



Επιστημονικές Εκδόσεις
ΤΖΙΟΛΑ

ΚΕΦΑΛΑΙΟ 6

Δομές Επανάληψης

Πρωτότυπο έργο: Computer Science: A Structured Programming Approach in C, 4th ed., H. Afyouni, B.A.Forouzan. Copyright 2000, 2007, 2023 Cengage Learning, Inc.
Αποκλειστικότητα για την ελληνική γλώσσα: Εκδόσεις TZIOΛA. Copyright 2024.
Με επιφύλαξη παντός νόμιμου δικαιώματος.

ΘΕΜΑΤΑ ΚΕΦΑΛΑΙΟΥ

- Βασικές έννοιες των βρόχων
- Βρόχοι με συνθήκη εισόδου έναντι των βρόχων με συνθήκη εξόδου
- Αρχικοποίηση και ενημέρωση ενός βρόχου
- Βρόχοι που ελέγχονται από συμβάντα
- Βρόχοι που ελέγχονται από μετρητή
- Χρήση των εντολών while, for ή do ... while για δημιουργία βρόχων σε προγράμματα

ΘΕΜΑΤΑ ΚΕΦΑΛΑΙΟΥ

- Επιλογή της καταλληλότερης δομή βρόχου για ένα δεδομένο πρόβλημα
- Χρήση των εντολών break, continue και goto σε βρόχους
- Τέσσερις τυπικές εφαρμογές των βρόχων:
 - υπολογισμός αθροίσματος
 - υπολογισμός γινομένου
 - εύρεση μικρότερου ή μεγαλύτερου
 - «ερωτήματα» αναζήτησης δεδομένων

6.1

Η έννοια του βρόχου

Η έννοια του βρόχου (1/2)

- Η πραγματική ισχύς των υπολογιστών έγκειται στην ικανότητά τους να επαναλαμβάνουν μία εργασία πολλές φορές
 - Μπορεί να είναι μία ενέργεια, ή μία αλληλουχία ενεργειών
 - Αυτή η δομή επανάληψης αποκαλείται **βρόχος** (loop) και αποτελεί μία από τις βασικές έννοιες του δομημένου προγραμματισμού



ΣΧΗΜΑ 6-1 Σχηματική αναπαράσταση της έννοιας του βρόχου.

Η έννοια του βρόχου (2/2)

- Συνθήκη ελέγχου του βρόχου
 - Οι δομές βρόχων επιτρέπουν τον έλεγχο της συνθήκης είτε πριν, είτε μετά από κάθε επανάληψη του βρόχου.
 - Θα πρέπει να σχεδιάζουμε τον βρόχο με τρόπο που να ελέγχει εάν η εργασία έχει ολοκληρωθεί, είτε πριν είτε μετά από κάθε επανάληψη
 - Εάν η εργασία δεν έχει ολοκληρωθεί, ο βρόχος επαναλαμβάνεται για άλλη μία φορά· εάν έχει ολοκληρωθεί, ο βρόχος τερματίζεται

6.2

**Βρόχοι με συνθήκη εισόδου
και βρόχοι με συνθήκη εξόδου**

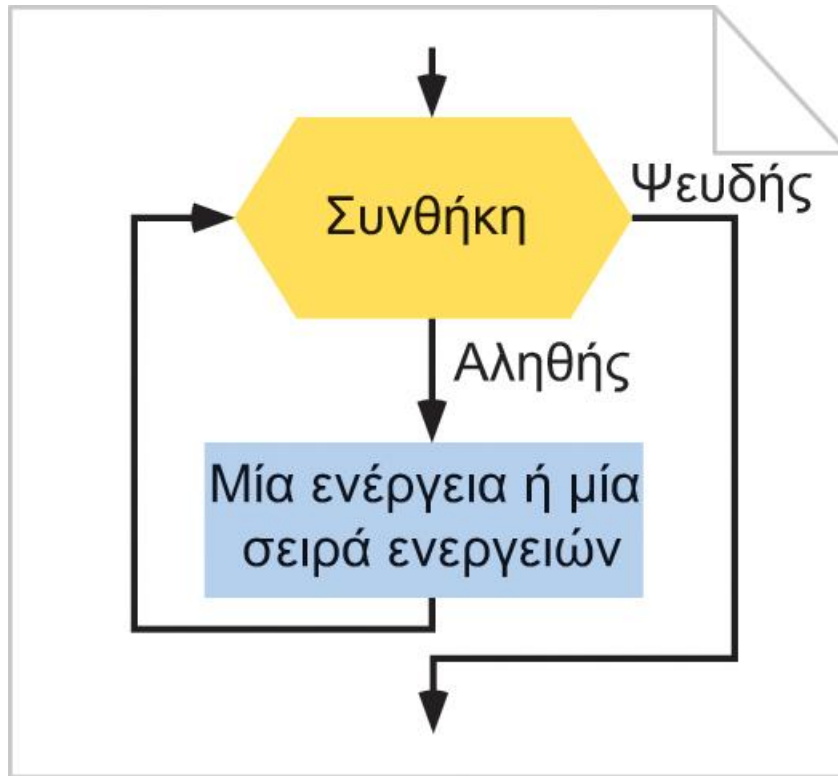
Βρόχοι με συνθήκη εισόδου

- Βρόχος με συνθήκη εισόδου (pre-test loop): η συνθήκη ελέγχεται κατά την έναρξη του βρόχου και στην αρχή κάθε επανάληψής του
 - Εάν η συνθήκη ελέγχου του βρόχου αποτιμηθεί ως αληθής (true), εκτελείται ο κώδικας που αποτελεί το σώμα του βρόχου
 - Εάν η συνθήκη αποτιμηθεί ως ψευδής (false), ο βρόχος τερματίζεται

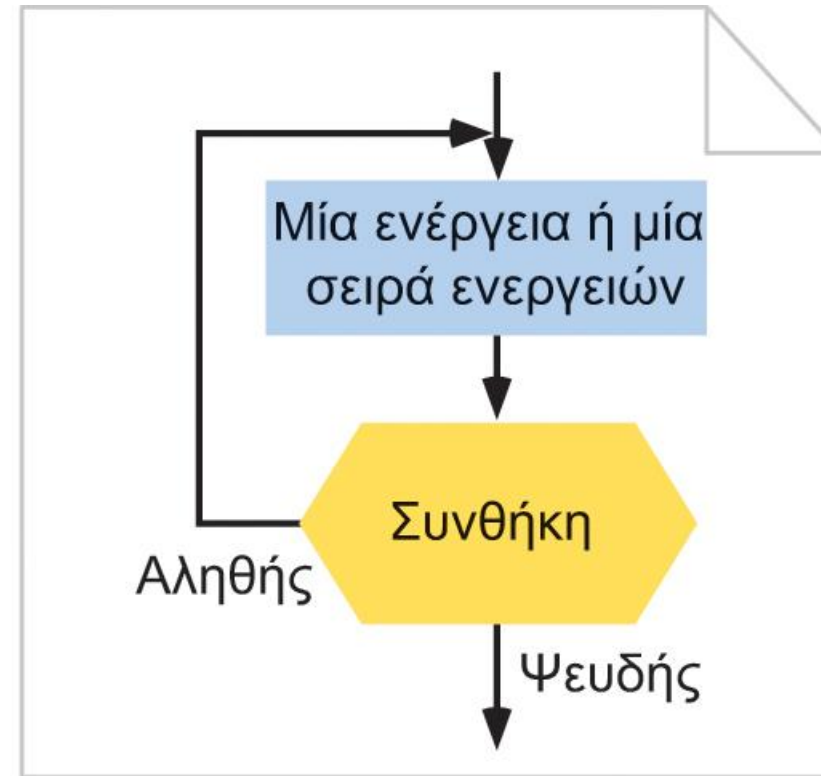
Βρόχοι με συνθήκη εξόδου

- Βρόχος με συνθήκη εξόδου (post-test loop): εκτελεί πάντα την ενέργεια που περιλαμβάνεται στο σώμα του βρόχου, τουλάχιστον μία φορά
 - Στη συνέχεια, ελέγχεται η συνθήκη τερματισμού
 - Εάν η συνθήκη αποτιμηθεί ως αληθής, ο βρόχος επαναλαμβάνεται
 - Εάν η συνθήκη αποτιμηθεί ως ψευδής, ο βρόχος τερματίζεται

Βρόχοι με συνθήκη εισόδου και συνθήκη εξόδου

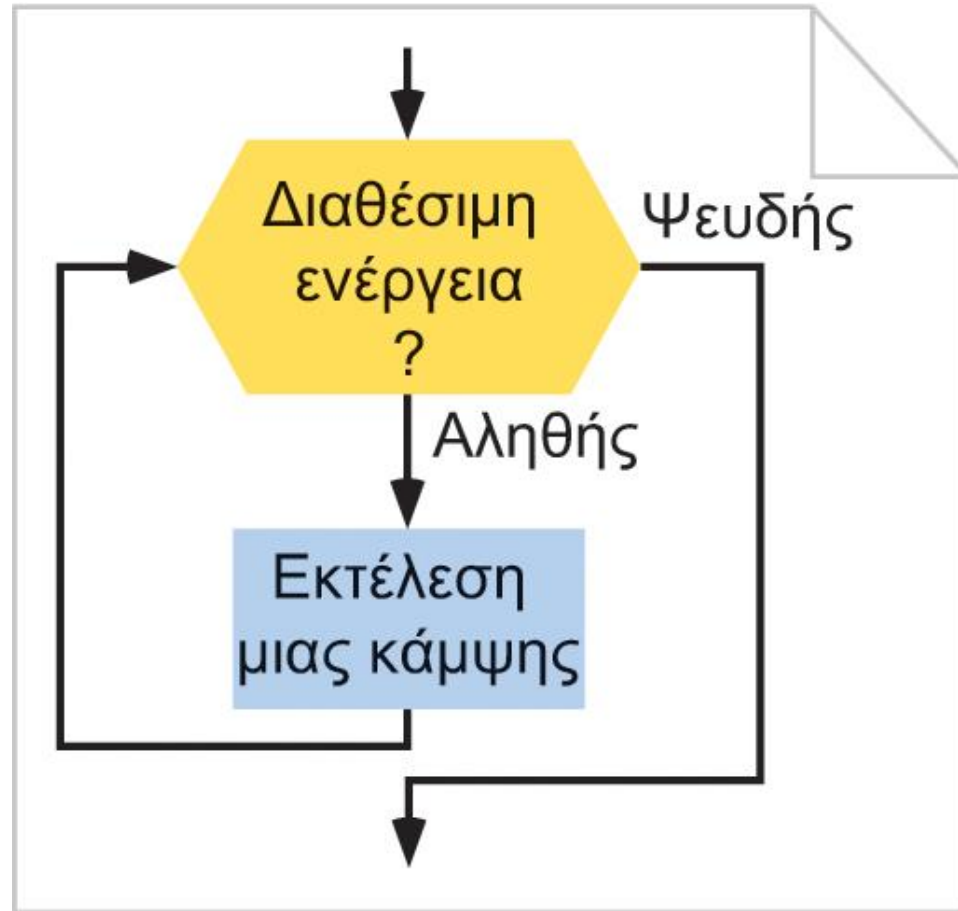


(α) Βρόχος με συνθήκη εισόδου

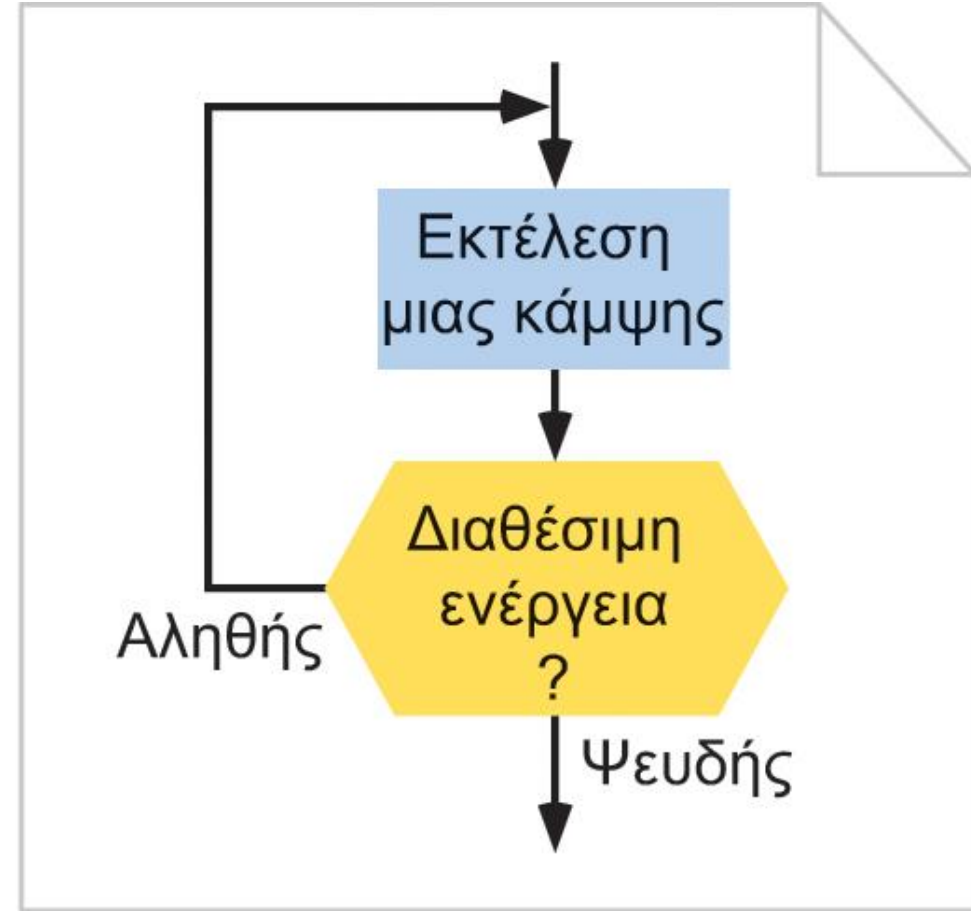


(β) Βρόχος με συνθήκη εξόδου

ΣΧΗΜΑ 6-2 Διαγράμματα ροής για βρόχους με συνθήκη εισόδου και συνθήκη εξόδου.

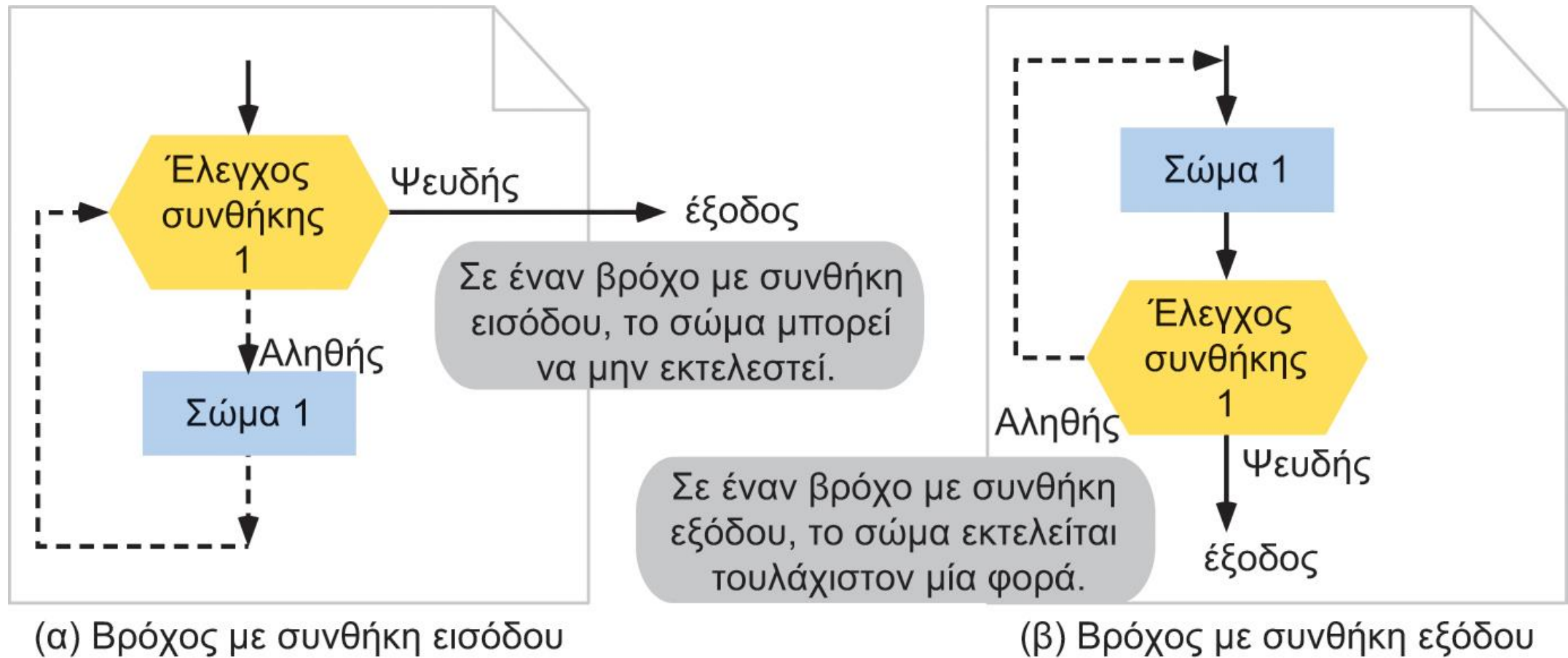


(α) Βρόχος με συνθήκη εισόδου



(β) Βρόχος με συνθήκη εξόδου

ΣΧΗΜΑ 6-3 Δύο στρατηγικές για την εκτέλεση του προγράμματος προσωπικής εκγύμνασης.



ΣΧΗΜΑ 6-4 Ελάχιστος αριθμός επαναλήψεων σε δύο τύπους βρόχων:
(α) με συνθήκη εισόδου και (β) με συνθήκη εξόδου.

6.3

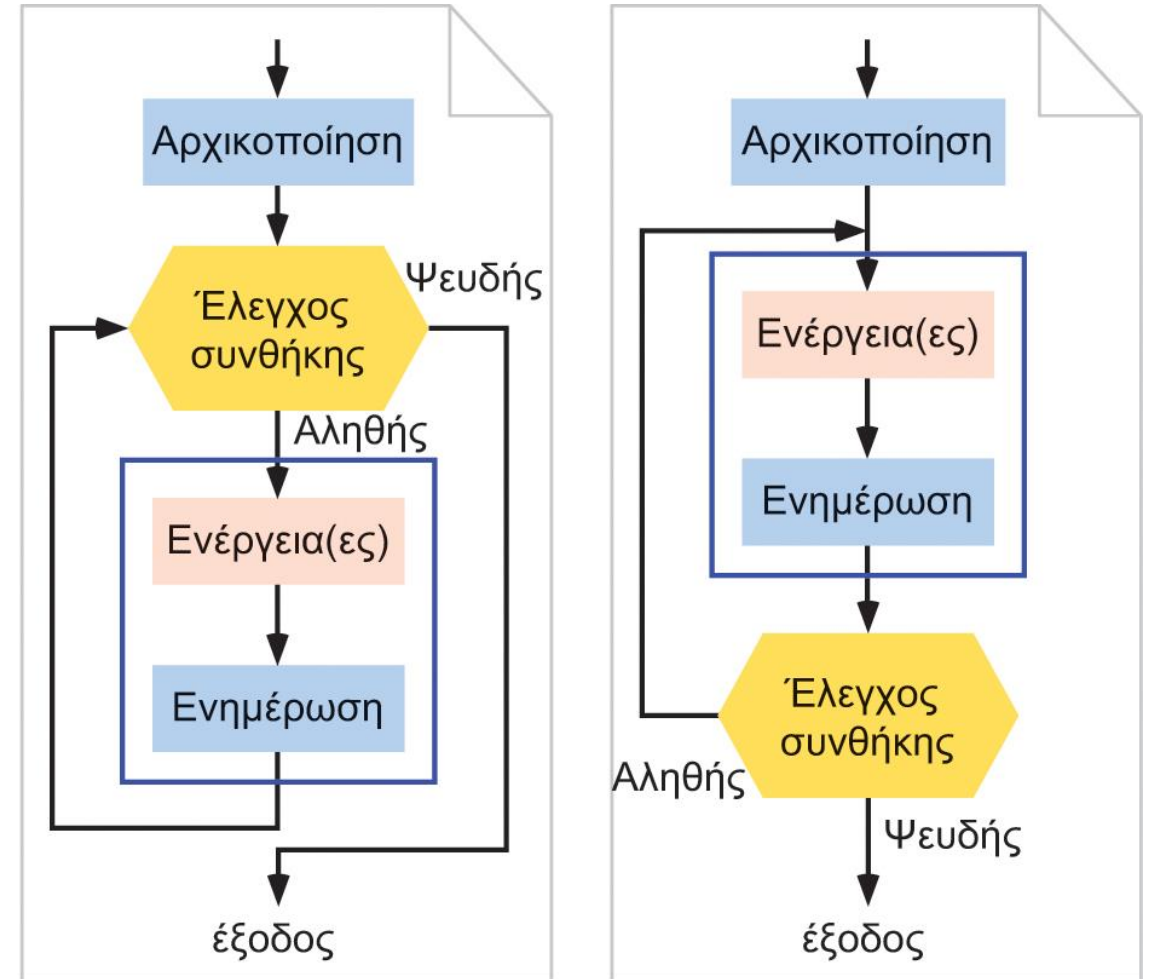
Αρχικοποίηση και ενημέρωση

Αρχικοποίηση ενός βρόχου

- Πριν μπορέσει να ξεκινήσει η εκτέλεση ενός βρόχου, απαιτείται συνήθως κάποια προετοιμασία
- Η αρχικοποίηση πρέπει να γίνει πριν από την πρώτη εκτέλεση του σώματος του βρόχου.
 - «Προετοιμάζει το έδαφος» για τις ενέργειες του βρόχου
 - Με τη ρητή αρχικοποίηση, περιλαμβάνουμε κώδικα για τον ορισμό των αρχικών τιμών των βασικών μεταβλητών του βρόχου
 - Με την έμμεση αρχικοποίηση δεν χρησιμοποιούμε ειδικό κώδικα αρχικοποίησης· αντ' αυτού, βασιζόμαστε σε μία προϋπάρχουσα κατάσταση-συνθήκη για τον έλεγχο του βρόχου

Ενημέρωση του βρόχου

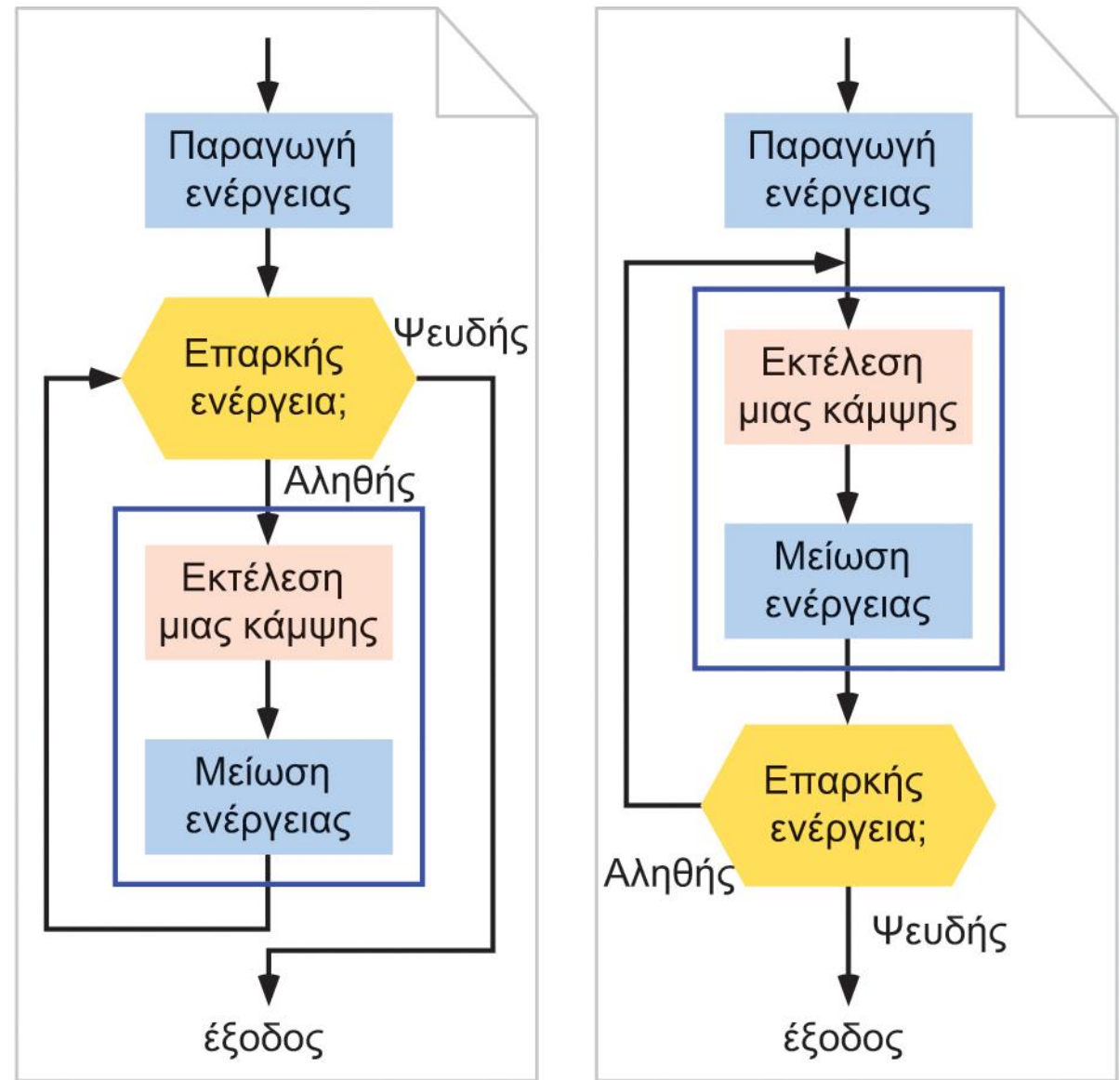
- Η ενημέρωση πραγματοποιείται, συνήθως ως τελευταία ενέργεια, σε κάθε επανάληψη του βρόχου
 - Εάν το σώμα του βρόχου επαναλαμβάνεται m φορές, τότε η ενημέρωση πραγματοποιείται επίσης m φορές
- Ένας βρόχος που δεν τερματίζεται ποτέ αποκαλείται ατέρμων ή άενσος βρόχος (infinite loop)



(α) Βρόχος με συνθήκη εισόδου (β) Βρόχος με συνθήκη εξόδου

ΣΧΗΜΑ 6-5 Αρχικοποίηση και ενημέρωση βρόχου.

ΣΧΗΜΑ 6-6 Αρχικοποίηση και ενημέρωση βρόχου στο πρόγραμμα προσωπικής εκγύμνασης.



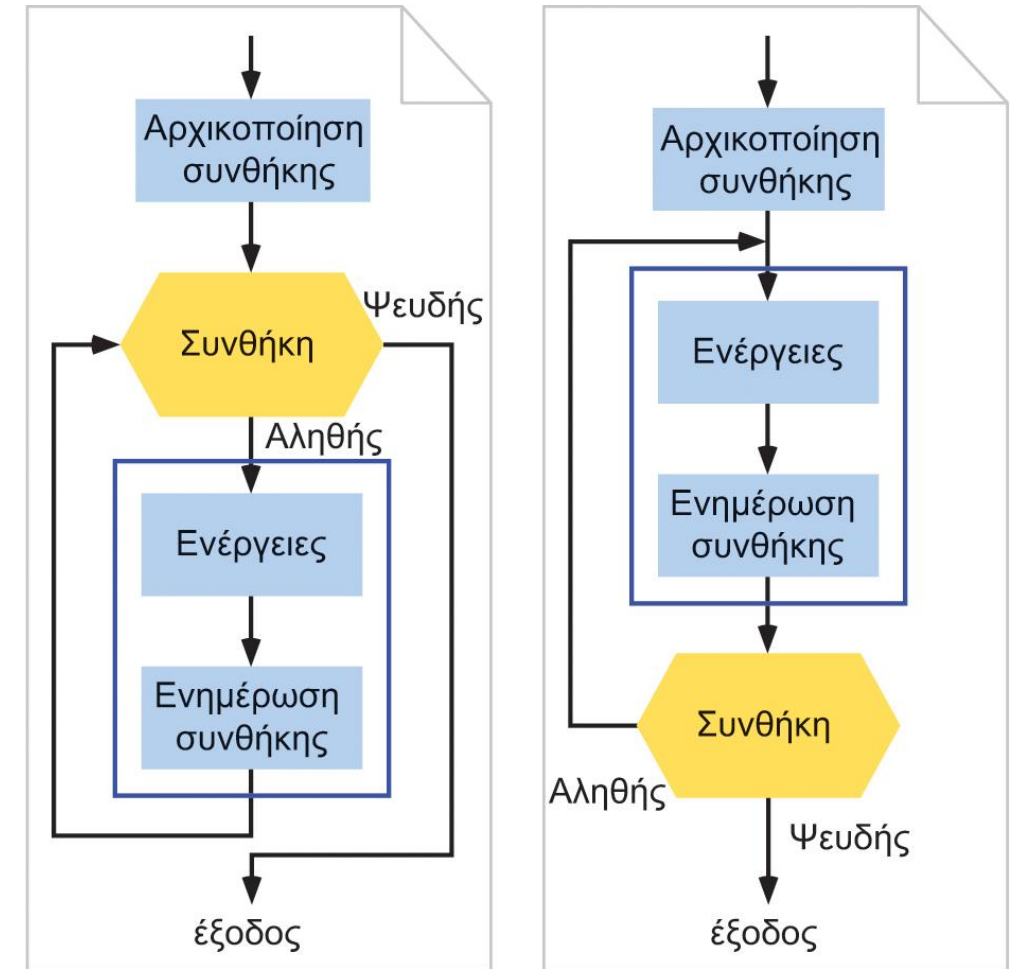
(α) Βρόχος με συνθήκη εισόδου (β) Βρόχος με συνθήκη εξόδου

6.4

**Βρόχοι ελεγχόμενοι από συμβάν
και βρόχοι ελεγχόμενοι με μετρητή**

Βρόχοι ελεγχόμενοι από συμβάν

- Σε έναν βρόχο ελεγχόμενο από συμβάν, ένα συμβάν μεταβάλλει την συνθήκη ελέγχου του βρόχου από αληθή σε ψευδή
- Σε βρόχους ελεγχόμενους από συμβάντα, η διεργασία ενημέρωσης μπορεί να είναι ρητή ή έμμεση
 - Εάν είναι ρητή, ελέγχεται από τον βρόχο
 - Εάν είναι έμμεση, τότε ελέγχεται από κάποια εξωτερική συνθήκη



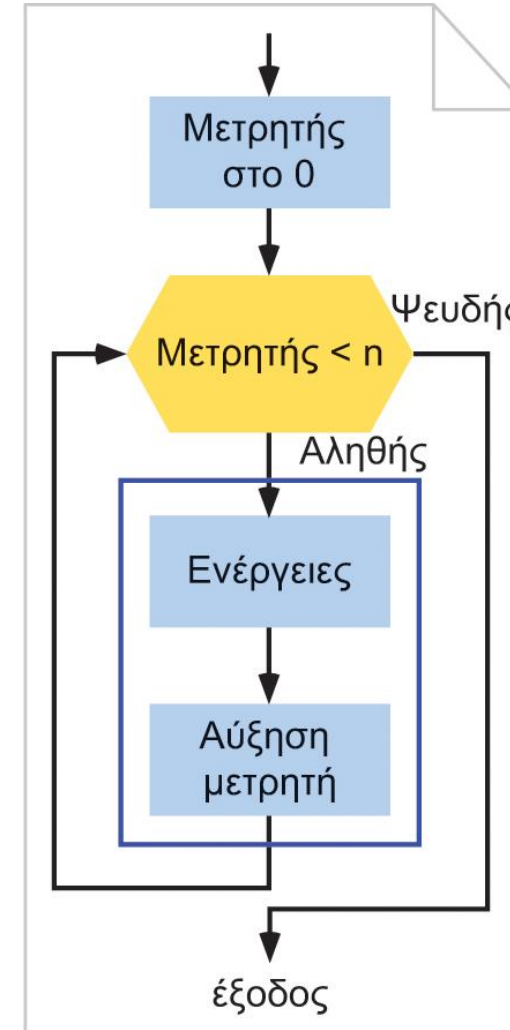
(α) Βρόχος με συνθήκη εισόδου (β) Βρόχος με συνθήκη εξόδου

ΣΧΗΜΑ 6-7 Διαγράμματα ροής για βρόχους ελεγχόμενους από συμβάντα.

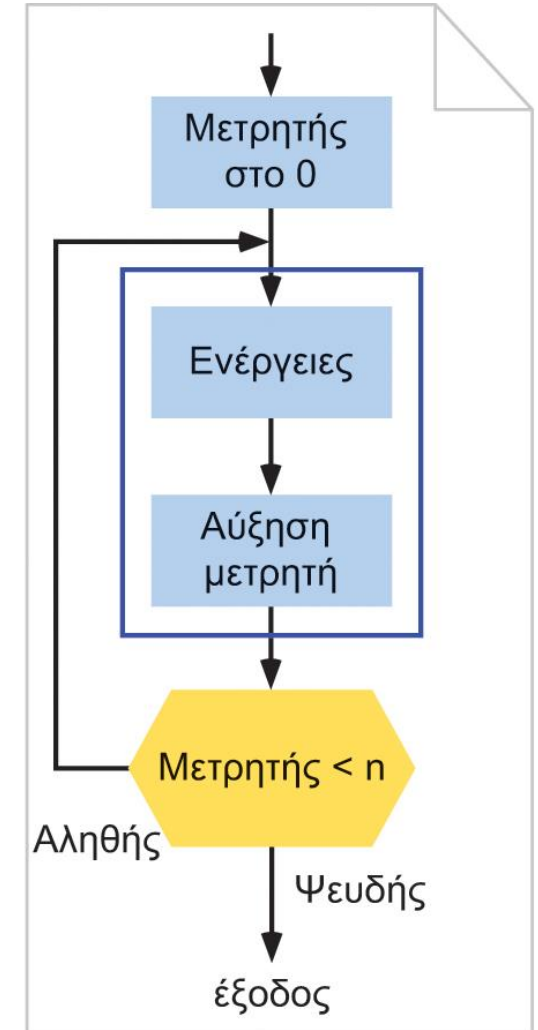
Βρόχοι ελεγχόμενοι από μετρητή

- Χρησιμοποιούμε έναν βρόχο ελεγχόμενο από μετρητή όταν γνωρίζουμε επακριβώς πόσες φορές πρέπει να επαναληφθεί μία ενέργεια.

ΣΧΗΜΑ 6-8 Διαγράμματα ροής για βρόχους ελεγχόμενους από μετρητή.



(α) Βρόχος με συνθήκη εξόδου



(β) Βρόχος με συνθήκη εξόδου

Σύγκριση μεταξύ βρόχων με συνθήκη εισόδου και συνθήκη εξόδου

ΠΙΝΑΚΑΣ 6-1 Συγκριτική παρουσίαση βρόχων με συνθήκη εισόδου και συνθήκη εξόδου

ΒΡΟΧΟΣ ΜΕ ΣΥΝΘΗΚΗ ΕΙΣΟΔΟΥ		ΒΡΟΧΟΣ ΜΕ ΣΥΝΘΗΚΗ ΕΞΟΔΟΥ	
Αρχικοποίηση:	1	Αρχικοποίηση:	1
# ελέγχων συνθήκης τερματισμού:	$n + 1$	# ελέγχων συνθήκης τερματισμού:	n
Εκτέλεση σώματος βρόχου:	n	Εκτέλεση σώματος βρόχου:	n
Ενημέρωση βρόχου:	n	Ενημέρωση βρόχου:	n
Ελάχιστος αρ. επαναλήψεων:	0	Ελάχιστος αρ. επαναλήψεων:	1

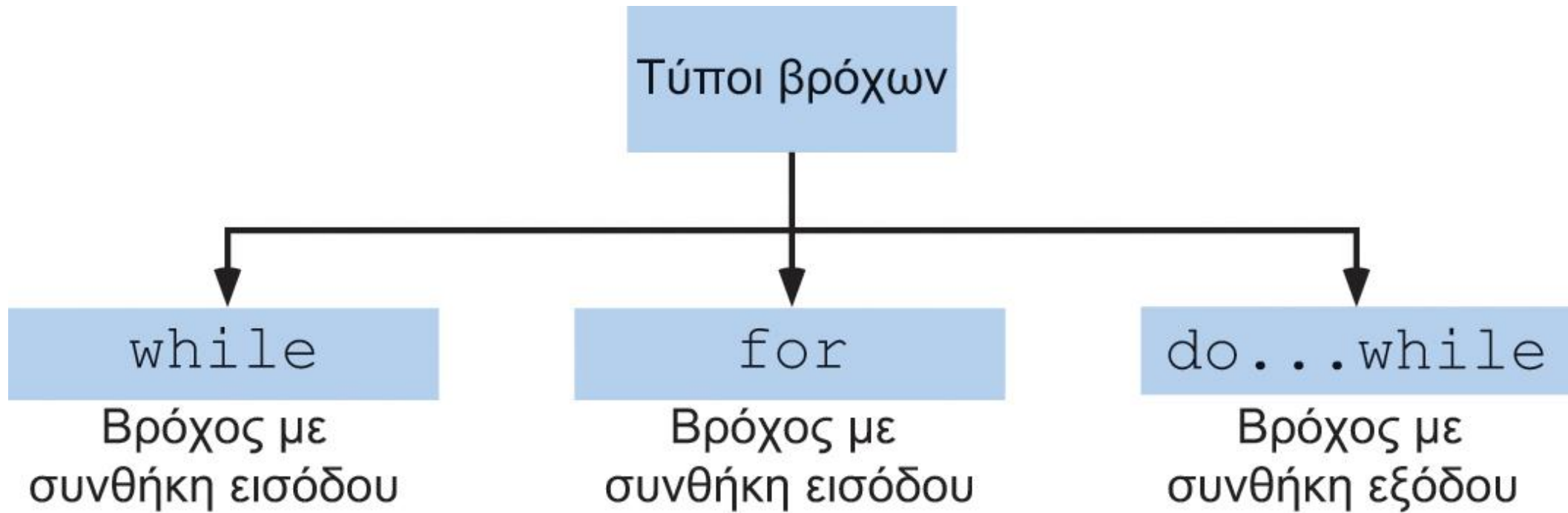
6.5

Βρόχοι στη C

Βρόχοι στη C

- Η γλώσσα C διαθέτει τρεις εντολές για τη δημιουργία βρόχων:
 - `while`
 - `for`
 - `do ... while`
 - Και με τις τρεις δομές, η εκτέλεση του βρόχου συνεχίζεται για όσο χρόνο η ελεγχόμενη συνθήκη αποτιμάται ως αληθής και τερματίζεται όταν η συνθήκη αποτιμηθεί ως ψευδής

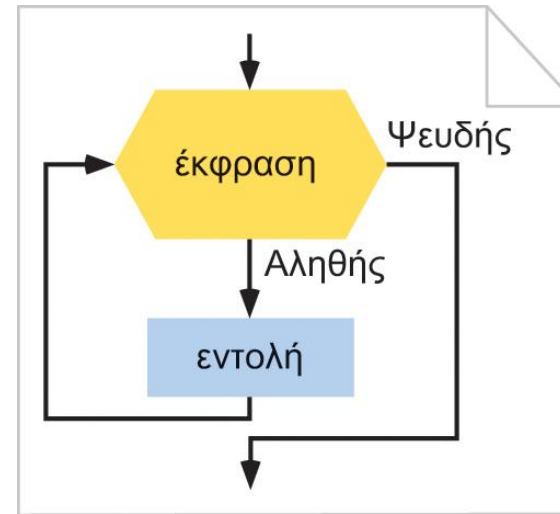
Βρόχοι στη C



ΣΧΗΜΑ 6-9 Δομές βρόχων της C.

Ο βρόχος `while`

- Ανήκει στην κατηγορία των βρόχων με συνθήκη εισόδου
 - Χρησιμοποιεί μία έκφραση, τη συνθήκη τερματισμού, για τον έλεγχο της εκτέλεσης του βρόχου
 - Επειδή είναι βρόχος με συνθήκη εισόδου, αυτή η έκφραση (συνθήκη) ελέγχεται πριν από κάθε επανάληψη του βρόχου
- Κοινές χρήσεις: έλεγχος διεργασιών, λειτουργίες ανάγνωσης αρχείων και εκτύπωσης



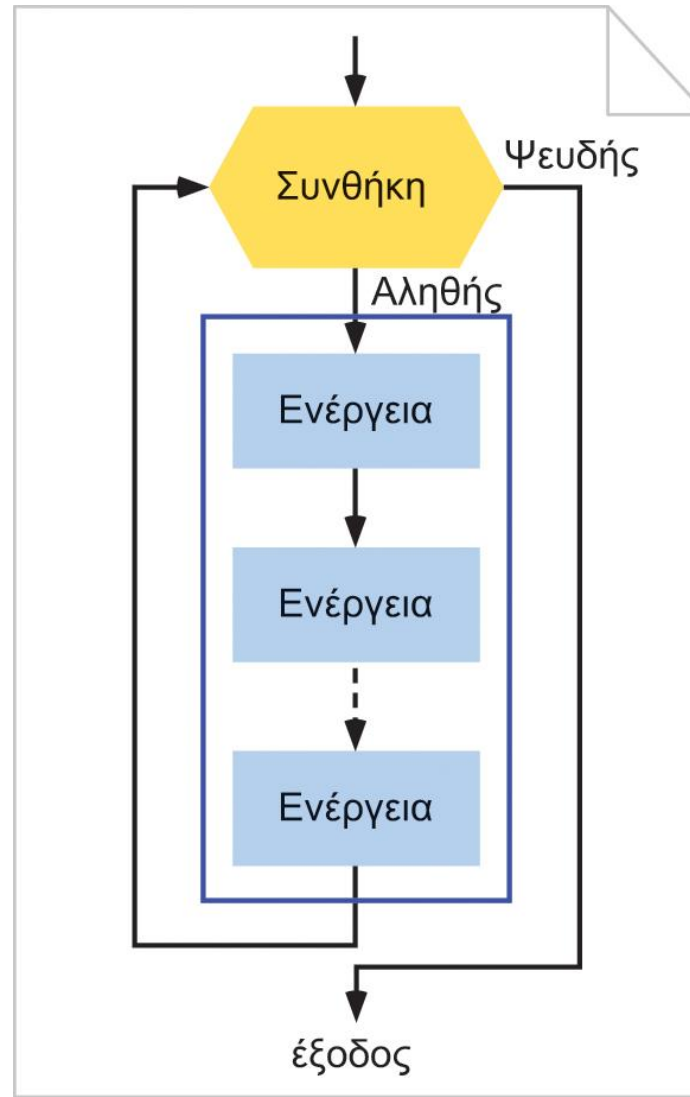
(α) Διάγραμμα ροής

```
while (έκφραση)  
    εντολή
```

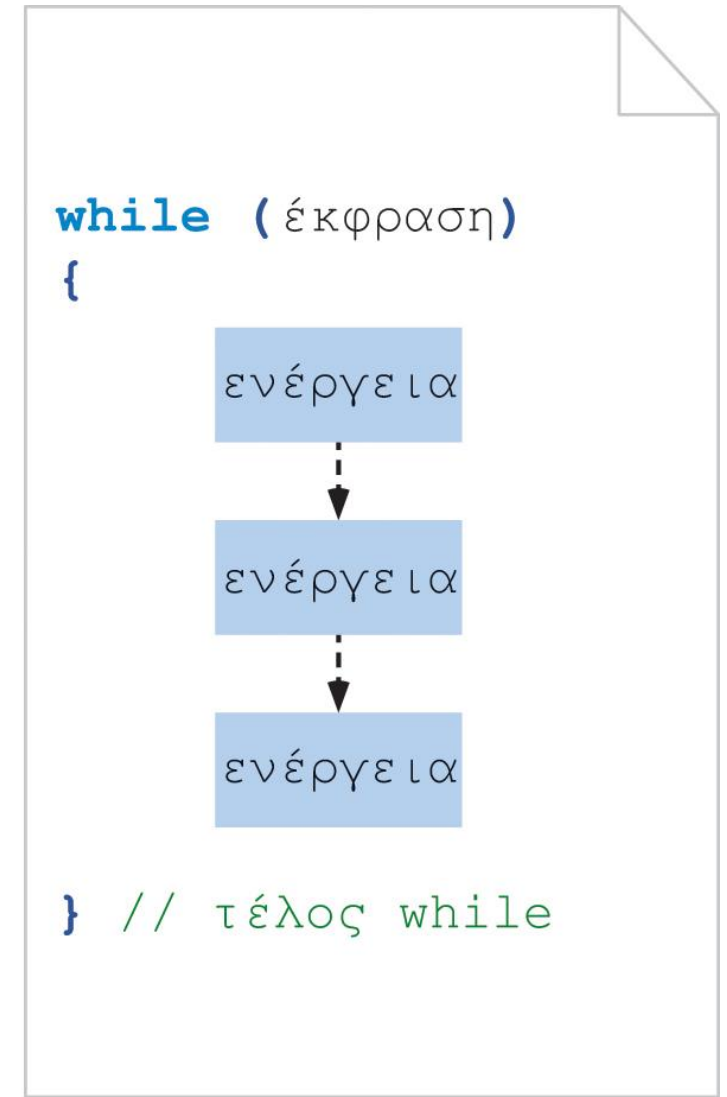
(β) Δείγμα κώδικα

ΣΧΗΜΑ 6-10 Δείγμα σύνταξης της εντολής `while` και το αντίστοιχο διάγραμμα ροής.

ΣΧΗΜΑ 6-11 Χρήση σύνθετης εντολής στο σώμα του βρόχου while.



(α) Διάγραμμα ροής



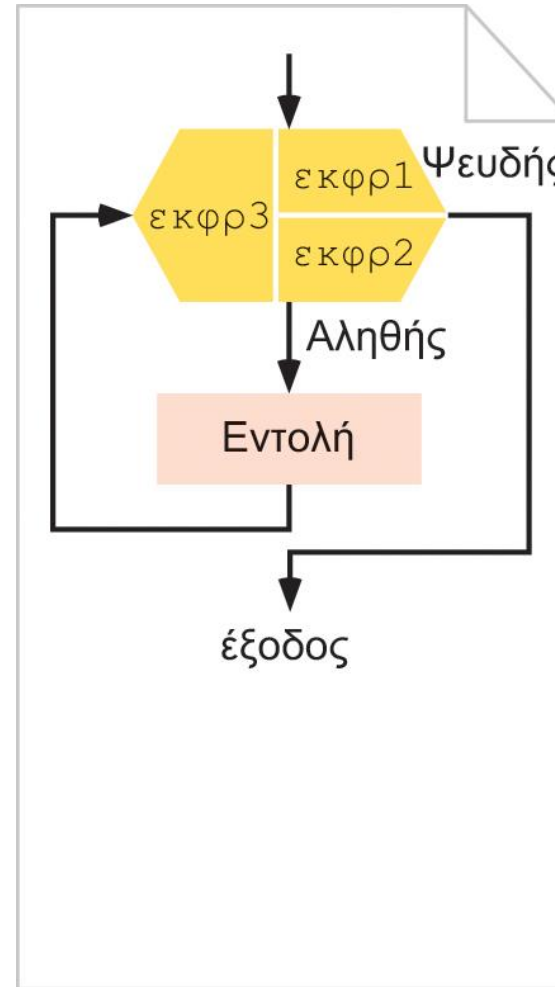
(β) Γλώσσα C

Ο βρόχος for (1/4)

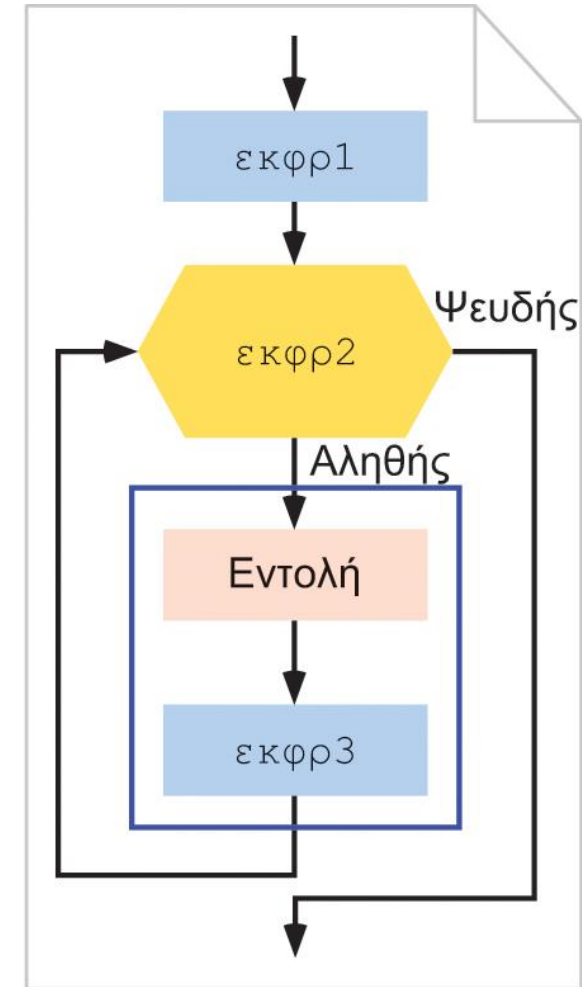
- Η εντολή for χρησιμοποιείται για τη δημιουργία βρόχων με συνθήκη εισόδου
Η σύνταξη της εντολής περιλαμβάνει τρεις εκφράσεις
 - Η πρώτη έκφραση περιέχει τις απαραίτητες εντολές αρχικοποίησης
 - Η δεύτερη έκφραση περιέχει τη συνθήκη τερματισμού του βρόχου
 - Η τρίτη έκφραση είναι η έκφραση ενημέρωσης
- Ένθεση βρόχων for
 - Στο σώμα ενός βρόχου for μπορεί να χρησιμοποιηθεί οποιαδήποτε εντολή, ακόμη και ένας άλλος βρόχος for

Ο βρόχος for (2/4)

ΣΧΗΜΑ 6-12 Δείγμα σύνταξης και διαγράμματα ροής για την εντολή for.



(α) Διάγραμμα ροής

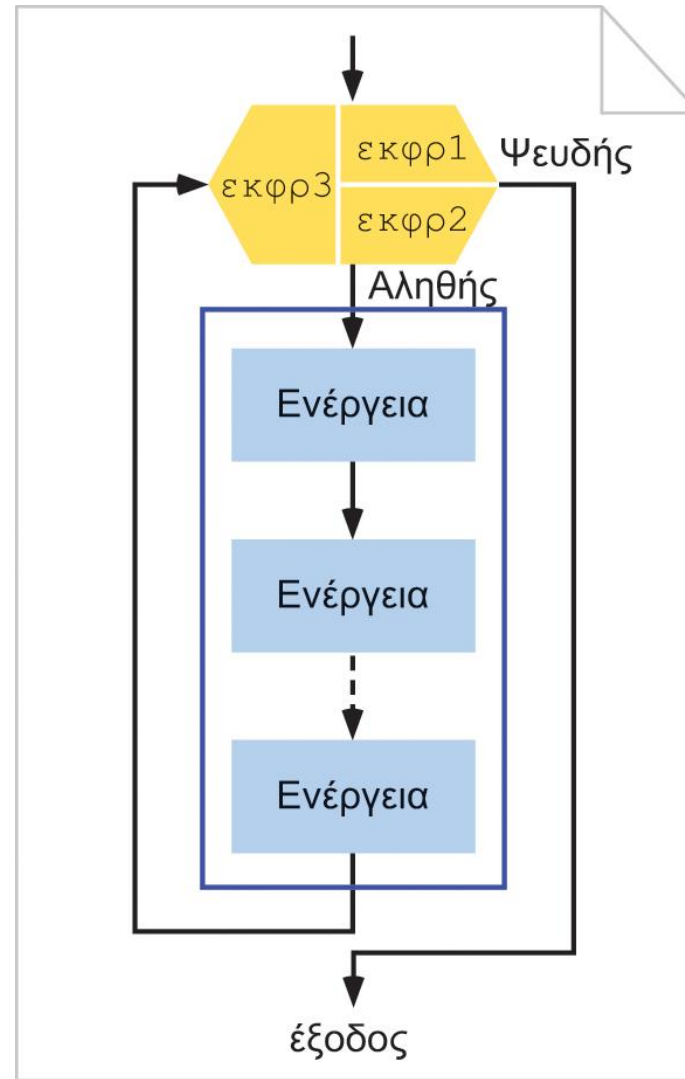


(β) Διευρυμένο διάγραμμα ροής

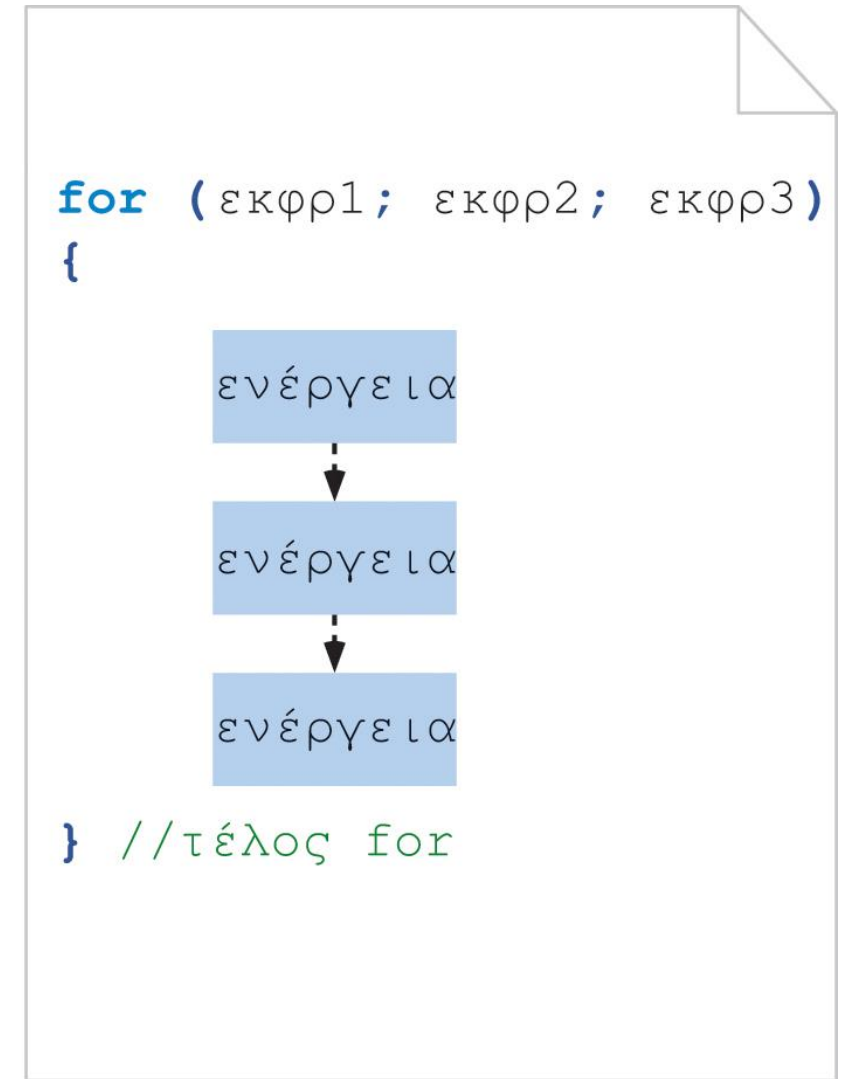
```
for ( εκφρ1 ; εκφρ2 ; εκφρ3 )  
    εντολή ;
```

Ο βρόχος for (3/4)

ΣΧΗΜΑ 6-13 Δείγμα σύνταξης και διάγραμμα ροής βρόχου for με σύνθετη εντολή.

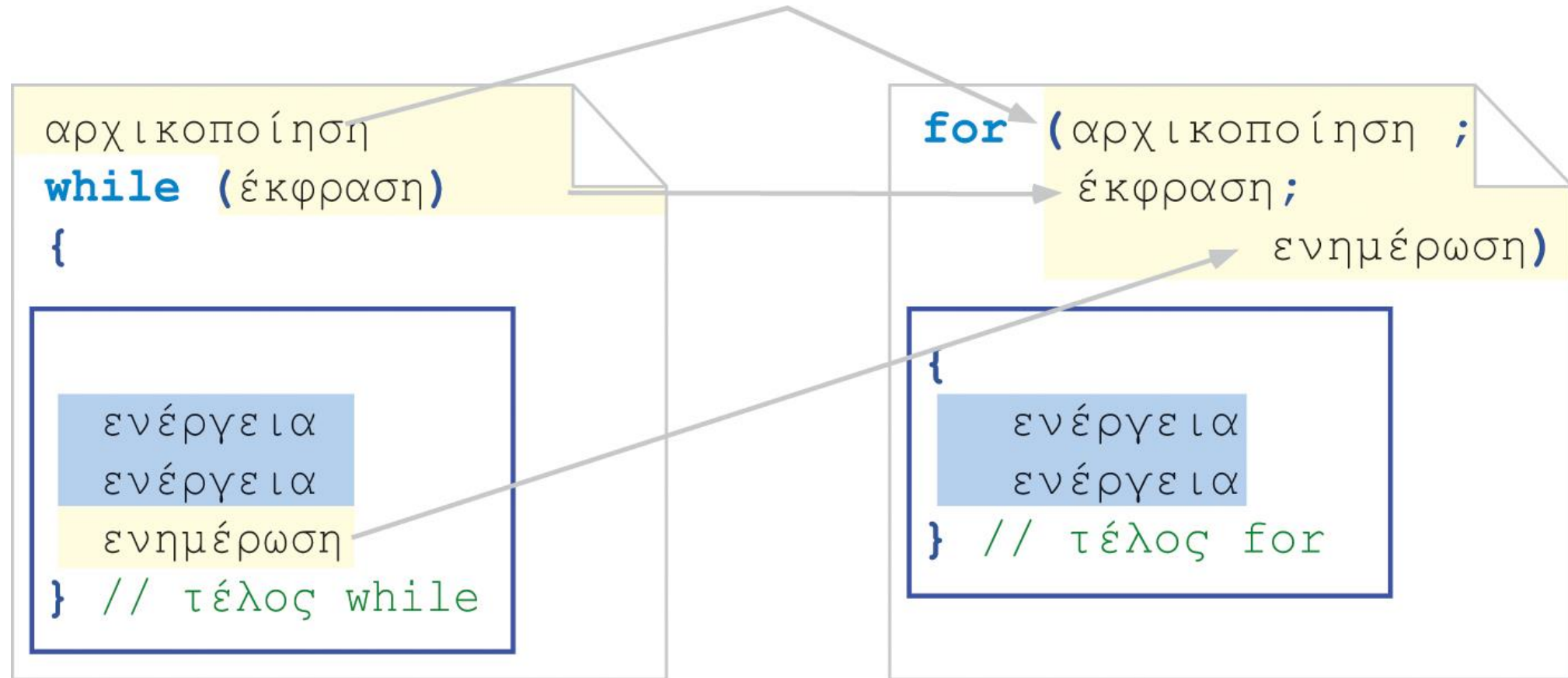


(α) Διάγραμμα ροής



(β) Γλώσσα C

Ο βρόχος for (4/4)

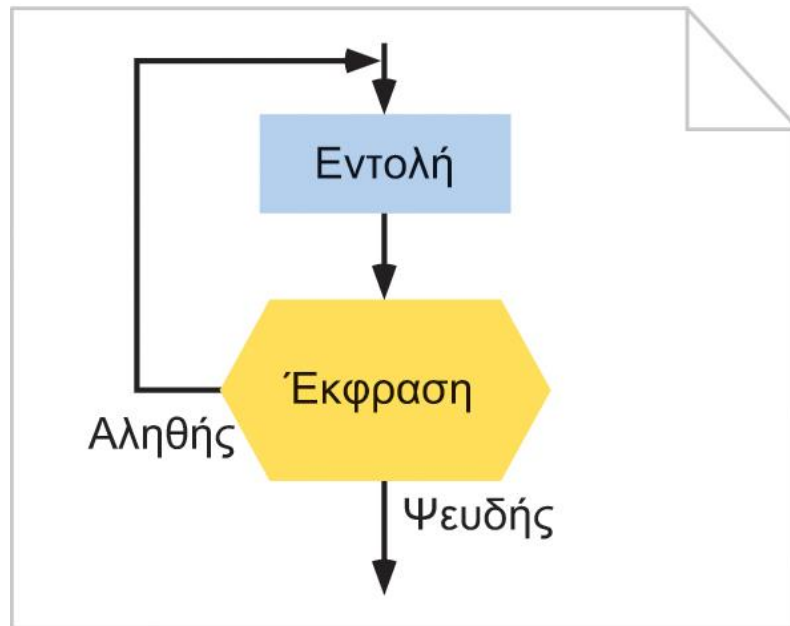


ΣΧΗΜΑ 6-14 Συγκριτική παρουσίαση βρόχων for και while.

Ο βρόχος `do ... while`

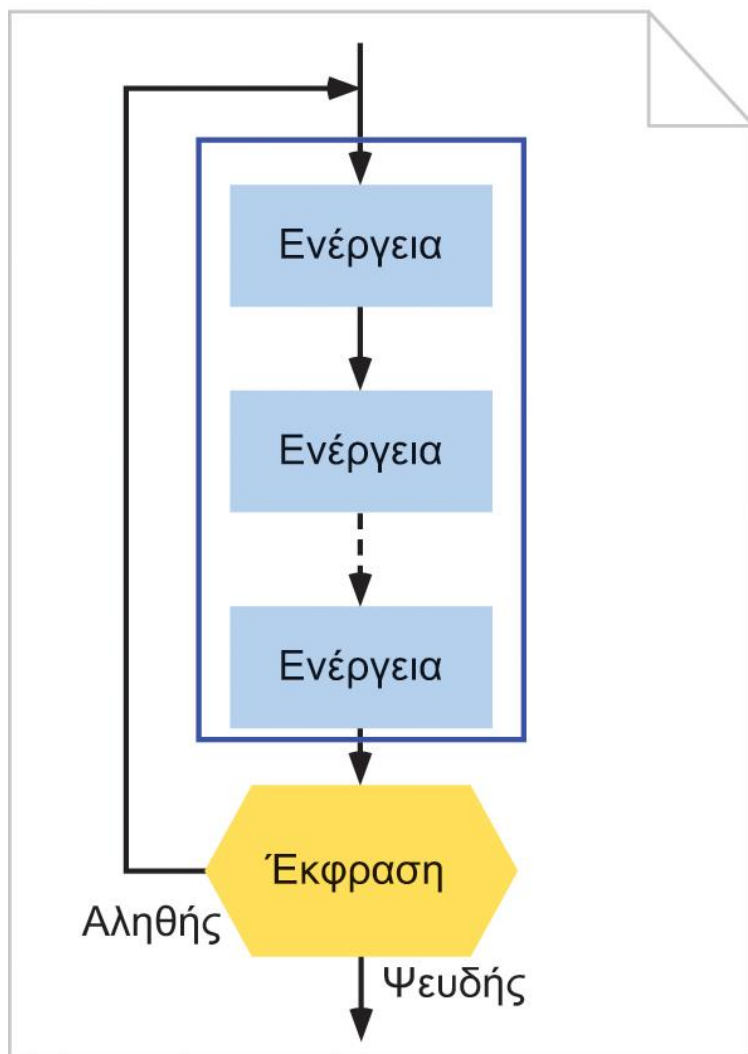
- Χρησιμοποιείται για τη δημιουργία βρόχων με συνθήκη εξόδου
 - Η συνθήκη που ελέγχει την εκτέλεση του βρόχου ελέγχεται αφού εκτελεστεί το σώμα εντολών του βρόχου

Διάγραμμα ροής



Δείγμα κώδικα

```
do  
    εντολή  
while ( έκφραση );
```



```
do
{
    ενέργεια
    ενέργεια
    ...
    ενέργεια
} while (έκφραση);
```

ΣΧΗΜΑ 6-15 Παραδείγματα σύνταξης και αντίστοιχα διαγράμματα ροής για την εντολή do ... while.

Συνθήκη εισόδου:
δεν εκτυπώνεται τίποτα

```
while (false)
{
    printf("Hello World");
} // τέλος while
```

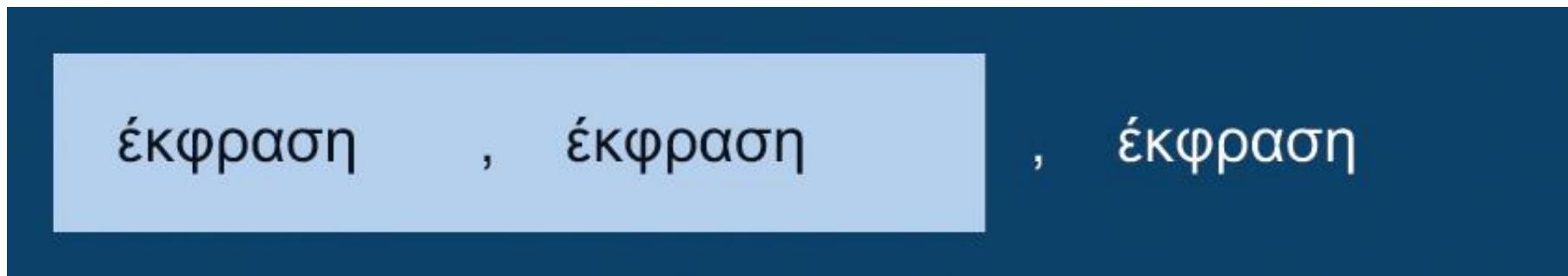
```
do
{
    printf("Hello World");
} while (false);
```

Συνθήκη εξόδου:
εκτυπώνεται το Hello World

ΣΧΗΜΑ 6-16 Επίδραση της θέσης όπου πραγματοποιείται ο έλεγχος της συνθήκης τερματισμού στους βρόχους while και do ... while.

Σύνθετες εκφράσεις με χρήση του τελεστή κόμμα

- Μία σύνθετη έκφραση αποτελείται από δύο ή περισσότερες εκφράσεις, οι οποίες διαχωρίζονται μεταξύ τους με τον **τελεστή κόμμα** (,)
- Αν και η C επιτρέπει τη χρήση σύνθετων εκφράσεων σε πολλές θέσεις, συνηθέστερα χρησιμοποιούνται σε εντολές for
- Ως προς τη σειρά αποτίμησης, ο τελεστής κόμμα έχει τη χαμηλότερη προτεραιότητα 1 απ' όλους τους τελεστές της C



ΣΧΗΜΑ 6-17 Ένθεση σύνθετων εκφράσεων.

6.6

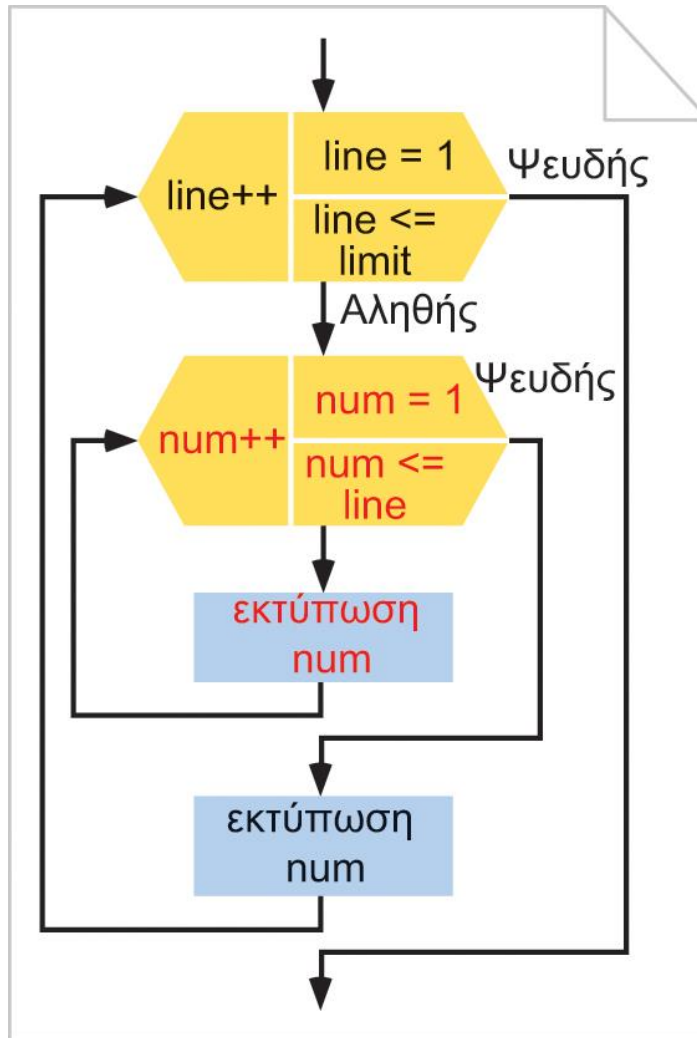
Παραδείγματα βρόχων

Παραδείγματα βρόχων (1/2)

- Βρόχος for: Υπολογισμός ανατοκίζόμενης απόδοσης μιας επένδυσης
 - Υπολογισμός της απόδοσης του κεφαλαίου στην πάροδο του χρόνου
- Ένθετοι βρόχοι for: Εκτύπωση τριγωνικού σχηματισμού ακεραίων
 - Εκτυπώνει μία σειρά αριθμών σε σχηματισμό ορθογώνιου τριγώνου
- Ένθετοι βρόχοι for: Εκτύπωση ορθογώνιου σχηματισμού
 - Εκτυπώνει έναν τριγωνικό σχηματισμό ακεραίων όπως και το προηγούμενο παράδειγμα, με τη διαφορά ότι αυτή τη φορά οι γραμμές των ψηφίων «συμπληρώνονται» με αστερίσκους, κατά τρόπο ώστε να σχηματίζεται ένα ορθογώνιο

Παραδείγματα βρόχων (2/2)

- Βρόχος for: Εκτύπωση ημερολογίου μηνός
 - Πρόγραμμα εκτύπωσης του ημερολογίου για έναν μήνα
- Βρόχος while: Εκτύπωση πλήθους και αθροίσματος ψηφίων
 - Εκτυπώνει το πλήθος και άθροισμα των ψηφίων ενός ακέραιου αριθμού που εισάγεται από τον χρήστη
- Βρόχος while: Εκτύπωση αριθμού με αντίστροφη σειρά ψηφίων
- Εκτύπωση ενός αριθμού αφού αντιστρέψουμε τη σειρά των ψηφίων του
- Βρόχος do ... while: Επικύρωση δεδομένων
 - Πρόγραμμα που επιδεικνύει μία κλασική τεχνική επικύρωσης δεδομένων

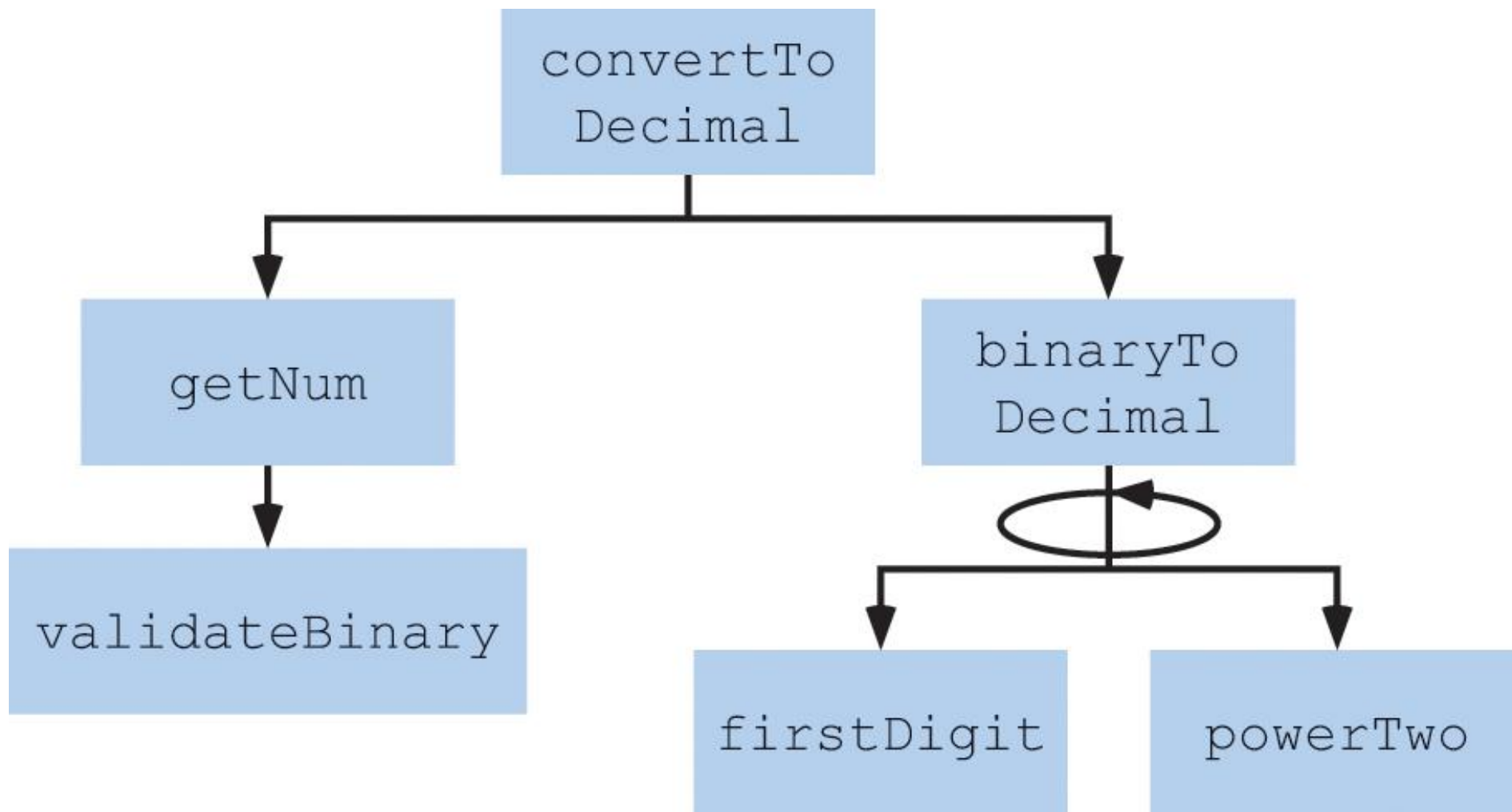


(α) Διάγραμμα ροής

1 ορισμός γραμμής (line) σε 1
 2 για όσο
 (γραμμή όχι μεγαλύτερη από όριο)
 2.1 ορισμός αριθμού (num) σε 1
 2.2 **για όσο**
 (αριθμός όχι μεγαλύτερος από γραμμή)
 2.2.1 **εκτύπωση αριθμού**
 2.2.2 **αύξηση αριθμού**
 2.3 **τέλος βρόχου**
 2.4 μετάβαση σε επόμενη γραμμή
 2.5 **αύξηση γραμμής**
 3 τέλος βρόχου

(β) Ψευδοκώδικας

ΣΧΗΜΑ 6-18 Διάγραμμα ροής και ψευδοκώδικας για το πρόγραμμα εκτύπωσης τριγωνικού σχηματισμού αριθμών.



ΣΧΗΜΑ 6-19 Σχεδίαση του προγράμματος μετατροπής δυαδικών αριθμών στο δεκαδικό.

6.7

**Άλλες εντολές
που σχετίζονται με βρόχους**

Άλλες εντολές που σχετίζονται με βρόχους

- Η γλώσσα C διαθέτει τρεις επιπλέον εντολές που σχετίζονται με τους βρόχους:
 - break
 - continue
 - goto
 - Η εντολή goto αντιβαίνει στις αρχές του δομημένου προγραμματισμού

Η εντολή `break`

- Όταν χρησιμοποιείται σε έναν βρόχο, η εντολή `break` προκαλεί τον τερματισμό του βρόχου
 - Πρακτικά, ισοδυναμεί με το να θέσουμε τη συνθήκη ελέγχου του βρόχου σε `false`
- Μπορεί να χρησιμοποιηθεί σε συνδυασμό με οποιοδήποτε είδος βρόχου

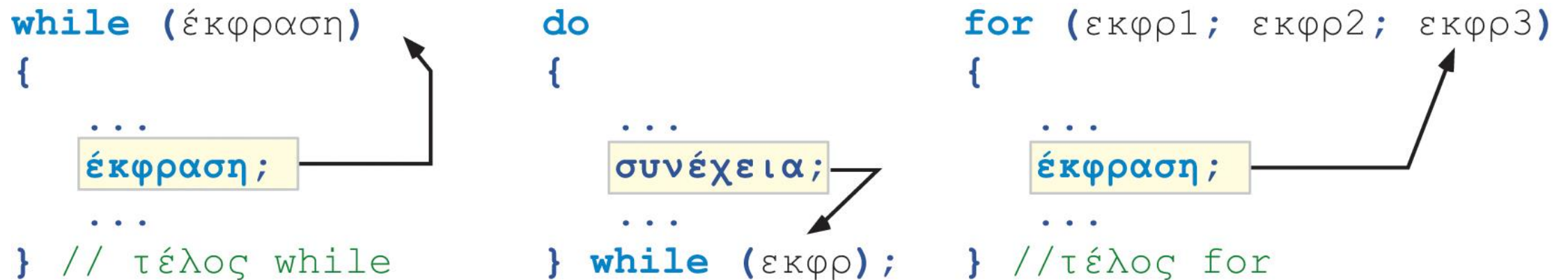
```
while (συνθήκη)
{
    ...
    for ( ...; ...; ... )
    {
        ...
        if (διαφορετικήΣυνθήκη)
            break;
        ...
    } // τέλος βρόχου for
    // επιπλέον εντολές βρόχου while
    ...
} // τέλος βρόχου while
```

Η εντολή `break` διακόπτει τον εσωτερικό βρόχο `for`. Ο βρόχος `while` εξακολουθεί να εκτελείται.

ΣΧΗΜΑ 6-20 Χρήση της εντολής `break` σε ένθετους βρόχους.

Η εντολή `continue`

- Η εντολή `continue` μεταφέρει τη ροή εκτέλεσης
 - στην έκφραση ελέγχου της συνθήκης τερματισμού για τους βρόχους `while` και `do ... while`
 - στην έκφραση ενημέρωσης για την εντολή `for`



ΣΧΗΜΑ 6-21 Η εντολή `continue`.

6.8

Εφαρμογές των επαναληπτικών διαδικασιών

Εφαρμογές των επαναληπτικών διαδικασιών

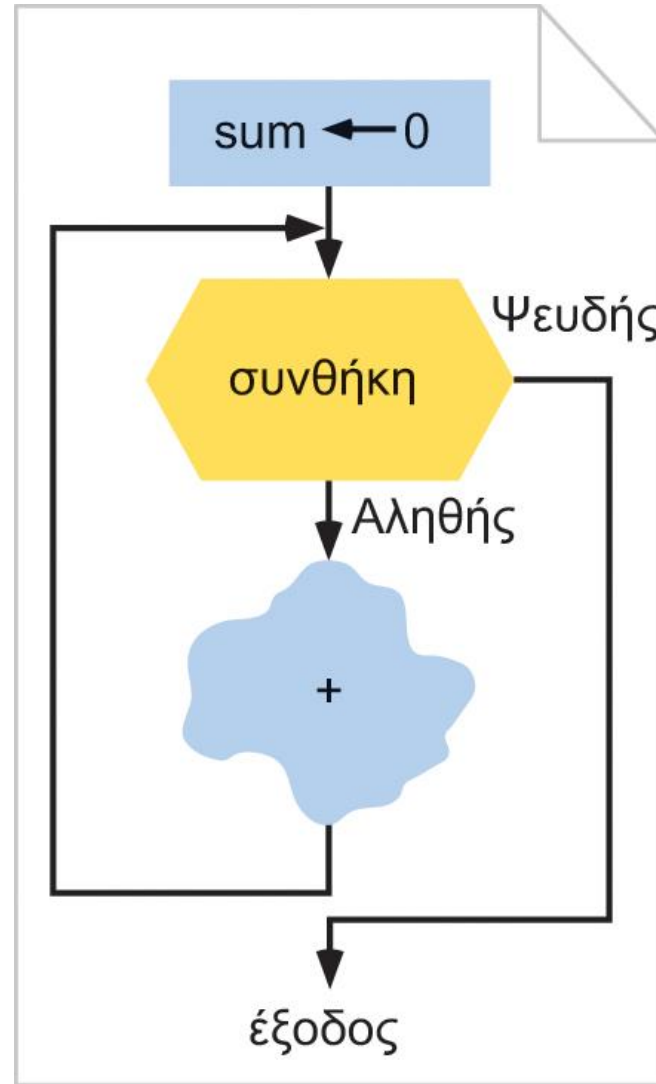
- Τέσσερις κοινές εφαρμογές των βρόχων:
 - Άθροιση
 - Ύψωση σε δύναμη
 - Εύρεση ελαχίστου/μεγίστου
 - Ερωτήματα αναζήτησης δεδομένων

Άθροιση (1/2)

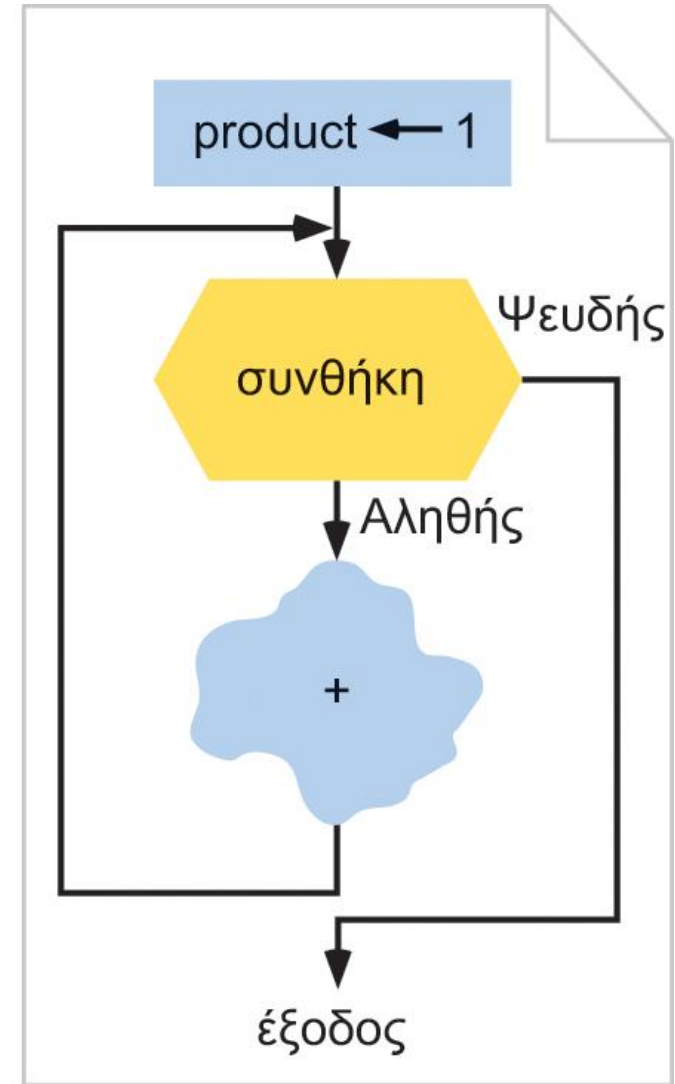
- Ο τελεστής άθροισης σε έναν βρόχο μπορεί να χρησιμοποιηθεί για την πρόσθεση πολλών αριθμών ή μιας μεταβλητής σειράς αριθμών
- Η διαδικασία άθροισης έχει τρία λογικά μέρη:
 - Αρχικοποίηση τυχόν απαραίτητων μεταβλητών, όπως η μεταβλητή δημιουργίας του αθροίσματος (ή τρέχοντος συνόλου)
 - Τον βρόχο, ο οποίος περιλαμβάνει τον κώδικα άθροισης και τυχόν κώδικα επικύρωσης δεδομένων (π.χ., «μόνο μη μηδενικοί αριθμοί γίνονται δεκτοί»)
 - Κώδικα για την εκτύπωση ή την επιστροφή του αποτελέσματος

Άθροιση (2/2)

ΣΧΗΜΑ 6-22 Βρόχοι άθροισης και πολλαπλασιασμού.



Άθροιση



Γινόμενο

Ύψωση σε δύναμη

- Ένας βρόχος πολλαπλασιασμού είναι χρήσιμος για δύο κοινές εφαρμογές:
 - Την ύψωση ενός αριθμού σε δύναμη
 - Τον υπολογισμό του παραγοντικού ενός αριθμού

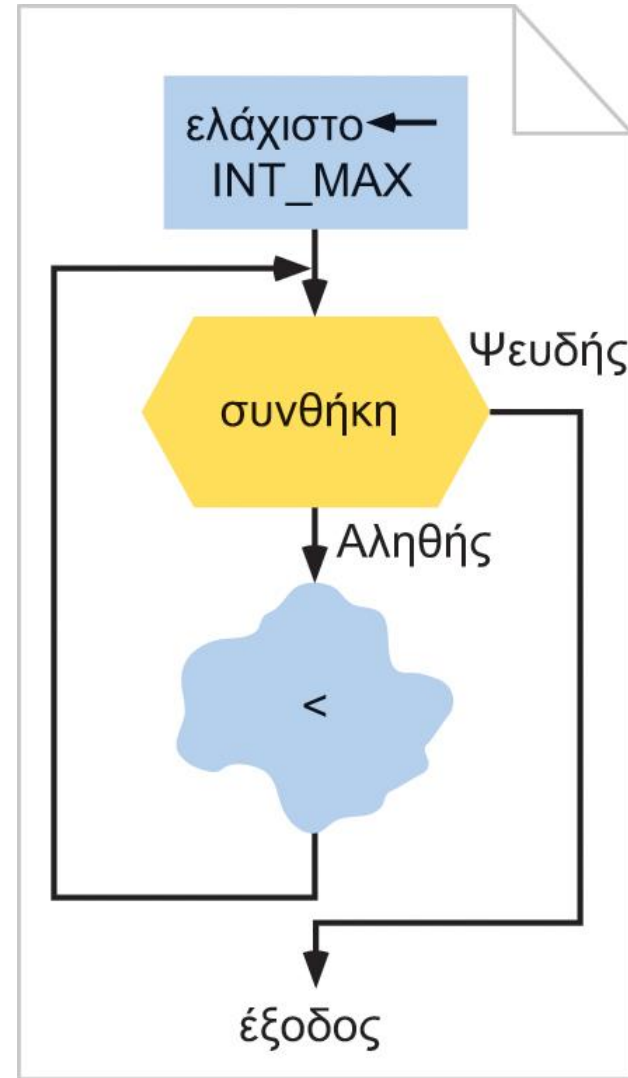
Σημείο προσοχής: αρχικοποίηση βρόχων

- Η λογική της αρχικοποίησης εξαρτάται από την εκάστοτε εφαρμογή:
δεν είναι ίδια για κάθε πρόβλημα
 - Στην περίπτωση της άθροισης, η αρχικοποίηση θέτει τη μεταβλητή κατακράτησης του αθροίσματος σε τιμή 0
 - Σε εφαρμογές που εμπλέκουν πολλαπλασιασμούς, όπως η συνάρτηση υπολογισμού δυνάμεων, η μεταβλητή κατακράτησης του αποτελέσματος πρέπει να αρχικοποιείται σε τιμή 1

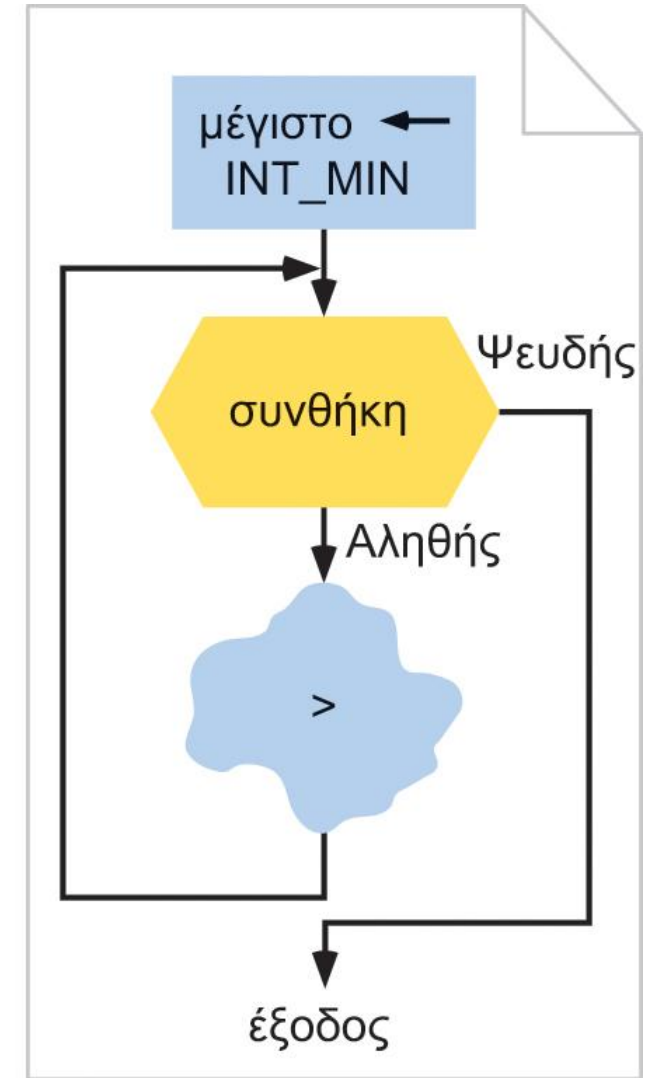
Εύρεση ελαχίστου και μεγίστου

ΣΧΗΜΑ 6-23

Βρόχοι για την εύρεση
ελαχίστου και μεγίστου.



Εύρεση ελαχίστου



Εύρεση μεγίστου

Ερωτήματα αναζήτησης δεδομένων

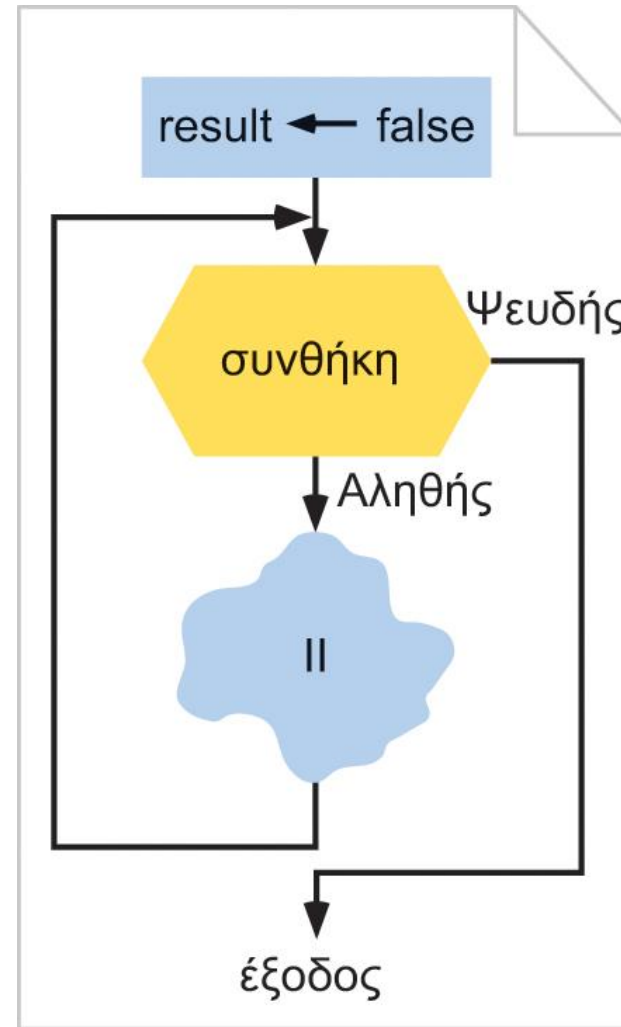
- Ερώτημα **any**: Χρησιμοποιείται όταν έχουμε μία λίστα δεδομένων και θέλουμε να μάθουμε εάν τουλάχιστον μία από τις καταχωρίσεις της λίστας ικανοποιεί ένα συγκεκριμένο κριτήριο
 - Η απάντηση στο ερώτημα any είναι ναι, εάν μία ή περισσότερες καταχωρίσεις δεδομένων ικανοποιούν το κριτήριο
 - Η απάντηση είναι όχι εάν καμία από τις καταχωρίσεις δεδομένων δεν ικανοποιεί το κριτήριο

Ερωτήματα αναζήτησης δεδομένων

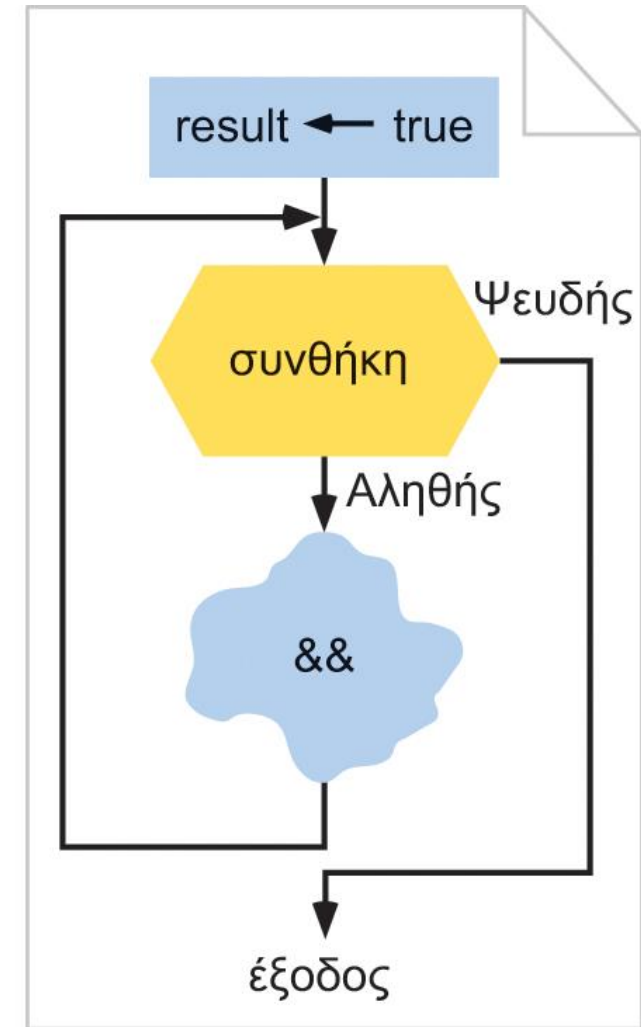
- Ερώτημα **all**: Χρησιμοποιείται όταν έχουμε μία λίστα δεδομένων και θέλουμε να βεβαιωθούμε ότι κάθε μία από τις καταχωρίσεις της ικανοποιεί ένα συγκεκριμένο κριτήριο

Ερωτήματα αναζήτησης δεδομένων

ΣΧΗΜΑ 6-24 Λογική δομή των ερωτημάτων any και all.



ερώτημα «any»



ερώτημα «all»

Σύνοψη

- Η πραγματική ισχύς των υπολογιστών έγκειται στην ικανότητά τους να επαναλαμβάνουν μία λειτουργία ή μία σειρά λειτουργιών πολλές φορές
- Για τον έλεγχο ενός βρόχου, χρειαζόμαστε μία συνθήκη μέσω της οποίας θα καθορίζεται το πλήθος των επαναλήψεων και ο τερματισμός του βρόχου
- Σε έναν βρόχο με συνθήκη εισόδου, σε κάθε επανάληψη, ελέγχουμε πρώτα τη συνθήκη. Εάν είναι αληθής, ο βρόχος επαναλαμβάνεται για άλλη μία φορά· διαφορετικά, ο βρόχος τερματίζεται
- Σε έναν βρόχο με συνθήκη εξόδου, σε κάθε επανάληψη, εκτελούμε πρώτα τις ενέργειες επεξεργασίας και κατόπιν ελέγχουμε τη συνθήκη. Εάν είναι αληθής, ο βρόχος επαναλαμβάνεται για άλλη μία φορά· διαφορετικά, ο βρόχος τερματίζεται
- Σε έναν βρόχο με συνθήκη εισόδου, η επεξεργασία (το σώμα του βρόχου) εκτελείται μηδέν ή περισσότερες φορές

Σύνοψη

- Σε έναν βρόχο με συνθήκη εξόδου, η επεξεργασία εκτελείται μία ή περισσότερες φορές
- Σε έναν βρόχο με συνθήκη εισόδου, εάν το σώμα εκτελείται n φορές, ο έλεγχος της συνθήκης τερματισμού εκτελείται $n + 1$ φορές
- Σε έναν βρόχο με συνθήκη εξόδου, εάν το σώμα εκτελείται n φορές, ο έλεγχος της συνθήκης τερματισμού εκτελείται n φορές
- Η έκφραση που ελέγχει τη συνθήκη τερματισμού ενός βρόχου πρέπει να αρχικοποιείται, είτε άμεσα είτε έμμεσα
- Εάν γνωρίζετε επακριβώς τον αριθμό των φορών που πρέπει να επαναληφθεί το σώμα, χρησιμοποιήστε έναν βρόχο ελεγχόμενο από μετρητή.
- Εάν πρέπει να λάβει χώρα ένα συμβάν για να τερματιστεί ο βρόχος, χρησιμοποιήστε έναν βρόχο ελεγχόμενο από συμβάν

Σύνοψη

- Η C παρέχει τρεις εντολές βρόχων: while, for και do...while
- Ο βρόχος while είναι ένας βρόχος με συνθήκη εισόδου. Μπορεί να χρησιμοποιηθεί τόσο για βρόχους ελεγχόμενους με μετρητή όσο και για βρόχους ελεγχόμενους από συμβάντα, αλλά συνήθως χρησιμοποιείται μόνο για τη δεύτερη περίπτωση
- Ο βρόχος for είναι ένας βρόχος με συνθήκη εισόδου. Μπορεί να χρησιμοποιηθεί τόσο για βρόχους ελεγχόμενους με μετρητή όσο και για βρόχους ελεγχόμενους από συμβάντα, αλλά συνηθέστερα χρησιμοποιείται για την πρώτη περίπτωση
- Σε έναν βρόχο for, η μεταβλητή-μετρητής μπορεί να οριστεί τοπικά στην εντολή for
- Ο βρόχος do ... while είναι ένας βρόχος με συνθήκη εξόδου. Χρησιμοποιείται συνήθως όταν το σώμα πρέπει να εκτελεστεί τουλάχιστον μία φορά

Σύνοψη

- Η εντολή `break` χρησιμοποιείται για τον πρόωρο τερματισμό ενός βρόχου. Για εναρμόνιση με τις καλές πρακτικές δομημένου προγραμματισμού, συνιστούμε να χρησιμοποιείτε την εντολή `break` μόνο σε εντολές `switch`
- Η εντολή `continue` χρησιμοποιείται για την παράλειψη των υπόλοιπων εντολών στο σώμα ενός βρόχου και την έναρξη μιας επόμενης επανάληψής του (δεν τερματίζει τον βρόχο). Για εναρμόνιση με τις καλές πρακτικές δομημένου προγραμματισμού, συνιστούμε να μη χρησιμοποιείτε ποτέ την εντολή `continue`
- Για σκοπούς επικύρωσης δεδομένων, συνήθως προσφέρεται ο βρόχος `do ... while`
- Σε ένα διάγραμμα δομής, ένας βρόχος συμβολίζεται με έναν κύκλο στη γραμμή που τον ενώνει με τη συνάρτηση (ή τις συναρτήσεις) που καλεί. Στο διάγραμμα αποτυπώνονται μόνο οι βρόχοι που καλούν άλλες συναρτήσεις



Το παρόν συνοδευτικό έργο **παρέχεται** αποκλειστικά και μόνο στους διδάσκοντες που έχουν επιλέξει το αντίστοιχο βιβλίο μέσω του συστήματος του **Ευδόξου** ως διδακτικό σύγγραμμα για τη διδασκαλία των μαθημάτων τους και την αξιολόγηση της εκμάθησης των φοιτητών και για όσο χρονικό διάστημα διατηρείται η επιλογή του συγκεκριμένου συγγράμματος. Η **διάδοση, δημοσίευση, αναπαραγωγή ή πώληση** οπουδήποτε μέρους αυτού του έργου με οποιοδήποτε μέσο καταστρέφει την ακεραιότητα του έργου **και για σκοπούς διαφορετικούς από αυτούς για τους οποίους έχει χορηγηθεί η σχετική άδεια δεν επιτρέπεται**. Το περιεχόμενο του έργου **δεν θα πρέπει να διατίθεται** στους φοιτητές παρά μόνο όταν αυτό αναφέρεται ρητά ότι μπορεί να γίνει. Η **χρήση** του παρόντος έργου γίνεται αποκλειστικά από τον διδάσκοντα στα πλαίσια των εκπαιδευτικών καθηκόντων του και με τη χρήση των μέσων που αυτός διαχειρίζεται και έχει στη διάθεσή του από τις **επίσημες υπηρεσίες του εκπαιδευτικού ιδρύματος** όπου ανήκει, διασφαλίζοντας τη μη περαιτέρω διάδοση, δημοσίευση και αναπαραγωγή του έργου προς τρίτους. Ο διδάσκων δύναται να **προσαρμόζει** μέρος του υλικού των διαφανειών σύμφωνα με τις διδακτικές του ανάγκες, αναφερόμενος όμως πάντοτε στην αρχική πηγή αναφοράς. Το παρόν έργο προστατεύεται από την ελληνική και ευρωπαϊκή νομοθεσία περί πνευματικής ιδιοκτησίας καθώς και από τη νομοθεσία του κράτους όπου ανήκει η ξενόγλωσση πρωτότυπη έκδοση του έργου.