

Κεφάλαιο 2 - Εισαγωγή

🟦 Το κεφάλαιο 2 αναφέρεται στην έννοια του αλγορίθμου. **Αλγόριθμος**, σύμφωνα με το βιβλίο, είναι μια πεπερασμένη σειρά ενεργειών, αυστηρά καθορισμένων και εκτελέσιμων σε πεπερασμένο χρόνο, που στοχεύουν στην επίλυση ενός προβλήματος.

Τα πέντε κριτήρια που πρέπει να ικανοποιεί ένας αλγόριθμος είναι:

- Να έχει δεδομένα εισόδου που θα επεξεργαστεί (είσοδος)
- Να περιέχει απλές εντολές (αποτελεσματικότητα)
- Κάθε εντολή να ορίζεται χωρίς αμφιβολία ως προς την εκτέλεσή της, π.χ. διαίρεση με το μηδέν (καθοριστικότητα)
- Να ολοκληρώνεται μετά από πεπερασμένο αριθμό βημάτων (περατότητα)
- Να επιστρέφει το αποτέλεσμα της επεξεργασίας (έξοδος)

🟦 Ένας αλγόριθμος μπορεί να αναπαρασταθεί με:

- Ελεύθερο κείμενο
- Διαγραμματικές τεχνικές (διάγραμμα ροής)
- Φυσική γλώσσα κατά βήματα
- Κωδικοποίηση (πρόγραμμα γραμμένο σε ψευδογλώσσα είτε σε γλώσσα προγραμματισμού)

🟦 Για την συγγραφή ενός αλγορίθμου χρησιμοποιούνται:

🟦 **Μεταβλητές:** Πρόκειται για ένα γλωσσικό αντικείμενο (το οποίο μπορούμε να χαρακτηρίσουμε και όνομα) όπου χρησιμοποιείται για να αναπαραστήσει ένα στοιχείο εισόδου. Μπορούμε να θεωρήσουμε τις μεταβλητές ως θέσεις μνήμης με συγκεκριμένο όνομα όπου περιέχεται μια τιμή η οποία και μπορεί να μεταβάλλεται κατά την εκτέλεση του αλγορίθμου. Οι μεταβλητές διακρίνονται σε:

- Αριθμητικές: που αποθηκεύουν τιμές όπως 70, -32,5 κ.ο.κ.
- Αλφαριθμητικές: που αποθηκεύουν τιμές όπως "Ιάσοντας", " abc " κ.ο.κ.
- Λογικές με τιμή *αληθής* ή *ψευδής*

Τα ονόματα των μεταβλητών μπορούν να περιλαμβάνουν πεζά ή κεφαλαία γράμματα, αριθμούς και το χαρακτήρα *κάτω παύλα* (_). Το όνομα πρέπει να ξεκινά από χαρακτήρα και όχι αριθμό χωρίς κενά. Αν επιθυμούμε την ύπαρξη δυο λέξεων τότε χρησιμοποιείται ή κάτω παύλα π.χ. Μέγιστη_Τιμή. Δεν επιτρέπεται να

χρησιμοποιηθεί ως όνομα μεταβλητής κάποια από τις δεσμευμένες λέξεις της γλώσσας προγραμματισμού ή ακόμη και της ψευδογλώσσας που χρησιμοποιείται για την απεικόνιση του αλγορίθμου

Ουσιαστικά, χρησιμοποιούμε μια μεταβλητή για να αποθηκεύσουμε σε αυτήν κάποια τιμή (είσοδο δεδομένων από το χρήστη ή το αποτέλεσμα κάποιας έκφρασης). Όταν χρησιμοποιούμε την μεταβλητή επικαλούμαστε την τιμή που αυτή περιέχει

Αποδεκτά ονόματα μεταβλητών: τιμή1, μέσος_όρος, ΜέγιστηΤιμή, A10

Μη αποδεκτά ονόματα μεταβλητών: 12a, μέση τιμή

Σταθερές: Πρόκειται για προκαθορισμένες τιμές που παραμένουν αμετάβλητες κατά την εκτέλεση του αλγορίθμου. Και αυτές διακρίνονται σε αριθμητικές, αλφαριθμητικές και λογικές. Η έννοια της σταθεράς θα γίνει καλύτερα αντιληπτή στο κεφάλαιο 7

Τελεστές: Πρόκειται για τα γνωστά σύμβολα των πράξεων

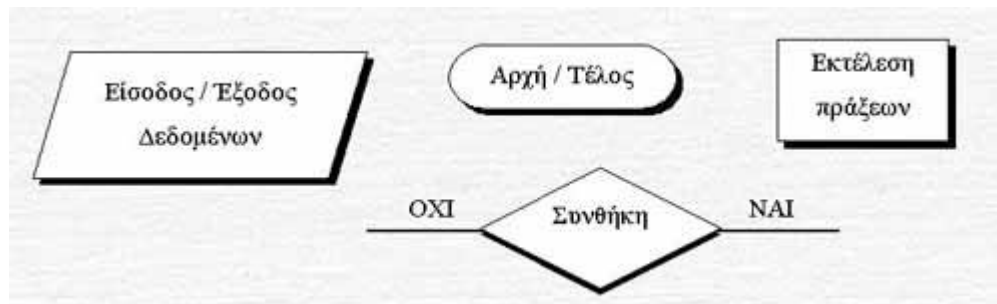
- Αριθμητικοί: +, -, *, /, ^ για την ύψωση σε δύναμη π.χ. το 8^2 συμβολίζεται με 8^2 . Ακόμη υπάρχουν και οι τελεστές div και mod. Ο τελεστής div επιστρέφει το ηλίκο της διαίρεσης δυο αριθμών και το mod το υπόλοιπο αντίστοιχα. Έτσι, $13 \text{ div } 3 = 4$ ενώ $13 \text{ mod } 3 = 1$. Ο τελεστής mod χρησιμοποιείται στις περιπτώσεις όπου πρέπει να διερευνηθεί αν κάποιος αριθμός είναι άρτιος ή περιττός, δηλαδή αν είναι πολλαπλάσιο του 2 ή όχι. Έτσι, $4 \text{ mod } 2 = 0$ ενώ $13 \text{ mod } 2 = 1$
- Συγκριτικοί: >, <, >=, <=, = και <>
- Λογικοί: και, ή, όχι οι οποίοι συνδέουν συνθήκες. Το αποτέλεσμα είναι πάντα μια λογική τιμή: αληθής ή ψευδής

Εκφράσεις: Συνδυασμοί μεταβλητών ή σταθερών και τελεστών. Το αποτέλεσμα μιας έκφρασης αποδίδεται σε μια μεταβλητή με εκχώρηση τιμής π.χ. τιμή ← α + β, όπου το αποτέλεσμα του αθροίσματος των τιμών των μεταβλητών α και β θα εκχωρηθεί στη μεταβλητή τιμή (το βελάκι δείχνει την ενέργεια). Η τελική τιμή μια εκχώρησης εξαρτάται από την ιεραρχία των πράξεων (γνωστή από τα μαθηματικά) και τις παρενθέσεις

Αποδεκτές εκφράσεις: τιμή ← 5, τιμή ← "αρκετά", τιμή ← α * β, τιμή ← τιμή + 3

Μη αποδεκτές εκφράσεις: τιμή $\leftarrow 5 + "χ"$, τιμή $\leftarrow ασ \leftarrow 6$, $α + τιμή \leftarrow 6$

● Διάγραμμα ροής. Ένας από τους δυνατούς τρόπους απεικόνισης αλγορίθμων είναι και το διάγραμμα ροής. Τα σύμβολα αυτά είναι:



● Δομή ακολουθίας

Όταν για την εκτέλεση ενός αλγορίθμου απαιτείται σειριακή εκτέλεση εντολών

Παράδειγμα 1: Ανάγνωση και εκτύπωση 2 αριθμών.

Αλγόριθμος Παράδειγμα_1

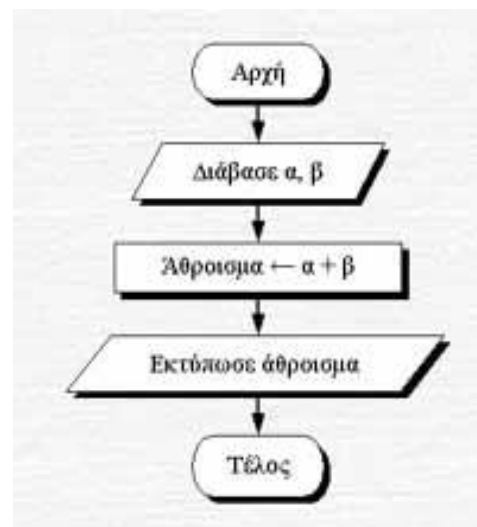
Διάβασε $α, β$ ή **Δεδομένα** // $α, β$ //

$άθροισμα \leftarrow α + β$

Εκτύπωσε $άθροισμα$ ή **Αποτελέσματα**

// $άθροισμα$ //

Τέλος Παράδειγμα_1



- Οι εντολές εκτελούνται σειριακά με τη φορά των βελών

Σχολιασμός διαγράμματος

ροής:

- Πάντοτε θα υπάρχει 1 έλλειψη για αρχή και μία για τέλος

Σχολιασμός ψευδοκώδικα:

- Οι εντολές εκτελούνται σειριακά όπως παρατίθενται στον αλγόριθμο

- Στην αρχή του αλγορίθμου πραγματοποιείται η είσοδος των δεδομένων και τελευταία ενέργεια είναι η έξοδος – εκτύπωσή τους
- Πάντοτε ξεκινάμε με τη λέξη *Αλγόριθμος* και ακολουθεί το όνομα του αλγορίθμου. Το όνομα ενός αλγορίθμου ακολουθεί τους ίδιους κανόνες με το όνομα μιας μεταβλητής. Επιπλέον, δεν μπορεί να χρησιμοποιείται ίδιο όνομα για μια μεταβλητή και για τον αλγόριθμο. Αυτές οι λέξεις καθώς και όλες όσες είναι έντονα γραμμένες καλούνται **δεσμευμένες** λέξεις και δεν επιτρέπεται να παρουσιαστούν αλλιώς, ούτε να χρησιμοποιηθούν με λάθος τρόπο
- Για την χρησιμοποίηση της τιμής κάποιας μεταβλητής πρέπει οπωσδήποτε να έχει προηγηθεί εκχώρηση τιμής σε αυτή τη μεταβλητή είτε με εντολή εκχώρησης είτε με είσοδο δεδομένων από το χρήστη. Πιο πρακτικά, δεν μπορούμε να χρησιμοποιήσουμε μια μεταβλητή στο δεξί τμήμα μιας εντολής εκχώρησης ή σε μια εκτύπωση αν προηγουμένως δεν έχει λάβει τιμή στον αλγόριθμο - **αρχικοποίηση**
- Η πρώτη ή οι πρώτες εντολές πραγματοποιούν είσοδο δεδομένων. Αυτό μπορεί να γίνει με την εντολή *Διάβασε* ακολουθούμενη από τα ονόματα μεταβλητών ή με την εντολή *Δεδομένα* (δες αλγόριθμο)
Όταν η εκφώνηση αναφέρει ρητώς την ανάγνωση των δεδομένων από το χρήστη τότε χρησιμοποιείται η εντολή *Διάβασε*. Σε διαφορετική περίπτωση π.χ. όταν η εκφώνηση αναφέρει «έστω μεταβλητή N που περιέχει το πλήθος αριθμών», τότε μπορούμε να εισάγουμε την τιμή της μεταβλητής N στον αλγόριθμό μας με την εντολή *Δεδομένα*. Όμοια για την έξοδο και τις εντολές *Εκτύπωσε* (για αποστολή εκτύπωσης στον εκτυπωτή), *Εμφάνισε* (για εμφάνιση των στοιχείων στην οθόνη) και της εντολής *Αποτελέσματα*

Δομές Επιλογής

Δομή επιλογής

Πολλές φορές για να λυθεί ένα πρόβλημα πρέπει να ελεγχθεί αν ισχύει κάποια συνθήκη

Παράδειγμα 2: Να διαβαστεί ένας αριθμός και να επιστραφεί η απόλυτη τιμή του

Ο εισαγόμενος αριθμός ελέγχεται αν είναι θετικός ή αρνητικός. Στην πρώτη περίπτωση δεν χρειάζεται να γίνει καμιά ενέργεια, ενώ στη δεύτερη πρέπει να πολλαπλασιαστεί ο αριθμός με το -1 ώστε να γίνει θετικός

Αλγόριθμος Παράδειγμα_2

Διάβασε αριθμός

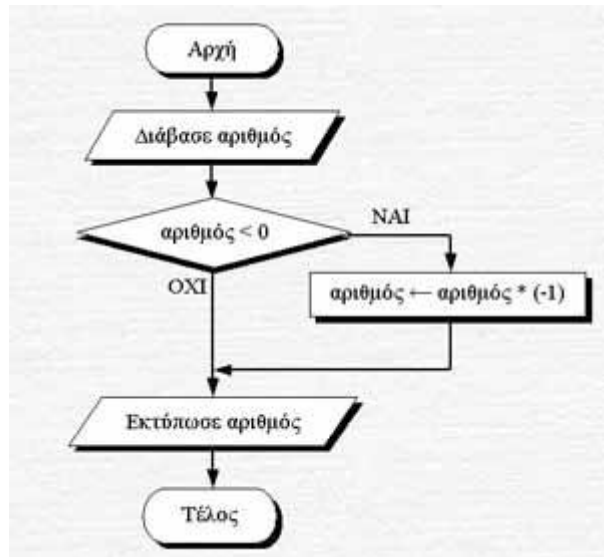
Αν αριθμός < 0 **τότε**

αριθμός \leftarrow αριθμός $\cdot (-1)$

Τέλος_Αν

Εκτύπωσε αριθμός

Τέλος Παράδειγμα_2



Σχολιασμός διαγράμματος ροής: Ο ρόμβος (έλεγχος συνθήκης) ελέγχει συνθήκες και έχει δυο εξόδους (μία θετική και μία αρνητική στην ερώτηση)

Σχολιασμός ψευδοκώδικα:

- Η εντολή αριθμός \leftarrow αριθμός $\cdot (-1)$, δεν είναι μαθηματική έκφραση αλλά εντολή εκχώρησης. Επιτρέπεται λοιπόν, η παρουσία της ίδιας μεταβλητής αριστερά και δεξιά του \leftarrow . Έτσι, πολλαπλασιάζεται το περιεχόμενο της μεταβλητής αριθμός με το -1 και το αποτέλεσμα καταχωρείται εκ νέου στη μεταβλητή αριθμός

Αν συνθήκη τότε

Εντολή1

...

ΕντολήN

Τέλος_αν

Παράδειγμα 3: Να διαβαστεί ένας αριθμός και να εκτυπωθεί ο αντίστροφός του αριθμός

Αλγόριθμος Παράδειγμα_3

Διάβασε αριθμός

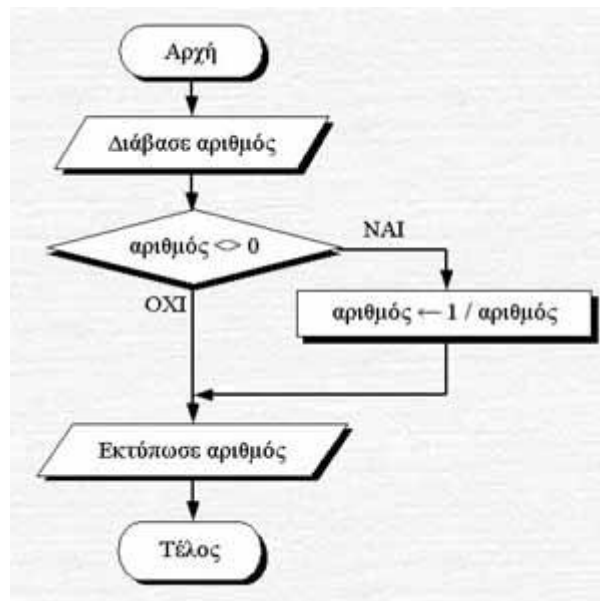
Αν αριθμός \neq 0 **τότε**

αριθμός \leftarrow 1 / αριθμός

Τέλος_Αν

Εκτύπωσε αριθμός

Τέλος Παράδειγμα_3



Δομή Σύνθετης επιλογής

Η δομή σύνθετης επιλογής χρησιμοποιείται στην περίπτωση που επιθυμούμε να εκτελέσουμε εναλλακτικά δυο ομάδες εντολών. Κριτήριο για το ποιο σεν εντολών θα εκτελεστεί αποτελεί κάποια συνθήκη. Στην περίπτωση που η συνθήκη ισχύει θα εκτελεστεί η πρώτη ομάδα εντολών διαφορετικά θα εκτελεστεί η δεύτερη

Παράδειγμα 4: Να διαβαστούν δύο αριθμοί και να εκτυπωθεί ο μεγαλύτερος από τους δυο.

Αλγόριθμος Παράδειγμα_4

Διάβασε αριθμός1, αριθμός2

Αν αριθμός1 < αριθμός2 **τότε**

αριθμός \leftarrow αριθμός2

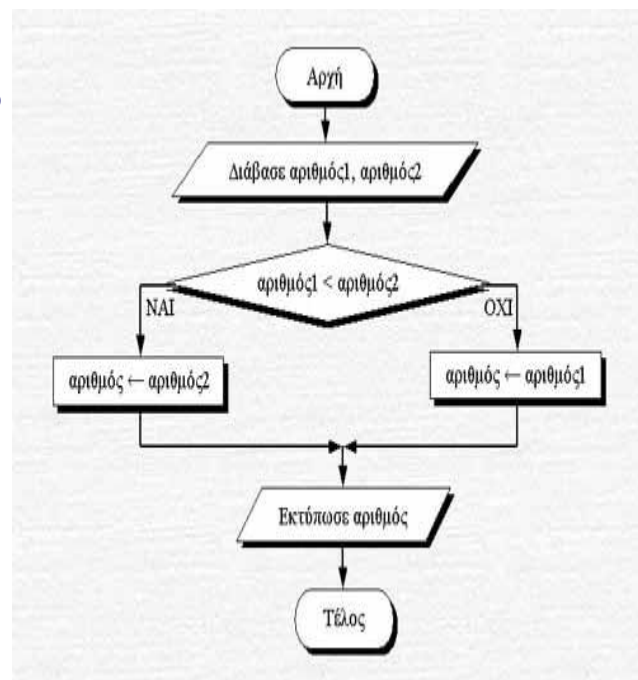
Αλλιώς

αριθμός \leftarrow αριθμός1

Τέλος_Αν

Εκτύπωσε αριθμός

Τέλος Παράδειγμα_4



Παράδειγμα 5: Να διαβαστεί αριθμός και να εκτυπωθεί αντίστοιχο μήνυμα για το αν είναι άρτιος ή περιττός

Αλγόριθμος

Παράδειγμα_5

Διάβασε αριθμός

Αν αριθμός mod 2 = 0 **τότε**

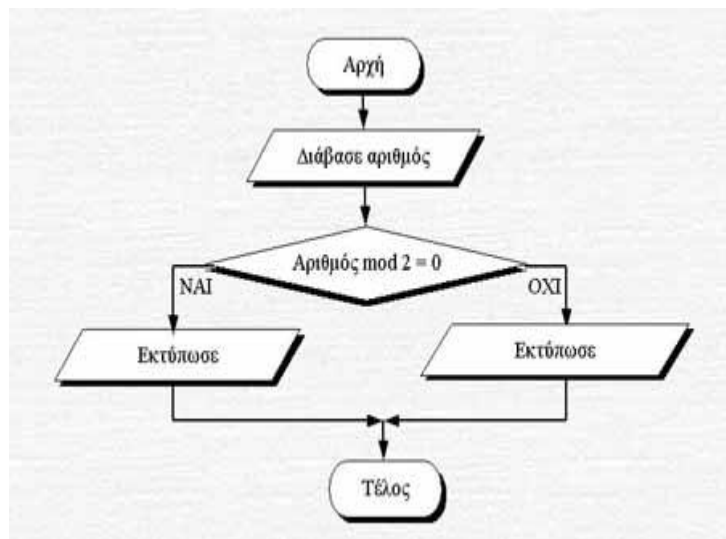
Εκτύπωσε "Άρτιος"

Αλλιώς

Εκτύπωσε "Περιττός"

Τέλος_Αν

Τέλος Παράδειγμα_5



Δομή πολλαπλής επιλογής

Κάποια προβλήματα απαιτούν την επιλογή μεταξύ περισσότερων από δυο περιπτώσεις

Παράδειγμα 6: Να διαβαστεί ο μέσος όρος ενός μαθητή και να εκτυπωθεί χαρακτηρισμός του

Αλγόριθμος Παράδειγμα_6

Διάβασε βαθμός

Αν βαθμός < 9,5 **τότε**

Εκτύπωσε "Απορρίπτεται"

Αλλιώς_Αν βαθμός < 16 **τότε**

Εκτύπωσε "Καλώς"

Αλλιώς_Αν βαθμός < 18 **τότε**

Εκτύπωσε "Λίαν καλώς"

Αλλιώς_Αν βαθμός < 20 **τότε**

Εκτύπωσε "Άριστα"

Αλλιώς

Εκτύπωσε "Λάθος δεδομένο"

Τέλος_Αν

Τέλος Παράδειγμα_6

Αλγόριθμος Παράδειγμα_6

Διάβασε βαθμός

Επίλεξε βαθμός

Περίπτωση < 9,5

Εκτύπωσε "Απορρίπτεται"

Περίπτωση < 16

Εκτύπωσε "Καλώς"

Περίπτωση < 18

Εκτύπωσε "Λίαν καλώς"

Περίπτωση < 20

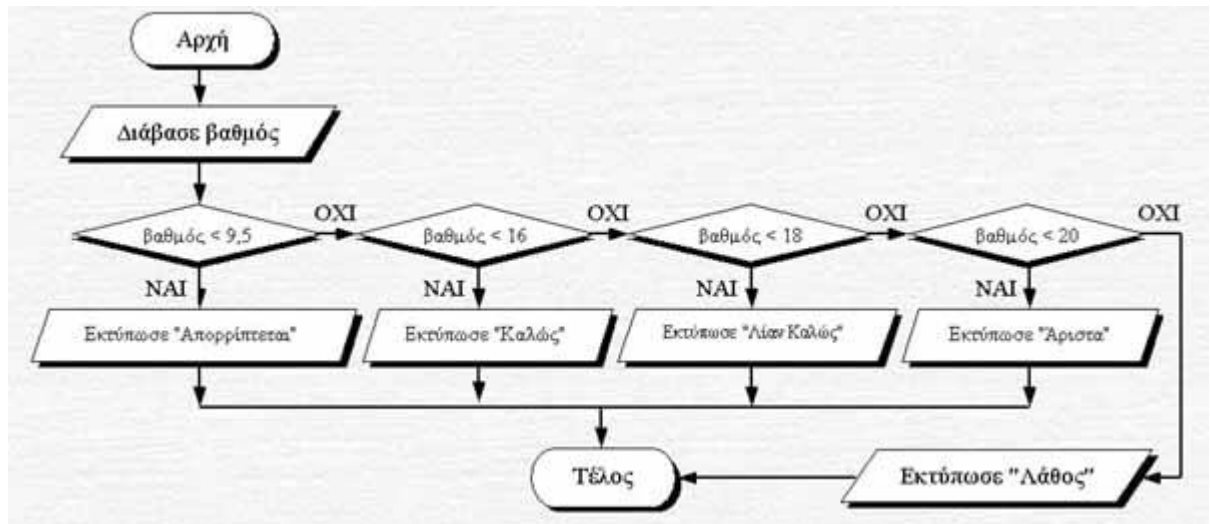
Εκτύπωσε "Άριστα"

Περίπτωση Αλλιώς

Εκτύπωσε "Λάθος δεδομένο"

Τέλος_Επιλογών

Τέλος Παράδειγμα_6



Σχολιασμός:

- **Μια από τις περιπτώσεις** της δομής επιλογής εκτελούνται σε κάθε περίπτωση
- Στη δομή **Επίλεξε** αν δεν ικανοποιείται καμία περίπτωση τότε εκτελείται το **Περίπτωση Αλλιώς**
- Τα σύμβολα ' και " χρησιμοποιούνται ισοδύναμα
- Τα παραπάνω θα μπορούσαν να υλοποιηθούν και με πολλές δομές σύνθετης επιλογής

Εμφωλευμένες Δομές

Στην υλοποίηση ενός αλγορίθμου μπορούμε να χρησιμοποιούμε κάποιες από τις παραπάνω δομές ως τμήμα εντολών σε άλλες

Παράδειγμα Ζ: Να διαβαστούν τα έτη υπηρεσίας και ο μισθός ενός υπαλλήλου και να εκτυπωθεί το μπόνους σύμφωνα με τον πίνακα

| Έτη υπηρεσίας | Μισθός | Μπόνους |
|---------------|-----------|---------|
| < 10 | | - |
| > 10 | < 300.000 | 10 % |
| | > 300.000 | 12 % |

Αλγόριθμος Παράδειγμα_7

Διάβασε έτη_υπηρεσίας, μισθός

Αν έτη_υπηρεσίας < 10 **τότε**
 μπόνους ← 0

Αλλιώς

Αν μισθός < 300000 **τότε**
 μπόνους ← 0,1 * μισθός

Αλλιώς

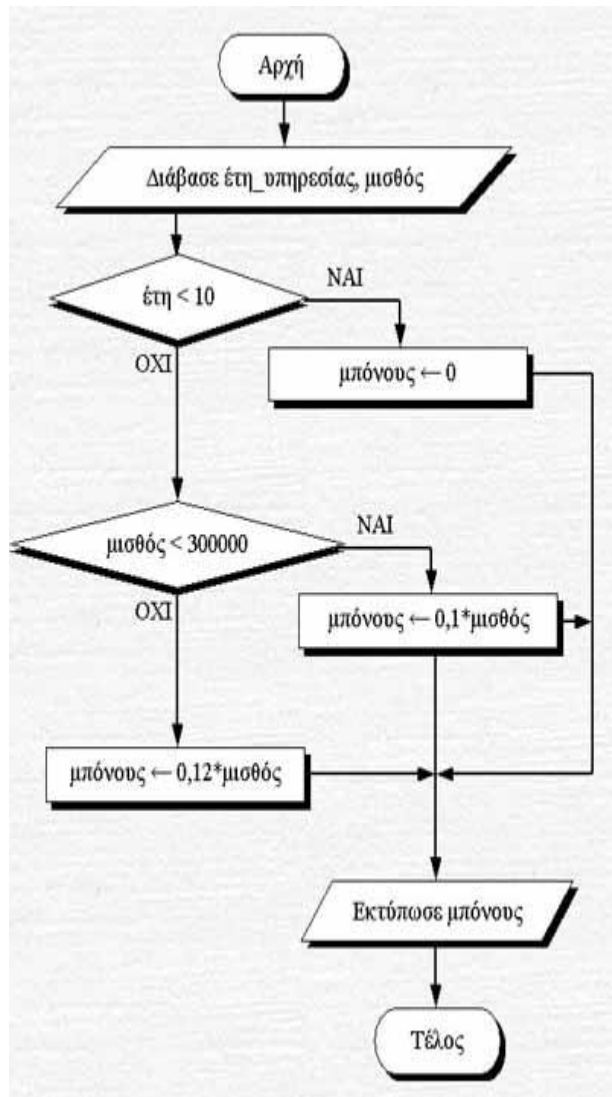
μπόνους ← 0,12 * μισθός

Τέλος_Αν

Τέλος_Αν

Εκτύπωσε μπόνους

Τέλος Παράδειγμα_7



Παρατηρήσεις: Στο παραπάνω πρόβλημα οι δυο δομές επιλογής θα μπορούσαν να συγχωνευτούν σε μια, της οποίας η συνθήκη περιέχει και τις δυο παραπάνω συνθήκες. Για το σκοπό αυτό χρησιμοποιούνται οι συγκριτικοί τελεστές

| Πρόταση A | Πρόταση B | A ή B | A και B | όχι A |
|-----------|-----------|--------|---------|--------|
| Αληθής | Αληθής | Αληθής | Αληθής | Ψευδής |
| Αληθής | Ψευδής | Αληθής | Ψευδής | Ψευδής |
| Ψευδής | Αληθής | Αληθής | Ψευδής | Αληθής |
| Ψευδής | Ψευδής | Ψευδής | Ψευδής | Αληθής |

Σχολιασμός πίνακα:

- Η πρόταση A ή B είναι αληθής όταν κάποια από τις δύο ή και οι δυο είναι αληθείς
- Η πρόταση A και B είναι αληθής όταν και οι δυο προτάσεις A, B είναι αληθείς
- Η πρόταση όχι A είναι αληθής όταν και η πρόταση A είναι ψευδής

Η ιεραρχία των πράξεων των λογικών τελεστών είναι: όχι, και, ή

Έτσι, ας υποθέσουμε ότι: $A=8, B=2$

$(A \geq 7)$ και $(B=1)$
 Ψευδής Ψευδής **Ψευδής**

$A < 20$ ή $B < > 4$
 αληθής Ψευδής **αληθής**

$A > B$ και $B < > 6$
 αληθής αληθής **αληθής**

όχι $B < 6$
 αληθής **ψευδής**

$A \bmod 2 = 0$ ή $B \bmod 2 = 1$
 αληθής Ψευδής **αληθής**

Σύμφωνα με τα παραπάνω ο αλγόριθμος του παραδείγματος 7 τροποποιείται ως εξής:

Αλγόριθμος Παράδειγμα_7_εναλλακτικός

Διάβασε έτη_υπηρεσίας, μισθός

Αν έτη_υπηρεσίας < 10 **τότε**

μόνους ← 0

Τέλος_Αν

Αν (έτη_υπηρεσίας > 10) **και** (μισθός < 300000) **τότε**

μόνους ← 0,1 * μισθός

Τέλος_Αν

Αν (έτη_υπηρεσίας > 10) **και** (μισθός > 300000) **τότε**

μόνους ← 0,12 * μισθός

Τέλος_Αν

Εκτύπωσε μπόνους

Τέλος Παράδειγμα_7_εναλλακτικός

Δομές Επανάληψης

Πολλές φορές πρέπει να εκτελεστεί μια ομάδα εντολών περισσότερες από μια φορές. Σ' αυτές τις περιπτώσεις χρησιμοποιούνται οι δομές επανάληψης. Υπάρχουν δυο περιπτώσεις: το πλήθος των επαναλήψεων να είναι γνωστό εξ αρχής ή όχι

Η δομή επανάληψης ΓΙΑ

Όταν είναι γνωστό το πλήθος των επαναλήψεων χρησιμοποιούμε την δομή επανάληψης ΓΙΑ

Παράδειγμα 8: Να διαβαστούν 20 αριθμοί και να εκτυπωθεί ο μέσος όρος τους.

Αλγόριθμος Παράδειγμα_8

άθροισμα ← 0

Για *i* από 1 μέχρι 20

Διάβασε αριθμός

 άθροισμα ← άθροισμα + αριθμός

Τέλος_Επανάληψης

μέσος_όρος ← άθροισμα / 20

Εκτύπωσε μέσος_όρος

Τέλος Παράδειγμα_8

Επαναληπτική δομή, οι εντολές που περιέχονται θα εκτελεστούν 20 φορές

Μετά τον υπολογισμό του αθροίσματος η ροή του αλγορίθμου επανέρχεται στην εντολή Για

Για *i* από 1 μέχρι 20

...

Τέλος_Επανάληψης

Η μεταβλητή *i* θα αυξηθεί από την τιμή 1 μέχρι την τιμή 20 (την ονομάζουμε *μετρητή*) κατά μία μονάδα κάθε φορά. Σύμφωνα με την παράγραφο 5.1.3 του σχολικού βιβλίου η μεταβλητή *i* θα αυξηθεί 20 φορές, άρα η τελική τιμή που θα λάβει είναι 21. Αφού διαβαστεί κάθε ένας αριθμός από τους 20 και εισχωρηθεί στην ομώνυμη μεταβλητή, προστίθεται στη μεταβλητή. Το τμήμα εντολών που βρίσκεται εντός του Για ονομάζεται βρόχος

Πρέπει επίσης, να τονιστεί ότι η εντολή άθροισμα ← άθροισμα + αριθμός, προσθέτει το περιεχόμενο της μεταβλητής *άθροισμα* με αυτό της *αριθμός* και τοποθετεί το αποτέλεσμα στην μεταβλητή *άθροισμα*. Ένα σύνηθες λάθος των μαθητών είναι να γράφουν λανθασμένα άθροισμα ← αριθμός

Με τη χρήση της εντολής αυτής όμως, θα καταχωρηθεί στη μεταβλητή *άθροισμα* ο τελευταίος αριθμός που διαβάστηκε. Πρέπει λοιπόν, στην τρέχουσα τιμή της μεταβλητής *άθροισμα* να προστίθεται η νέα τιμή

Επιπρόσθετα, πρέπει έξω από την επαναληπτική δομή να μηδενίσουμε την μεταβλητή *άθροισμα*, διαφορετικά δεν μπορούμε να γνωρίζουμε την τιμή της κατά την πρώτη επανάληψη

Γενικώς, δεν μπορούμε να χρησιμοποιήσουμε μια μεταβλητή, αν προηγούμενα δεν έχει λάβει τιμή – αρχικοποίηση. Η αρχικοποίηση μπορεί να πραγματοποιηθεί είτε με την είσοδο δεδομένων σε μια μεταβλητή (εντολή Διάβασε) είτε με εκχώρηση τιμής

Ας δούμε κάποια νέα στοιχεία με τη βοήθεια ενός ακόμη παραδείγματος

Παράδειγμα 9: Να βρεθούν και να εκτυπωθούν οι διψήφιοι άρτιοι αριθμοί

Αλγόριθμος Παράδειγμα_9

Για i **από** 10 **μέχρι** 98 **με_βήμα** 2

Εκτύπωσε i

Τέλος_Επανάληψης

Τέλος Παράδειγμα_9

Η συμπλήρωση της εντολής Για με το βήμα να είναι 2 αυξάνει την τιμή της μεταβλητής i αντίστοιχα ώστε με αρχική τιμή την τιμή 10 να λαμβάνει τις τιμές 10, 12, 14, 16, 18, 20, ..., 94, 96, 98. Ο επόμενος άρτιος, το 100, δεν είναι διψήφιος και το i δεν θα λάβει την τιμή αυτή

Σημείωση 1: αν επιθυμούσαμε φθίνουσα διάταξη στην εκτύπωση των αριθμών ο βρόχος θα είχε τη μορφή

Για i **από** 98 **μέχρι** 10 **με_βήμα** -2

Εκτύπωσε i

Τέλος_Επανάληψης

Όπου η μεταβλητή i θα λάβει τις τιμές 98, 96, 94, ..., 16, 14, 12, 10.

Σημείωση 2: Η τιμή του βήματος δεν είναι απαραίτητο να είναι ακέραιοι αριθμοί, αλλά μπορεί να λάβει και δεκαδική τιμή

 **Η δομή επανάληψης Όσο...Επανάλαβε**

Όταν δεν είναι καθορισμένο το πλήθος των επαναλήψεων εξ' αρχής τότε δεν μπορεί να χρησιμοποιηθεί η δομή επανάληψης Για που παρουσιάστηκε στην προηγούμενη παράγραφο. Σε αυτήν την περίπτωση χρησιμοποιείται η δομή Όσο...επανάλαβε, η οποία θα περιγραφεί με τη βοήθεια ενός ακόμη παραδείγματος

Παράδειγμα 10: Ο ταμίας σε ένα super market καταχωρεί στην ταμειακή απόδειξη τον κωδικό του προϊόντος και την τιμή. Όταν ολοκληρωθεί η καταχώρηση για την αγορά ενός πελάτη, τότε εισάγεται ο κωδικός 0. Να εκτυπωθεί το συνολικό ποσό της αγοράς και το πλήθος των προϊόντων που αγοράστηκαν

Για την επίλυση του προβλήματος πρέπει να διαβάζονται οι κωδικοί και οι τιμές των προϊόντων μέχρι την εισαγωγή του κωδικού 0

Αλγόριθμος Παράδειγμα_10

σύνολο ← 0

πλήθος ← 0

Διάβασε κωδικός

Όσο κωδικός<>0 **επανάλαβε**

Διάβασε τιμή, τεμάχια

σύνολο ← σύνολο + τιμή * τεμάχια

πλήθος ← πλήθος + 1

Διάβασε κωδικός

Τέλος_Επανάληψης

Εκτύπωσε σύνολο, πλήθος

Τέλος Παράδειγμα_10

Ο βρόχος επαναλαμβάνεται για άγνωστο αριθμό επαναλήψεων.

Η ροή εκτέλεσης μετά την εντολή *Τέλος_επανάληψης* επανέρχεται στην εντολή *Όσο...Επανάλαβε*

Όσο κωδικός<>0 **επανάλαβε**

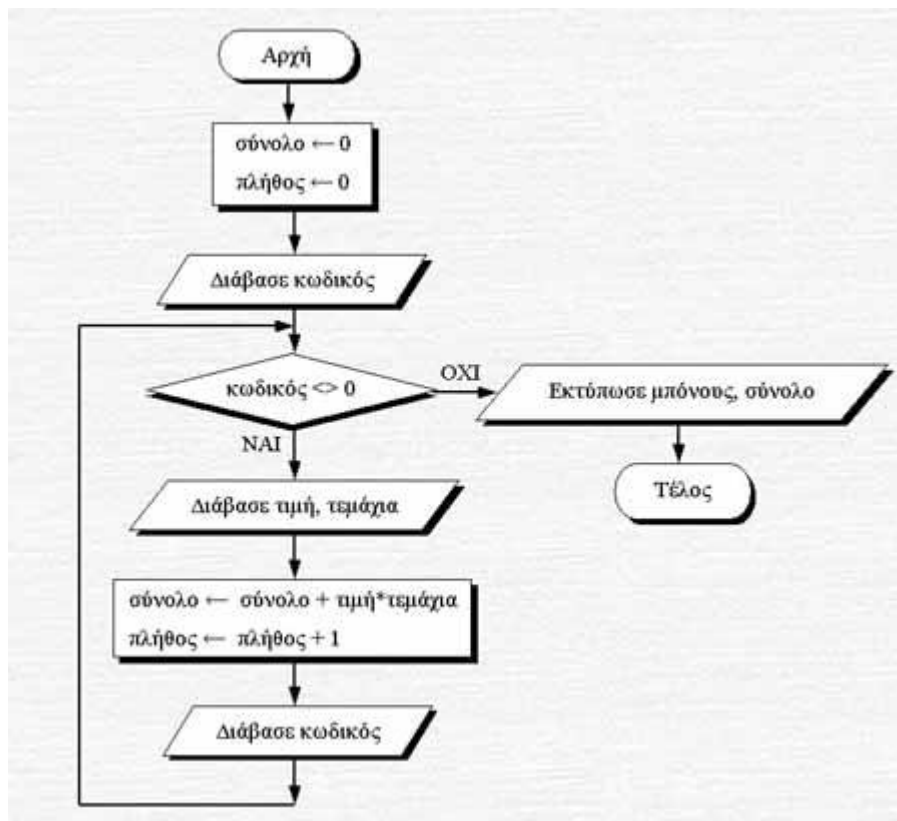
...

Τέλος_Επανάληψης

Η μεταβλητή *σύνολο* περιέχει το σύνολο της αγοράς μέχρι την τρέχουσα επανάληψη. Είναι σαφές ότι δεν είναι γνωστό το πλήθος των προϊόντων που αγοράστηκαν. Για το λόγο αυτό χρησιμοποιείται η μεταβλητή *πλήθος*. Ένα σύνηθες λάθος που κάνουν οι μαθητές είναι ότι ξεχνούν την εντολή *Διάβασε κωδικός* πριν το βρόχο. Ωστόσο, αν δεν διαβαστεί η μεταβλητή *κωδικός* ώστε να λάβει τιμή, πώς θα την χρησιμοποιήσουμε στη συνθήκη του Όσο;

Στο εσωτερικό του βρόχου διαβάζουμε και τις μεταβλητές *τιμή* και *τεμάχια* που αφορούν το προϊόν και προσθέτουμε το γινόμενο τους στη μεταβλητή *σύνολο* (το οποίο έχει αρχικοποιηθεί). Αυξάνουμε δε, τη μεταβλητή *πλήθος* κατά μία μονάδα. Τέλος, διαβάζεται ο κωδικός του νέου προϊόντος, ώστε να ελεγχθεί κατά την νέα επανάληψη

Πρέπει να τονιστεί ότι ο έλεγχος της συνθήκης στη δομή Όσο...επανάλαβε πραγματοποιείται στην αρχή του βρόχου. Το διάγραμμα ροής του παραπάνω αλγορίθμου παρουσιάζεται στη συνέχεια



Παρατηρούμε ότι στο διάγραμμα ροής δεν υπάρχει κάποιο ειδικό σύμβολο για τη δομή Όσο...επανάλαβε αλλά υλοποιείται με τη δημιουργία κύκλου

● Η δομή επανάληψης Αρχή_Επανάληψης...Μέχρις_Ότου

Συμπληρωματικά της δομής Όσο...επανάλαβε χρησιμοποιείται η δομή Αρχή_επανάληψης...Μέχρις_Ότου. Η σύνταξη της δομής είναι:

Αρχή_Επανάληψης

Εντολή1

Εντολή2

Μέχρις_Ότου <συνθήκη>

Παρατηρούμε ότι σε αυτή τη δομή ο έλεγχος της συνθήκης πραγματοποιείται στο τέλος του βρόχου

Παράδειγμα 11: Να γραφεί αλγόριθμος που να διαβάζει αριθμούς από το χρήστη, να υπολογίζει το πλήθος των περιττών αριθμών και να εκτυπώνει το μέσο όρο τους. Η διαδικασία ανάγνωσης να σταματά όταν διαβαστούν 20 περιττοί αριθμοί

Αλγόριθμος Παράδειγμα_11

άθροισμα \leftarrow 0

πλήθος \leftarrow 0

Αρχή_Επανάληψης

Διάβασε αριθμός

Αν αριθμός **mod** 2 = 1 **τότε**

 άθροισμα \leftarrow άθροισμα + αριθμός

 πλήθος \leftarrow πλήθος + 1

Τέλος_Αν

Μέχρις_Ότου πλήθος = 20

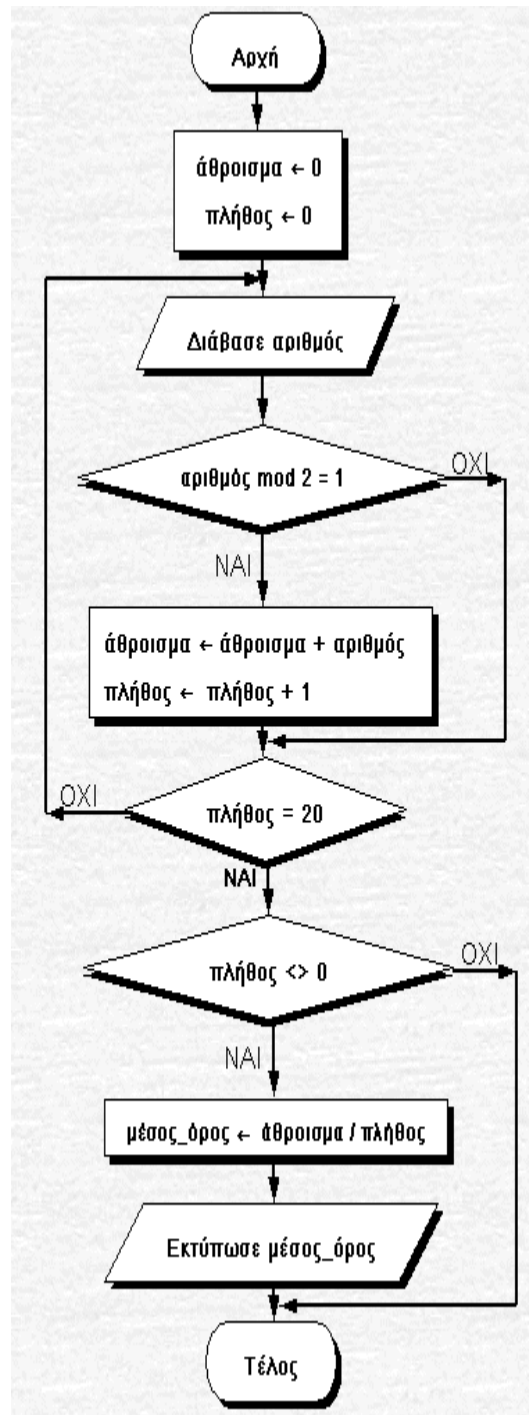
Αν πλήθος \neq 0 **τότε**

 μέσος_όρος \leftarrow άθροισμα / πλήθος

Εκτύπωσε μέσος_όρος

Τέλος_Αν

Τέλος Παράδειγμα_11

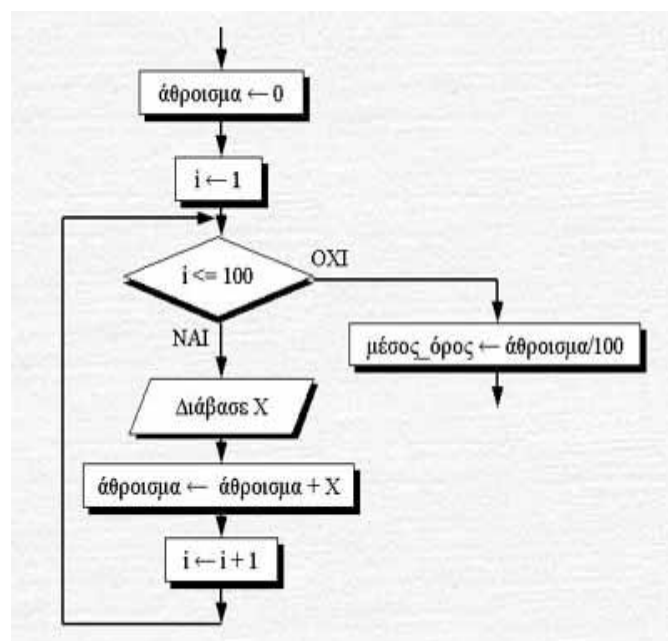


Παρατηρήσεις για τις δομές επανάληψης

Η δομή επανάληψης χρησιμοποιείται όταν είναι γνωστό το πλήθος των επαναλήψεων ενώ οι δομές Όσο...επανάλαβε και Αρχή_Επανάληψης...Μέχρις_Ότου στις άλλες περιπτώσεις.

Ένα ζήτημα που μπορεί να μας απασχολήσει στις ασκήσεις είναι η μετατροπή από την μία δομή στην άλλη. Μια δομή Για πάντα μπορεί να μετατραπεί στις Όσο...επανάλαβε και Αρχή_Επανάληψης...Μέχρις_Ότου, το αντίθετο δεν ισχύει. Η δομή Όσο...επανάλαβε μετατρέπεται πάντα στην Αρχή_Επανάληψης...Μέχρις_Ότου. Ας υποθέσουμε ότι έχουμε το εξής τμήμα αλγορίθμου:

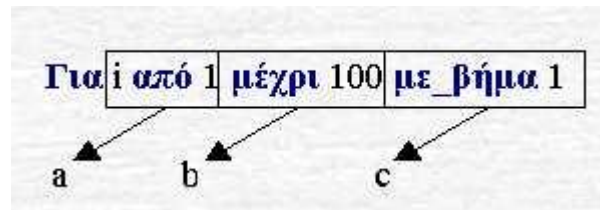
άθροισμα \leftarrow 0
Για i από 1 μέχρι 100
 Διάβασε X
 άθροισμα \leftarrow άθροισμα + X
Τέλος_Επανάληψης
μέσος_όρος \leftarrow άθροισμα / 100



Παρατηρούμε ότι στο διάγραμμα ροής δεν υπάρχει ειδικό σύμβολο για τη δομή *Για*. Η υλοποίησή της απαιτεί τη χρήση μιας σειράς εντολών. Αυτές είναι:

- Η εντολή εκχώρησης $i \leftarrow 1$, για αρχικοποίηση της μεταβλητής
- Η συνθήκη τερματισμού $i \leq 100$
- Η εντολή εκχώρησης $i \leftarrow i + 1$

Και οι τρεις παραπάνω εντολές περιέχονται στην δομή *Για*



Από το διάγραμμα ροής του προηγούμενου παραδείγματος είναι πολύ εύκολο να δημιουργήσουμε το ίδιο βρόχο με τις δομές *Όσο...επανάλαβε* και *Αρχή_Επανάληψης...Μέχρις_Ότου*.

```

άθροισμα ← 0
i ← 1
Όσο i <= 100 επανάλαβε
    Διάβασε X
    άθροισμα ← άθροισμα + X
    i ← i + 1
Τέλος_Επανάληψης
μέσος_όρος ← άθροισμα / 100
    
```

```

άθροισμα ← 0
i ← 1
Αρχή_Επανάληψης
    Διάβασε X
    άθροισμα ← άθροισμα + X
    i ← i + 1
Μέχρις_Ότου i > 100
μέσος_όρος ← άθροισμα / 100
    
```

Παρατηρούμε ότι οι συνθήκες των δύο δομών είναι αντίθετες ενώ το τμήμα εντολών του βρόχου το ίδιο.

Πολλαπλασιασμός αλά Ρωσικά

Ρωσικός πολλαπλασιασμός (παράγραφος 2.4.5 σχολ. βιβλίου)

Ο αλγόριθμος πολλαπλασιασμού ακεραίων αλά ρωσικά με φυσική γλώσσα κατά βήματα, είναι:

| Αλγόριθμος: Πολλαπλασιασμός δύο θετικών ακεραίων (αλά ρωσικά) | |
|---|--|
| Είσοδος: | Δύο ακεραίοι M1 και M2, όπου $M1, M2 \geq 1$ |
| Έξοδος: | Το γινόμενο $P = M1 * M2$ |
| Βήμα 1 | Θέσε $P = 0$ |
| Βήμα 2 | Αν $M2 > 0$, τότε πήγαινε στο Βήμα 3, αλλιώς πήγαινε στο Βήμα 7 |
| Βήμα 3 | Αν ο M2 είναι περιττός, τότε θέσε $P = P + M1$ |
| Βήμα 4 | Θέσε $M1 = M1 * 2$ |
| Βήμα 5 | Θέσε $M2 = M2 / 2$ (θεώρησε μόνο το ακεραίο μέρος) |
| Βήμα 6 | Πήγαινε στο Βήμα 2 |
| Βήμα 7 | Τύπωσε τον P. |

Ο αλγόριθμος σε ψευδοκώδικα για το πρόβλημα του πολλαπλασιασμού αλά ρωσικά, είναι:

Αλγόριθμος Πολλαπλασιασμός_αλά_ρωσικά
Δεδομένα // M1,M2 ακεραίοι //

$P \leftarrow 0$

Όσο $M2 > 0$ **επανάλαβε**

Αν $M2 \bmod 2 = 1$ **τότε**

$P \leftarrow P + M1$

Τέλος_Αν

$M1 \leftarrow M1 * 2$

$M2 \leftarrow M2 \text{ DIV } 2$! ακέραιο μέρος διαίρεσης

Τέλος_επανάληψης

Αποτελέσματα // P, το γινόμενο των ακεραίων M1,M2 //

Τέλος Πολλαπλασιασμός_αλά_ρωσικά

Έστω ότι δίνονται δύο θετικοί ακέραιοι αριθμοί, οι αριθμοί 45 και 19. Οι αριθμοί γράφονται δίπλα-δίπλα και ο πρώτος διπλασιάζεται αγνοώντας το δεκαδικό μέρος, ενώ ο δεύτερος υποδιπλασιάζεται.

Στο διπλανό σχήμα παρουσιάζεται η επαναλαμβανόμενη διαδικασία, που συνεχίζεται μέχρις ότου στη δεύτερη στήλη να προκύψει μονάδα. Τελικώς, το γινόμενο ισούται με το άθροισμα των στοιχείων της πρώτης στήλης, όπου αντίστοιχα στη δεύτερη στήλη υπάρχει περιττός αριθμός. Για το παράδειγμά μας, τα στοιχεία αυτά παρουσιάζονται στην τρίτη στήλη.

| | | |
|------------|----|-----|
| 45 | 19 | 45 |
| 90 | 9 | 90 |
| 180 | 4 | |
| 360 | 2 | |
| 720 | 1 | 720 |
| Άθροισμα = | | 855 |