβλητή*, το β, και οι *τιμή_1*, *τιμή_2* είναι ακέραιου τύπου, το πλήθος των επαναλήψεων που πραγματοποιούνται υπολογίζεται από τον τύπο: $(\text{τιμή}_2 - \text{τιμή}_1) \text{ div } \beta + 1$.

Ως προς το πλήθος των επαναλήψεων μιας **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ** ισχύουν οι κανόνες του ακόλουθου πίνακα:

Σχέση ανάμεσα σε τιμή_1 και τιμή_2	Βήμα (β)	Πλήθος επαναλήψεων
τιμή_1 < τιμή_2	θετικό	τουλάχιστον μία
	αρνητικό	0
	μηδέν	άπειρες
τιμή_1 > τιμή_2	θετικό	0
	αρνητικό	τουλάχιστον μία
	μηδέν	0
τιμή_1 = τιμή_2	θετικό	1
	αρνητικό	1
	μηδέν	άπειρες

ΠΑΡΑΔΕΙΓΜΑ

**ΓΙΑ Κ ΑΠΟ 1 ΜΕΧΡΙ 100
ΓΡΑΨΕ Κ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ**

Η εντολή GOTO

Η εντολή αυτή αλλάζει απότομα τη ροή εκτέλεσης των εντολών ενός προγράμματος. Η γενική μορφή της εντολής είναι:

GOTO N

όπου N ο αριθμός της εντολής από την οποία θα συνεχιστεί η εκτέλεση.

ΠΑΡΑΔΕΙΓΜΑ

**ΔΙΑΒΑΣΕ X, Y
AN X <> Y ΤΟΤΕ GOTO 100
ΓΡΑΨΕ 'Οι αριθμοί είναι ίσοι'
GOTO 200**

100: **ΓΡΑΨΕ** 'Οι αριθμοί είναι διαφορετικοί μεταξύ τους'

200: **ΓΡΑΨΕ** 'Τέλος ελέγχου'

Ιεραρχική σχεδίαση προγράμματος

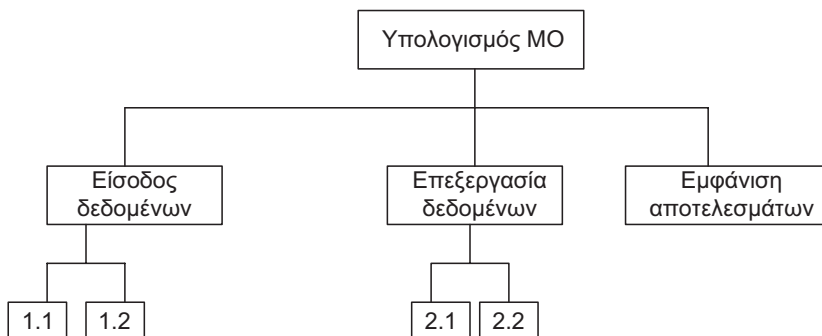
Η ιεραρχική σχεδίαση (αποκαλείται και διαδικασία σχεδίασης «από επάνω προς τα κάτω») είναι μια τεχνική διάσπασης του προβλήματος σε μια σειρά από απλούστερα υποπροβλήματα τα οποία να είναι εύκολο να επιλυθούν, οδηγώντας έτσι στην επίλυση του αρχικού προβλήματος. Για την υποβοήθηση αυτής της τεχνικής χρησιμοποιούνται διάφορες διαγραμματικές τεχνικές. Στα διαγράμματα αυτά τόσο το αρχικό πρόβλημα όσο και τα υποπροβλήματά του παριστάνονται με ορθογώνια παραλληλόγραμμα, ενώ σε κάθε επίπεδο αναλύονται τα υποπροβλήματα του αμέσως προηγούμενου επιπέδου.

ΠΑΡΑΔΕΙΓΜΑ

Για τον υπολογισμό του μέσου όρου βάρους των 28 μαθητών μιας τάξης και την εμφάνιση του σχετικού αποτελέσματος στην οθόνη του υπολογιστή πρέπει να κάνουμε τα ακόλουθα:

1. Είσοδος δεδομένων
 - 1.1. Καταχώριση δεδομένων
 - 1.2. Έλεγχος των δεδομένων
2. Επεξεργασία δεδομένων
 - 2.1 Προσθέτουμε τα ύψη των μαθητών
 - 2.2 Διαιρούμε το άθροισμα με το πλήθος των μαθητών
3. Εμφάνιση αποτελεσμάτων

Διαγραμματικά μπορούμε να αποτυπώσουμε την παραπάνω ανάλυση ως εξής:



Δομημένος προγραμματισμός

Ο δομημένος προγραμματισμός εμφανίστηκε αρχικά ως μια προσπάθεια περιορισμού των εντολών **GOTO**, σήμερα ωστόσο αποτελεί τη βασική μεθοδολογία προγραμματισμού. Στηρίζεται στη χρήση των δομών της ακολουθίας, επιλογής και επανάληψης. Όλα τα προγράμματα γράφονται χρησιμοποιώντας μόνο αυτές τις τρεις δομές καθώς και συνδυασμό τους. Η μεθοδολογία αυτή διασφαλίζει επιπρόσθετα ότι κάθε πρόγραμμα και κάθε υποπρόγραμμα έχει μόνο μία είσοδο και μόνο μία έξοδο.

ΠΑΡΑΔΕΙΓΜΑ

Αν θέλουμε να ξαναγράψουμε το παρακάτω σύνολο εντολών με τέτοιον τρόπο ώστε να ικανοποιεί τις αρχές του δομημένου προγραμματισμού

10: **ΓΡΑΨΕ** 'Δώσε θετικό αριθμό:'

ΔΙΑΒΑΣΕ X

ΑΝ X <= 0 **ΤΟΤΕ** **GOTO** 10

θα πρέπει να απομακρύνουμε την **GOTO** και να γράψουμε το ακόλουθο ισοδύναμο σύνολο εντολών

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

ΓΡΑΨΕ 'Δώσε θετικό αριθμό:'

ΔΙΑΒΑΣΕ X

ΜΕΧΡΙΣ_ΤΟΥ X > 0

Λογικά και συντακτικά λάθη

Αν σε μια εντολή έχουμε κάνει λάθος στο αλφάβητο ή στο λεξιλόγιο ή στο συντακτικό, το λάθος αυτό χαρακτηρίζεται ως **συντακτικό**. Παραδείγματα συντακτικών λαθών είναι αναγραμματισμοί ονομάτων εντολών, παράληψη δήλωσης μεταβλητών, χρήση μη αποδεκτών συμβόλων από το αλφάβητο της γλώσσας κ.ά. Τα συντακτικά λάθη εντοπίζονται στο στάδιο της μεταγλώττισης του προγράμματος. Τα **λογικά** λάθη οφείλονται σε σφάλματα κατά την υλοποίηση του αλγορίθμου, είναι πιο δύσκολα στη διάρθωσή τους και εντοπίζονται κατά την εκτέλεση του προγράμματος (όταν διαπιστώνουμε ότι τα αποτελέσματα είναι μη αναμενόμενα). Παράδειγμα λογικού λάθους είναι μια δομή επανάληψης που δεν τελειώνει (ατέρμων βρόχος), ο υπολογισμός ενός πηλίκου χωρίς προηγούμενη διασφάλιση ότι ο παρονομαστής είναι διάφορος του μηδενός κ.ά.

ΠΑΡΑΔΕΙΓΜΑΤΑ

α) Η εντολή **ΓΡΑΨΤΕ** 'ΑΕΠΠ' έχει συντακτικό λάθος.

β) Αν το πλήθος των μαθητών μας τάξης είναι ΠΛΗΘ και το πλήθος των αγοριών που φοιτούν στην τάξη αυτή είναι ΑΓΟΡ, η εντολή $ΠΟΣ \leftarrow 100 * ΠΛΗΘ / ΑΓΟΡ$ που υπολογίζει το ποσοστό (%) αυτών των αγοριών έχει λογικό λάθος (η σωστή είναι $ΠΟΣ \leftarrow 100 * ΑΓΟΡ / ΠΛΗΘ$).

Μετατροπή από μια μορφή δομής επανάληψης σε άλλη

Ζητούμενο είναι να αντικαταστήσουμε μια εντολή επανάληψης με μια εντολή επανάληψης διαφορετικής μορφής που να επιφέρει ωστόσο το ίδιο αλγοριθμικό αποτέλεσμα.

α) **Μετατροπή από ΟΣΟ..ΕΠΑΝΑΛΑΒΕ σε ΜΕΧΡΙΣ_ΟΤΟΥ**

Η εντολή

ΟΣΟ *συνθήκη* **ΕΠΑΝΑΛΑΒΕ**

Εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορεί να αντικατασταθεί από την

ΑΝ *συνθήκη* **ΤΟΤΕ**

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ

Εντολές

ΜΕΧΡΙΣ_ΟΤΟΥ ΟΧΙ(*συνθήκη*)

ΤΕΛΟΣ_ΑΝ

Αν ωστόσο είμαστε σίγουροι ότι η **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ** εκτελείται τουλάχιστον μια φορά, η εντολή **ΑΝ..ΤΟΤΕ** μπορεί να παραληφθεί.

ΠΑΡΑΔΕΙΓΜΑ

Η εντολή

ΟΣΟ X > 50 ΕΠΑΝΑΛΑΒΕ
ΓΡΑΨΕ X²
ΔΙΑΒΑΣΕ X
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορεί να αντικατασταθεί από την

ΑΝ X > 50 ΤΟΤΕ
ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ X²
ΔΙΑΒΑΣΕ X
ΜΕΧΡΙΣ_ΟΤΟΥ X <= 50
ΤΕΛΟΣ_ΑΝ

β) Μετατροπή από ΜΕΧΡΙΣ_ΟΤΟΥ σε ΟΣΟ..ΕΠΑΝΑΛΑΒΕ

Η εντολή

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
Εντολές
ΜΕΧΡΙΣ_ΟΤΟΥ συνθήκη

μπορεί να αντικατασταθεί από την ακόλουθη ομάδα εντολών:

Εντολές
ΟΣΟ ΟΧΙ(συνθήκη) ΕΠΑΝΑΛΑΒΕ
Εντολές
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Αν ωστόσο είμαστε σίγουροι ότι η **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ** εκτελείται τουλάχιστον μία φορά, δεν χρειάζεται να γράψουμε τις *Εντολές* πριν την **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**.

ΠΑΡΑΔΕΙΓΜΑ

Η εντολή

ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
ΓΡΑΨΕ ‘Δώσε αριθμό:’
ΔΙΑΒΑΣΕ X
ΜΕΧΡΙΣ_ΟΤΟΥ X > 0

μπορεί να αντικατασταθεί από την

ΓΡΑΨΕ ‘Δώσε αριθμό:’
ΔΙΑΒΑΣΕ X
ΟΣΟ X <= 0 ΕΠΑΝΑΛΑΒΕ
ΓΡΑΨΕ ‘Δώσε αριθμό:’
ΔΙΑΒΑΣΕ X
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

γ) Μετατροπή της ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ σε ΟΣΟ..ΕΠΑΝΑΛΑΒΕ

Η εντολή

ΓΙΑ μεταβλητή **ΑΠΟ** τιμή_1 **ΜΕΧΡΙ** τιμή_2 **ΜΕ_ΒΗΜΑ** β
Εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορεί να αντικατασταθεί από την ακόλουθη ομάδα εντολών:

μεταβλητή ← τιμή_1
ΟΣΟ συνθήκη **ΕΠΑΝΑΛΑΒΕ**

Εντολές

μεταβλητή ← μεταβλητή+β

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

όπου η συνθήκη είναι :

- μεταβλητή \leq τιμή_2, αν τιμή_1 \leq τιμή_2 και $\beta > 0$
- μεταβλητή \geq τιμή_2, αν τιμή_1 \geq τιμή_2 και $\beta < 0$

ΠΑΡΑΔΕΙΓΜΑ

Η εντολή

ΓΙΑ Κ **ΑΠΟ** 15 **ΜΕΧΡΙ** 2 **ΜΕ_ΒΗΜΑ** -3

ΔΙΑΒΑΣΕ Χ

ΓΡΑΨΕ Κ*Χ

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορεί να αντικατασταθεί από την ακόλουθη ομάδα εντολών:

Κ ← 15

ΟΣΟ Κ \geq 2 **ΕΠΑΝΑΛΑΒΕ**

ΔΙΑΒΑΣΕ Χ

ΓΡΑΨΕ Κ*Χ

Κ ← Κ-3

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

δ) Μετατροπή της ΟΣΟ..ΕΠΑΝΑΛΑΒΕ σε ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ

Για να είναι δυνατή η μετατροπή αυτή θα πρέπει να γνωρίζουμε τον ακριβή αριθμό των επαναλήψεων της **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**, πράγμα που σημαίνει ότι δεν μετατρέπεται κάθε **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ** σε **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ**.

Έτσι, αν τιμή_1 \leq τιμή_2 και $\beta > 0$, η ακόλουθη σειρά εντολών

μεταβλητή ← τιμή_1

ΟΣΟ μεταβλητή \leq τιμή_2 **ΕΠΑΝΑΛΑΒΕ**

Εντολές

μεταβλητή ← μεταβλητή+β

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορεί να αντικατασταθεί από την

ΓΙΑ μεταβλητή **ΑΠΟ** τιμή_1 **ΜΕΧΡΙ** τιμή_2 **ΜΕ_ΒΗΜΑ** β

Εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Επίσης, αν $τιμή_1 \geq τιμή_2$ και $\beta < 0$, και η ακόλουθη σειρά εντολών

μεταβλητή \leftarrow *τιμή_1*

ΟΣΟ *μεταβλητή* \geq *τιμή_2* **ΕΠΑΝΑΛΑΒΕ**

Εντολές

μεταβλητή \leftarrow *μεταβλητή*+ β

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορεί να αντικατασταθεί από την

ΓΙΑ *μεταβλητή* **ΑΠΟ** *τιμή_1* **ΜΕΧΡΙ** *τιμή_2* **ΜΕ_ΒΗΜΑ** β

Εντολές

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ΠΑΡΑΔΕΙΓΜΑ

Οι εντολές

$K \leftarrow 5$

ΟΣΟ $K \leq 55$ **ΕΠΑΝΑΛΑΒΕ**

ΓΡΑΨΕ K

$K \leftarrow K+2$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

μπορούν να αντικατασταθούν από την:

ΓΙΑ K **ΑΠΟ** 5 **ΜΕΧΡΙ** 55 **ΜΕ_ΒΗΜΑ** 2

ΓΡΑΨΕ K

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

ε) Μετατροπή της **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ** σε **ΜΕΧΡΙΣ_ΟΤΟΥ**

Η μετατροπή πραγματοποιείται σε δύο φάσεις. Πιο συγκεκριμένα:

- μετατρέπουμε την **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ** σε **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**
- μετατρέπουμε την **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ** σε **ΜΕΧΡΙΣ_ΟΤΟΥ**

ΠΑΡΑΔΕΙΓΜΑ

Η εντολή

ΓΙΑ K **ΑΠΟ** 4 **ΜΕΧΡΙ** M **ΜΕ_ΒΗΜΑ** 3

ΓΡΑΨΕ K

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

αντικαθίσταται αρχικά από τις

$K \leftarrow 4$

ΟΣΟ $K \leq M$ **ΕΠΑΝΑΛΑΒΕ**

ΓΡΑΨΕ K

$K \leftarrow K+3$

ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

και τελικά από την

```

ΑΝ M > = 4 ΤΟΤΕ
  K ← 4
  ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
    ΓΡΑΨΕ K
    K ← K+3
  ΜΕΧΡΙΣ_ΟΤΟΥ K > M
ΤΕΛΟΣ_ΑΝ

```

στ) Μετατροπή της **ΜΕΧΡΙΣ_ΟΤΟΥ** σε **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ**

Για να είναι δυνατή η μετατροπή αυτή θα πρέπει να γνωρίζουμε τον ακριβή αριθμό των επαναλήψεων της **ΜΕΧΡΙΣ_ΟΤΟΥ**, πράγμα που σημαίνει ότι δεν μετατρέπεται κάθε **ΜΕΧΡΙΣ_ΟΤΟΥ** σε **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ**. Αν είναι εφικτή η μετατροπή, τότε πραγματοποιείται και πάλι σε δύο φάσεις. Πιο συγκεκριμένα:

- μετατρέπουμε την **ΜΕΧΡΙΣ_ΟΤΟΥ** σε **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ**
- μετατρέπουμε την **ΟΣΟ..ΕΠΑΝΑΛΑΒΕ** σε **ΓΙΑ..ΑΠΟ..ΜΕΧΡΙ**

ΠΑΡΑΔΕΙΓΜΑ

Οι εντολές

```

  K ← 1
  ΑΡΧΗ_ΕΠΑΝΑΛΗΨΗΣ
    A ← A+K
    K ← K+7
  ΜΕΧΡΙΣ_ΟΤΟΥ K > 120

```

αντικαθίστανται αρχικά από τις

```

  K ← 1
  ΟΣΟ K <= 120 ΕΠΑΝΑΛΑΒΕ
    A ← A+K
    K ← K+7
  ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

και τελικά από τις

```

ΓΙΑ K ΑΠΟ 1 ΜΕΧΡΙ 120 ΜΕ_ΒΗΜΑ 7
  A ← A+K
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

```

Εμφωλευμένες δομές επιλογής (επανάληψης)

Είναι δύο ή περισσότερες δομές επιλογής ή επανάληψης που περιέχονται η μία μέσα στην άλλη.

ΠΑΡΑΔΕΙΓΜΑ

ΓΙΑ Κ ΑΠΟ 1 ΜΕΧΡΙ 10
ΓΙΑ Λ ΑΠΟ 1 ΜΕΧΡΙ 5
ΓΡΑΨΕ Κ+Λ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ
ΤΕΛΟΣ_ΕΠΑΝΑΛΗΨΗΣ

Αναπαράσταση αλγορίθμου με κωδικοποίηση

Η κωδικοποίηση ενός αλγορίθμου γίνεται σε ψευδογλώσσα. Η γενική μορφή του είναι:

Αλγόριθμος Όνομα

Εντολές

Τέλος Όνομα

Η σύνταξη των εντολών και ο τρόπος λειτουργίας τους είναι ακριβώς ο ίδιος που περιγράψαμε και παραπάνω. Ωστόσο θα πρέπει να επισημάνουμε κάποιες βασικές διαφορές.

- α) Οι εντολές εξόδου του αλγορίθμου είναι η **Γράψε**, η **Εκτύπωσε** και η **Εμφάνισε**. Και οι τρεις λειτουργούν με τον ίδιο ακριβώς τρόπο.
- β) Δεν υπάρχει τμήμα δήλωσης μεταβλητών.
- γ) Δεν υπάρχει δυνατότητα δήλωσης και χρήσης συμβολικών σταθερών.
- δ) Μπορούμε αντί της εντολής εισόδου **Διάβασε** να χρησιμοποιήσουμε τη δήλωση **Δεδομένα**. Γράφοντας μετά την επικεφαλίδα του αλγορίθμου
Δεδομένα // μεταβλητή1, μεταβλητή2, ..., μεταβλητήN //
 υποδηλώνουμε ότι οι παραπάνω μεταβλητές έχουν ήδη τιμή και μπορούν άμεσα να χρησιμοποιηθούν στους υπολογισμούς μας. Χρησιμοποιούμε τη δήλωση αυτή όταν η εκφώνηση του προβλήματος που επιλύει ο αλγόριθμος δεν απαιτεί την εισαγωγή των απαιτούμενων δεδομένων από τη μονάδα εισόδου με κάποια εντολή **Διάβασε**.
- ε) Αντί για τις εντολές εξόδου μπορούμε να χρησιμοποιήσουμε τη δήλωση **Αποτελέσματα**, γράφοντας:

Αποτελέσματα // αποτέλεσμα1, αποτέλεσμα2, ..., αποτέλεσμαN //

αμέσως πριν από το τέλος του αλγορίθμου. Καταφεύγουμε σ' αυτή τη δήλωση όταν η εκφώνηση του προβλήματος που επιλύει ο αλγόριθμος ζητά να υπολογίσουμε κάποια αποτελέσματα χωρίς να απαιτεί την εμφάνισή τους στη μονάδα εξόδου.

Αλγοριθμικά κριτήρια

Κάθε αλγόριθμος πρέπει να ικανοποιεί τα ακόλουθα κριτήρια:

- α) **Είσοδος**. Καμία, μία ή περισσότερες τιμές δεδομένων πρέπει να δίνονται ως είσοδοι στον αλγόριθμο. Η περίπτωση που δεν δίνονται τιμές δεδομένων εμφανίζεται όταν ο αλγόριθμος δημιουργεί και επεξεργάζεται κάποιες πρωτο-

γενείς τιμές με τη βοήθεια συναρτήσεων παραγωγής τυχαίων αριθμών ή με τη βοήθεια άλλων απλών εντολών.

- β) **Έξοδος.** Ο αλγόριθμος πρέπει να δημιουργεί τουλάχιστον μία τιμή δεδομένων ως αποτέλεσμα προς τον χρήστη ή προς έναν άλλο αλγόριθμο.
- γ) **Καθοριστικότητα.** Κάθε εντολή πρέπει να καθορίζεται χωρίς καμία αμφιβολία για τον τρόπο εκτέλεσής της.
- δ) **Περατότητα.** Ο αλγόριθμος πρέπει να τελειώνει μετά από πεπερασμένα βήματα εκτέλεσης των εντολών του. Μια διαδικασία που δεν τελειώνει μετά από έναν συγκεκριμένο αριθμό βημάτων δεν αποτελεί αλγόριθμο, αλλά λέγεται απλώς υπολογιστική διαδικασία.
- ε) **Αποτελεσματικότητα.** Κάθε μεμονωμένη εντολή του αλγορίθμου πρέπει να είναι απλή. Αυτό σημαίνει ότι μία εντολή δεν αρκεί να έχει οριστεί, αλλά πρέπει να είναι και εκτελέσιμη.

ΠΑΡΑΔΕΙΓΜΑ

Οι παρακάτω εντολές

$X \leftarrow 100$

Αρχή_επανάληψης

Γράψε $X*10$

Μέχρις_ότου $X \bmod 2 = 1$

παραβιάζουν το κριτήριο της περατότητας, αφού το X δεν αλλάζει μέσα στην επανάληψη, $X \bmod 2 = 100 \bmod 2 = 0$, και έτσι η συνθήκη ελέγχου της **Μέχρις_ότου** είναι πάντοτε Ψευδής.

Ο πολλαπλασιασμός αλά ρωσικά

Ο αλγόριθμος αυτός χρησιμοποιείται από τον υπολογιστή προκειμένου να υπολογίσει το γινόμενο δύο θετικών ακέραιων αριθμών. Αν $M1, M2$ δύο θετικοί ακέραιοι, ο αλγόριθμος που υπολογίζει το γινόμενό τους P είναι ο ακόλουθος:

Αλγόριθμος Πολ_ΡΩΣ

Δεδομένα // $M1, M2$ //

$P \leftarrow 0$

Όσο $M2 > 0$ **επανάλαβε**

Αν $M2 \bmod 2 = 1$ **τότε**

$P \leftarrow P+M1$

Τέλος_αν

$M1 \leftarrow M1*2$

$M2 \leftarrow M2 \text{ div } 2$

Τέλος_επανάληψης

Γράψε 'Γινόμενο =', P

Τέλος Πολ_ΡΩΣ

ΠΑΡΑΔΕΙΓΜΑ

Αν $M1 = 13$ και $M2 = 20$ η εκτέλεση του αλγορίθμου αποτυπώνεται στον πίνακα που ακολουθεί:

M1	M2	
13	20	
26	10	
52	5	52
104	2	
208	1	208
Άθροισμα =		260

ΠΑΡΑΤΗΡΗΣΗ: Αν θέλουμε να επεκτείνουμε τη μέθοδο για οποιουδήποτε ακέραιους γράφουμε:

Αν $(M1 = 0)$ **Ή** $(M2 = 0)$ **τότε**

Γινόμενο $\leftarrow 0$

αλλιώς

Απολ1 $\leftarrow A_T(M1)$

Απολ2 $\leftarrow A_T(M2)$

$P \leftarrow 0$

Όσο $Aπολ2 > 0$ **επανάλαβε**

Αν $Aπολ2 \bmod 2 = 1$ **τότε**

$P \leftarrow P + Απολ1$

Τέλος_αν

$Aπολ1 \leftarrow Απολ1 * 2$

$Aπολ2 \leftarrow Απολ2 \text{ div } 2$

Τέλος_επανάληψης

Αν $((M1 < 0)$ **ΚΑΙ** $(M2 < 0))$ **Ή** $((M1 > 0)$ **ΚΑΙ** $(M2 > 0))$ **τότε**

Γινόμενο $\leftarrow P$

αλλιώς

Γινόμενο $\leftarrow P * (-1)$

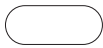
Τέλος_αν

Τέλος_αν

Γράψε 'Γινόμενο =', P

Διάγραμμα ροής

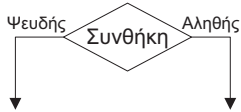
Είναι ένας γραφικός τρόπος αναπαράστασης ενός αλγορίθμου. Τα γεωμετρικά σχήματα που χρησιμοποιούνται είναι τα ακόλουθα:



Αρχή ή τέλος του διαγράμματος



Είσοδος ή έξοδος δεδομένων



Έλεγχος συνθήκης

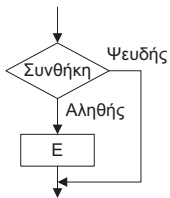


Εντολή εκχώρησης

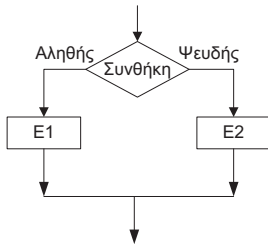
Η ροή των εντολών αποτυπώνεται με τη βοήθεια βελών.

Διαγραμματική αναπαράσταση δομών επιλογής

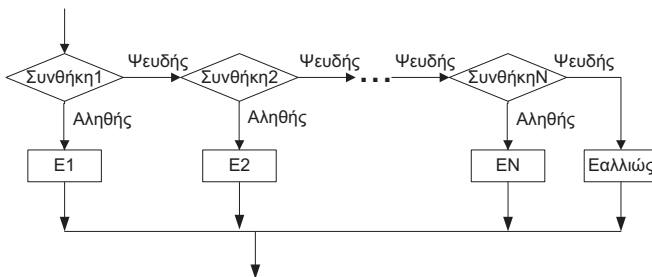
Η διαγραμματική αναπαράσταση των δομών επιλογής είναι η ακόλουθη:



Αν Συνθήκη τότε
E
Τέλος_αν



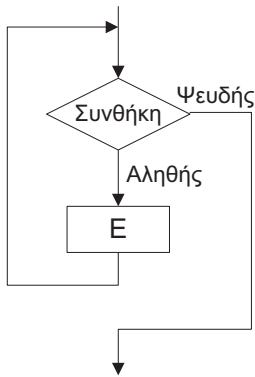
Αν Συνθήκη τότε
E1
αλλιώς
E2
Τέλος_αν



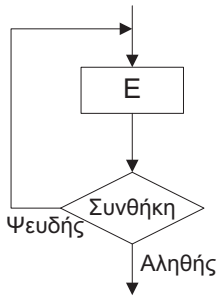
Αν Συνθήκη1 τότε
E1
αλλιώς_αν Συνθήκη2 τότε
E2
...
αλλιώς_αν ΣυνθήκηN τότε
EN
αλλιώς
Εαλλιώς
Τέλος_αν

Διαγραμματική αναπαράσταση δομών επανάληψης

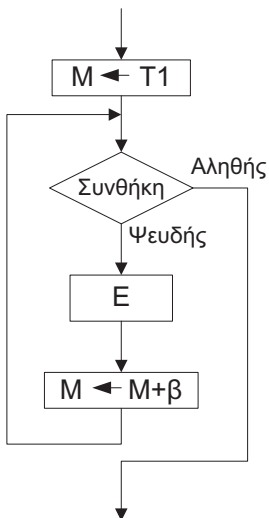
Η διαγραμματική αναπαράσταση των δομών επανάληψης είναι η ακόλουθη:



Όσο Συνθήκη επανάλαιβε
Ε
Τέλος_επανάληψης



Αρχή_επανάληψης
Ε
Μέχρις_ότου Συνθήκη



Για M από T1 μέχρι T2 με_βήμα β
Ε
Τέλος_επανάληψης

Συνθήκη = $M > T2$, αν $T1 \leq T2$ και $B > 0$