

Εισαγωγή στην επιστήμη των υπολογιστών

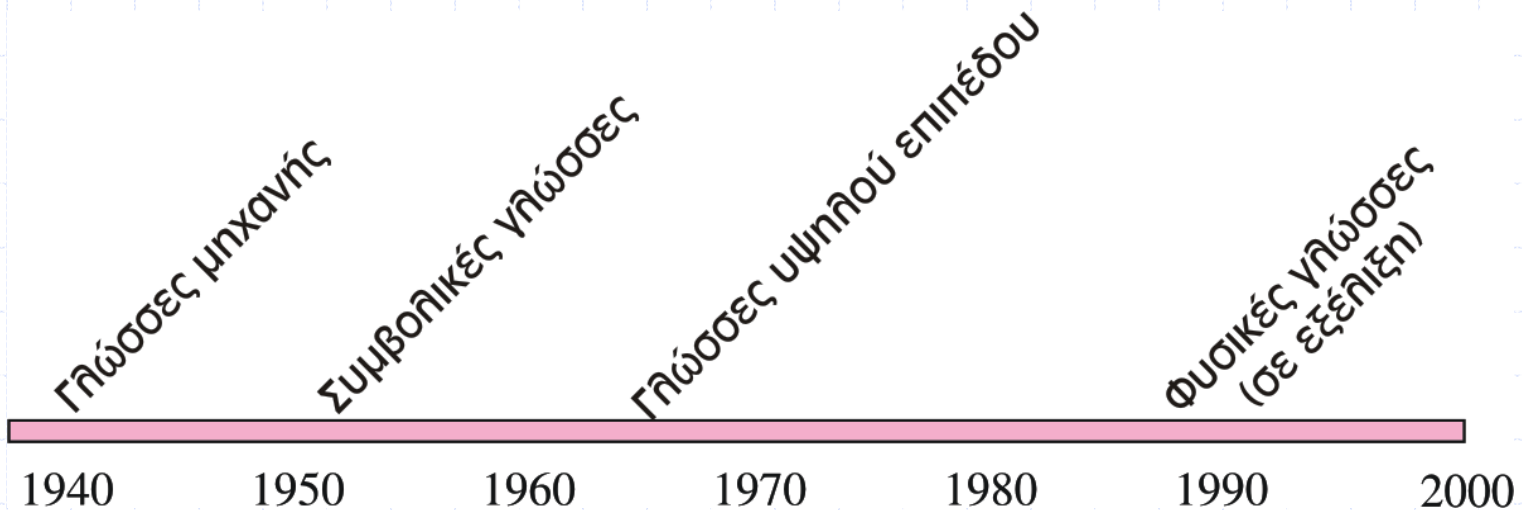
Λογισμικό Υπολογιστών

Κεφάλαιο 9ο

Γλώσσες Προγραμματισμού

Εξέλιξη

- ◆ Οι **γλώσσες προγραμματισμού** είναι σύνολα από προκαθορισμένες λέξεις οι οποίες συνδυάζονται σε προγράμματα σύμφωνα με προκαθορισμένους κανόνες (**σύνταξη**).



Γλώσσες Μηχανής

- ◆ Η μόνη γλώσσα που καταλαβαίνει ένας υπολογιστής είναι η γλώσσα μηχανής
- ◆ Το εσωτερικό κύκλωμα του υπολογιστή αποτελείται από διακόπτες, τρανζίστορ, και άλλες ηλεκτρονικές συσκευές οι οποίες μπορούν να έχουν μία από δύο καταστάσεις:
 - να είναι κλειστές (off)
 - ή ανοιχτές (on)
- ◆ Η κατάσταση off αναπαρίσταται από το 0, ενώ η κατάσταση on αναπαρίσταται από το 1

Γλώσσες Μηχανής

1	00000000	00000100	000000000000000000	
2	01011110	00001100	11000010	000000000000000010
3		11101111	00010110	000000000000000101
4		11101111	10011110	000000000000001011
5	11111000	10101101	11011111	00000000000010010
6		01100010	11011111	00000000000010101
7	11101111	00000010	11111011	00000000000010111
8	11110100	10101101	11011111	00000000000011110
9	00000011	10100010	11011111	00000000000100001
10	11101111	00000010	11111011	00000000000100100
11	01111110	11110100	10101101	
12	11111000	10101110	11000101	00000000000101011
13	00000110	10100010	11111011	00000000000110001
14	11101111	00000010	11111011	00000000000110100
15			00000100	00000000000111101
16			00000100	00000000000111101

Συμβολικές Γλώσσες

- ◆ Μια **συμβολική γλώσσα** χρησιμοποιεί σύμβολα για την αναπαράσταση των διαφόρων εντολών μηχανής.
- ◆ Οι συμβολικές γλώσσες ονομάζονται επίσης και γλώσσες assembly

Συμβολικές Γλώσσες

```
1  Entry    main,    ^m<r2>
2  subl2    #12,sp
3  jsb      C$MAIN_ARGS
4  movab    $CHAR_STRING_CON
5
6  pushal   -8(fp)
7  pushal   (r2)
8  calls    #2,read
9  pushal   -12(fp)
10 pushal   3(r2)
11 calls    #2,read
12 mull3    -8(fp),-12(fp),-
13 pushal   6(r2)
14 calls    #2,print
15 crl      r0
16 ret
```

Για την μετατροπή της μνημονικής μορφής των προγραμμάτων σε γλώσσα μηχανής αναπτύχθηκαν προγράμματα που ονομάζονται **συμβολομεταφραστές** (assemblers). Οι συμβολομεταφραστές είναι προγράμματα τα οποία μεταφράζουν άλλα προγράμματα που είναι γραμμένα σε μνημονική μορφή στην αριθμητική γλώσσα μηχανής που είναι συμβατή με την μηχανή. Τα μεταφραστικά αυτά προγράμματα ονομάσθηκαν "assembler", επειδή η δουλειά τους ήταν να συναρμολογούν τις πραγματικές εντολές μηχανής από τα μνημονικά ονόματα και τα αναγνωριστικά.

Ένα μνημονικό σύστημα για την αναπαράσταση προγραμμάτων ονομάζεται γενικά **γλώσσα assembly**.

Γλώσσες Υψηλού Επιπέδου

- ◆ Οι **γλώσσες υψηλού επιπέδου** μπορούν να εκτελεστούν σε όλους τους υπολογιστές και απαλλάσσουν τον προγραμματιστή από προβλήματα που αφορούν ένα συγκεκριμένο είδος υλικού.
- ◆ Γλώσσες υψηλού επιπέδου είναι οι BASIC, Pascal, Ada, C, C++, και Java

Γλώσσες Υψηλού Επιπέδου

```
1  /* Αυτό το πρόγραμμα διαβάζει δύο ακεραίους αριθμούς από το
2     πληκτρολόγιο και τυπώνει το γινόμενο τους.
3  */
4  #include <iostream.h>
5
6  int main (void)
7  {
8     // Local Declarations
9     int number1;
10    int number2;
11    int result;
12    // Statements
13    cin >> number1;
14    cin >> number2;
15    result = number1 * number2;
16    cout << result;
17    return 0;
18 }
```

Φυσικές Γλώσσες

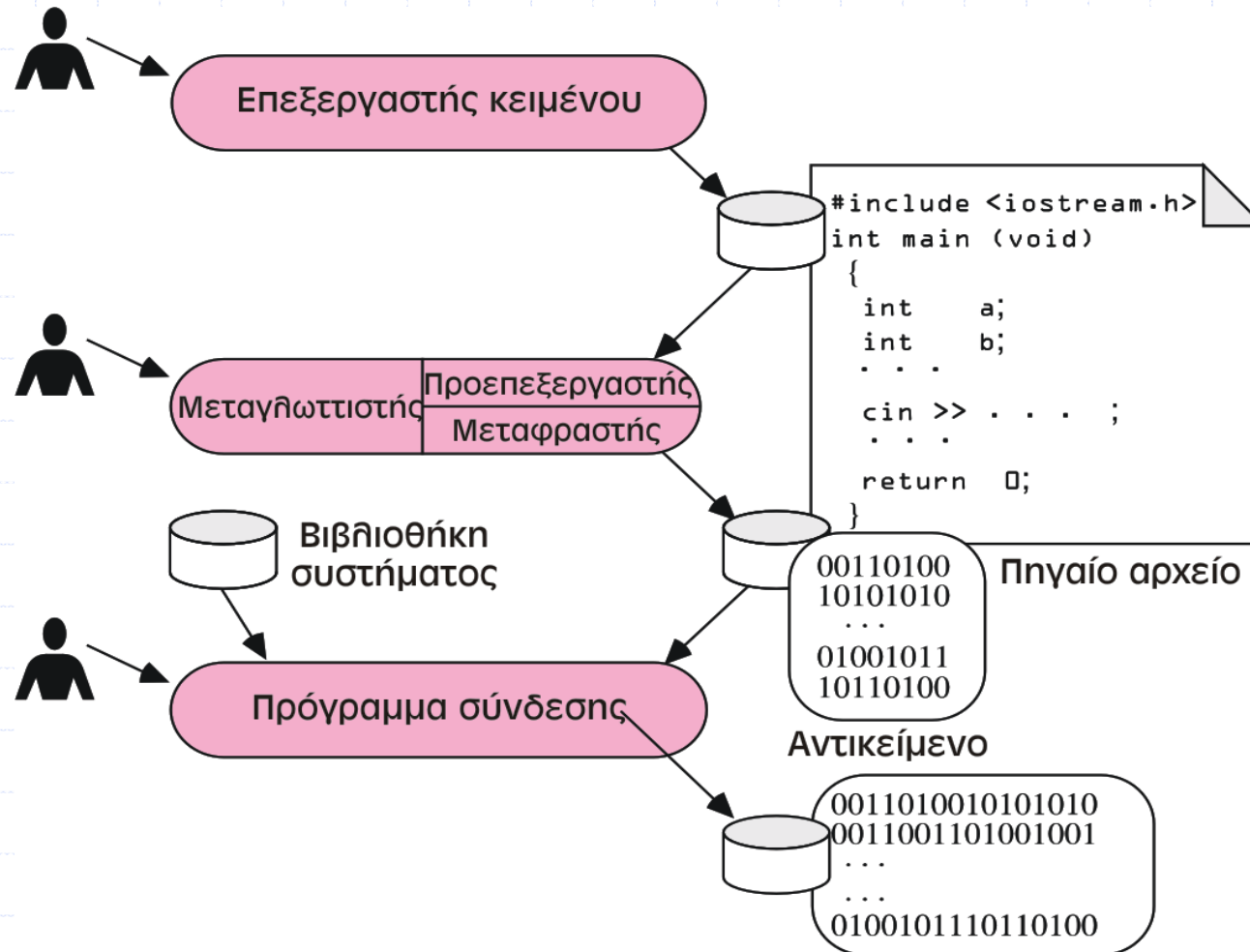
- ◆ Το ιδανικό θα ήταν να μπορούσαμε να χρησιμοποιούμε τη **φυσική** μας **γλώσσα** (για παράδειγμα, Ελληνικά, Αγγλικά, ή ακόμα και Κινέζικα), και ο υπολογιστής να καταλαβαίνει και να εκτελεί τις εντολές μας άμεσα.
- ◆ Σήμερα γίνεται σημαντική δουλειά στα εργαστήρια επάνω στο θέμα των φυσικών γλωσσών.
- ◆ Προς το παρόν, η χρήση των φυσικών γλωσσών στη βιομηχανία είναι πολύ περιορισμένη.

Κατασκευή ενός Προγράμματος

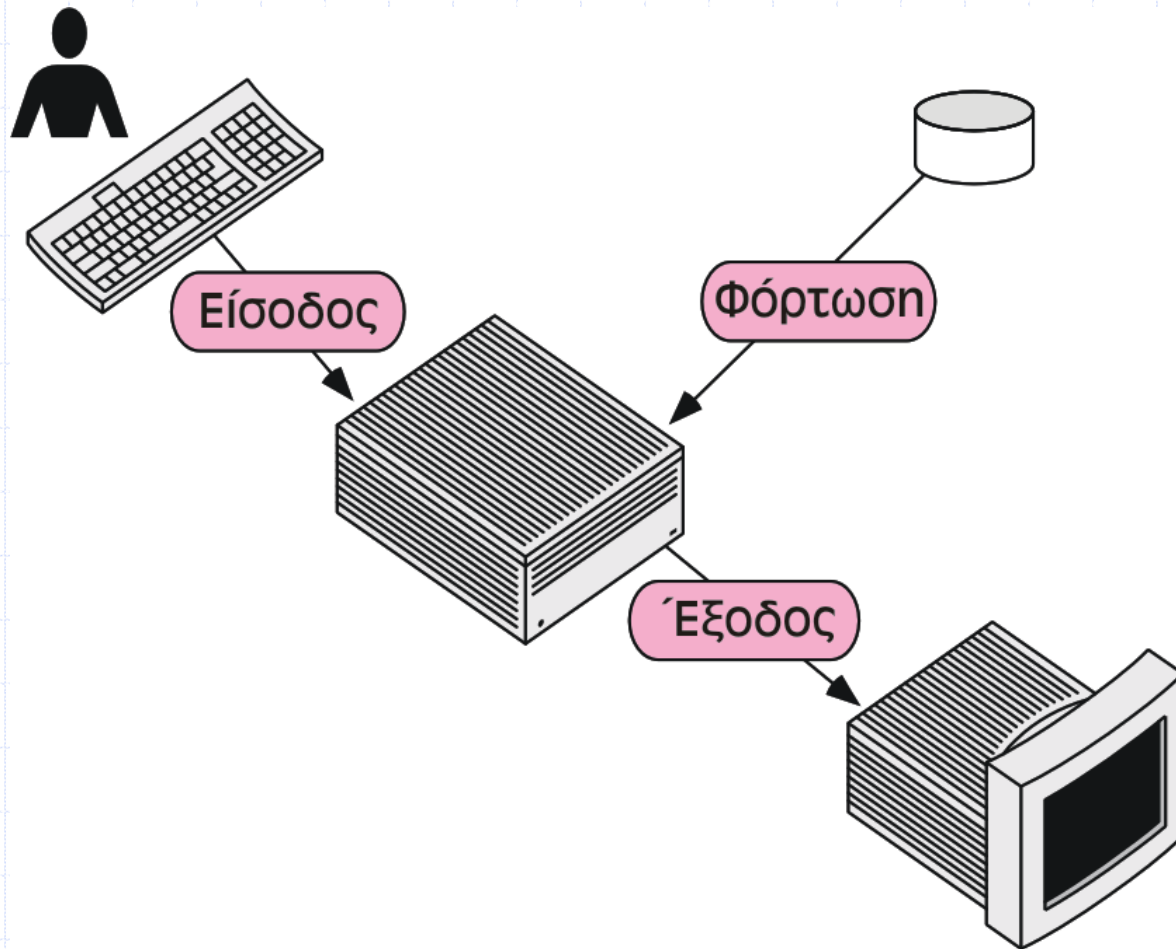
◆ Δουλειά του προγραμματιστή είναι να γράψει ένα πρόγραμμα και μετά να το μετατρέψει σε **εκτελέσιμο** (γλώσσας μηχανής) **αρχείο**. Αυτή η διαδικασία έχει τρία βήματα:

- Συγγραφή και διόρθωση του προγράμματος
- Μεταγλώττιση του προγράμματος
- Σύνδεση του προγράμματος με τις απαραίτητες υπομονάδες βιβλιοθηκών

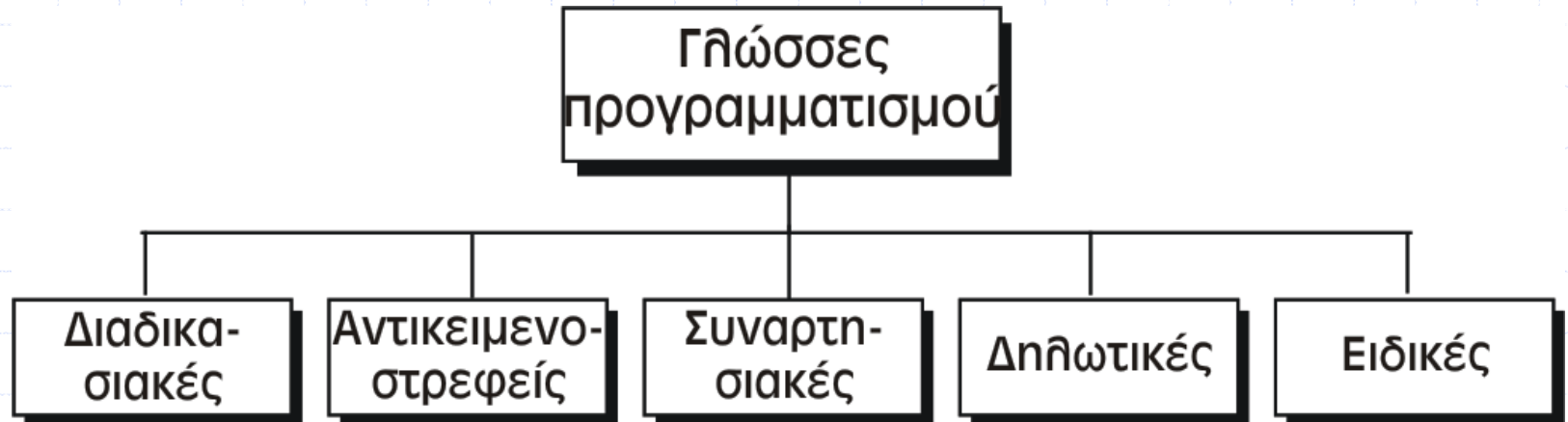
Κατασκευή ενός Προγράμματος



Εκτέλεση Προγράμματος



Κατηγορίες Γλωσσών



Διαδικασιακές (προστακτικές) Γλώσσες

- ◆ Σε μια διαδικασιακή γλώσσα ένας αλγόριθμος μεταφράζεται σε κώδικα. Ο κώδικας χειρίζεται δεδομένα και ελέγχει την εκτέλεση εντολών.
- ◆ Διαδικασιακές γλώσσες είναι η
 - FORTRAN
 - COBOL
 - Pascal
 - C
 - Ada

Αντικειμενοστρεφείς Γλώσσες

◆ Σε μια **αντικειμενοστραφή γλώσσα**, τα αντικείμενα και οι λειτουργίες που εφαρμόζονται σε αυτά είναι συνδεδεμένα μεταξύ τους.

- C++
- Java

Αντικειμενοστρεφείς Γλώσσες

- ◆ Στη σχεδίαση της γλώσσας C++ χρησιμοποιήθηκαν τρεις αρχές:
 - η ενθυλάκωση
 - η κληρονομικότητα
 - ο πολυμορφισμός

Αντικειμενοστρεφείς Γλώσσες

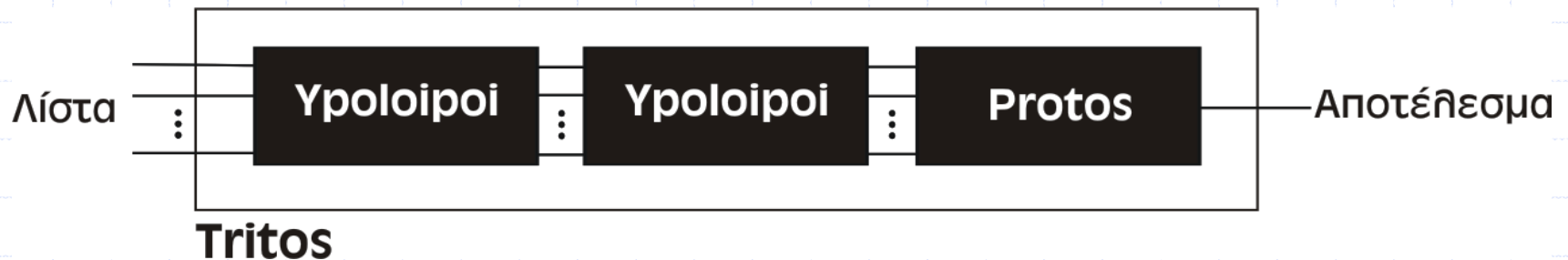
- ◆ **Ενθυλάκωση** (encapsulation) ονομάζεται η ιδέα της απόκρυψης των δεδομένων και κάποιων λειτουργιών που μπορούν να εκτελεστούν σε αυτά τα δεδομένα, μέσα στο αντικείμενο
- ◆ Στη C++ ένα αντικείμενο μπορεί να κληρονομήσει από κάποιο άλλο αντικείμενο. Αυτή η έννοια ονομάζεται **κληρονομικότητα** (inheritance).
- ◆ Στη C++, **πολυμορφισμός** (polymorphism) ονομάζεται η δυνατότητα ορισμού πολλών λειτουργιών με το ίδιο όνομα, οι οποίες κάνουν διαφορετικά πράγματα σε σχετιζόμενες κλάσεις

Αντικειμενοστρεφείς Γλώσσες

- ◆ Η γλώσσα **Java** αναπτύχθηκε από την εταιρεία Sun Microsystems και βασίζεται στη C και τη C++
- ◆ Η πολλαπλή κληρονομικότητα, έχει καταργηθεί
- ◆ Ένα πρόγραμμα της Java μπορεί να είναι είτε εφαρμογή (application) είτε μικροεφαρμογή (applet).
- ◆ Άλλο ένα ενδιαφέρον χαρακτηριστικό της Java είναι η πολυνημάτωση (multithreading). Νήμα (thread) ονομάζεται μια σειρά ενεργειών οι οποίες εκτελούνται η μία μετά την άλλη

Συναρτησιακές Γλώσσες

- ◆ Σε μια συναρτησιακή γλώσσα ο αλγόριθμος έχει μαθηματική φύση. Συναρτησιακές γλώσσες είναι η LISP και η Scheme



Δηλωτικές και Ειδικές Γλώσσες

- ◆ Μια δηλωτική γλώσσα χρησιμοποιεί τις αρχές του λογικού συλλογισμού.
 - Δηλωτική γλώσσα είναι η Prolog
- ◆ Μια ειδική γλώσσα δεν μπορεί να τοποθετηθεί σε κάποια από τις υπόλοιπες τέσσερις ομάδες.
 - Ειδικές γλώσσες είναι οι HTML, PERL, και SQL

HTML

- ◆ Η **HTML** (Hypertext Markup Language, δηλαδή Γλώσσα Σήμανσης Υπερκειμένου) είναι μια ψευδογλώσσα που περιέχει σύμβολα τα οποία χρησιμεύουν ως οδηγίες μορφοποίησης και σύνδεσης με άλλα αρχεία.
- ◆ Η HTML επιτρέπει τη χρήση χαρακτήρων ASCII τόσο για το κυρίως κείμενο όσο και για τις οδηγίες μορφοποίησης. Με αυτόν τον τρόπο κάθε υπολογιστής μπορεί να λάβει ολόκληρο το έγγραφο σε μορφή ASCII

HTML

- ◆ Ένα πρόγραμμα HTML αποτελείται από δύο μέρη:
 - την **κεφαλίδα**
 - το **σώμα**
- ◆ Υπάρχουν επίσης **ετικέτες** οι οποίες περιέχουν πληροφορίες σχετικά με τη δομή του κειμένου

ΕΤΙΚΕΤΕΣ

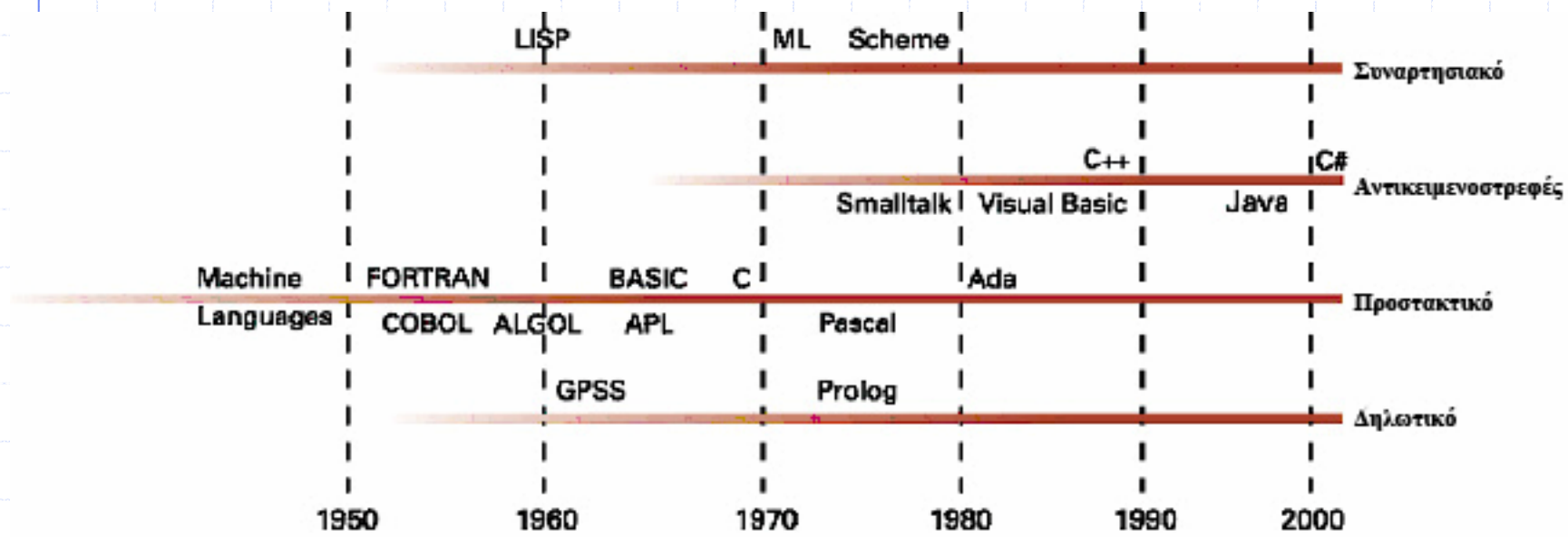
Ετικέτα αρχής	Ετικέτα τέλους	Σημασία
<HTML>	</HTML>	Ορίζει ένα έγγραφο HTML
<HEAD>	</HEAD>	Ορίζει την κεφαλίδα του εγγράφου
<BODY>	</BODY>	Ορίζει το σώμα του εγγράφου
<TITLE>	</TITLE>	Ορίζει τον τίτλο του εγγράφου
<H <i>i</i> >	</H <i>i</i> >	Ορίζει διάφορες κεφαλίδες (το <i>i</i> είναι ακέραιος)
		Έντονη γραφή
<I>	</I>	Πλάγια γραφή
<U>	</U>	Υπογράμμιση

ΕΤΙΚΕΤΕΣ

Ετικέτα αρχής	Ετικέτα τέλους	Σημασία
_		Δείκτης
[]	Εκθέτης
<CENTER>	</CENTER>	Κεντράρισμα
 		Αλλαγή γραμμής
		Διατεταγμένη λίστα
		Μη διατεταγμένη λίστα
		Στοιχείο λίστας
		Ορίζει μια εικόνα
<A>		Ορίζει μια διεύθυνση (υπερσύνδεσμο)

Πρόγραμμα HTML

```
<HTML>  
  <HEAD>  
    <TITLE> Δείγμα εγγράφου </TITLE>  
  </HEAD>  
<BODY>  
  Μια εικόνα του βιβλίου:  
  <IMG SRC="Pictures/book1.gif"  
  ALIGN=MIDDLE>  
</BODY>  
</HTML>
```



Μια διαδικασιακή γλώσσα: C

- ◆ Αναγνωριστικά

- ◆ Τύποι δεδομένων

- Βασικοί τύποι (int, char, float)
- Παραγόμενοι τύποι (pointer, enumerated, union, structure)

Μεταβλητές

Τύπος μεταβλητής

Αναγνωριστικό μεταβλητής

```
char code ;  
  
int i ;  
  
long national_debt;  
  
float payRate ;  
  
double pi ;
```

Πρόγραμμα

Αναγνωριστικό μεταβλητής

B	code
14	i
1000000000000	national_debt
14,25	payRate
3,1415926536	pi

Μνήμη

Μεταβλητές

- ◆ Δήλωση και ορισμός μεταβλητών
 - float price;
- ◆ Ανάθεση αρχικής τιμής σε μεταβλητή
 - float price=23.45;

Σταθερές

- ◆ Κυριολεκτική Σταθερά
 - `perifereia = 2*mhkos*platos`
- ◆ Επώνυμη Σταθερά
 - `const pi = 3.14`
- ◆ Συμβολική Σταθερά
 - `#define taxRate 0.0825`

Είσοδος και Έξοδος

◆ Είσοδος

- `scanf("%d", &num);`

◆ Έξοδος

- `printf("Η τιμή του αριθμού είναι : %d", num);`

Παραστάσεις

◆ Μια **παράσταση** (expression) είναι μια διαδοχή τελεστών και τελεστών που έχει ως αποτέλεσμα μια μοναδική τιμή. Για παράδειγμα, η ακόλουθη είναι μια παράσταση με τιμή 10.

- $2 * 5$

Τελεστές

◆ Αριθμητικοί τελεστές

Τελεστής	Ορισμός	Παράδειγμα
+	Πρόσθεση	3 + 5
-	Αφαίρεση	2 - 4
*	Πολλαπλασιασμός	Num * 5
/	Διαίρεση (πηλίκο)	Sum/Count
%	Διαίρεση (υπόλοιπο)	Count % 4
++	Αύξηση κατά ένα (πρόσθεση του 1 στην τιμή της μεταβλητής)	Count++
--	Μείωση κατά ένα (αφαίρεση του 1 από την τιμή της μεταβλητής)	Count--

Τελεστές

◆ Σχεσιακοί τελεστές

Τελεστής	Ορισμός	Παράδειγμα
<	Μικρότερο από	Num < 5
<=	Μικρότερο από ή ίσο με	Num1 <= 5
>	Μεγαλύτερο από	Num2 > 3
>=	Μεγαλύτερο από ή ίσο με	Num2 >= 3
==	Ίσο με	Num1 == Num2
!=	Διάφορο του	Num1 != Num2

Τελεστές

◆ Λογικοί τελεστές

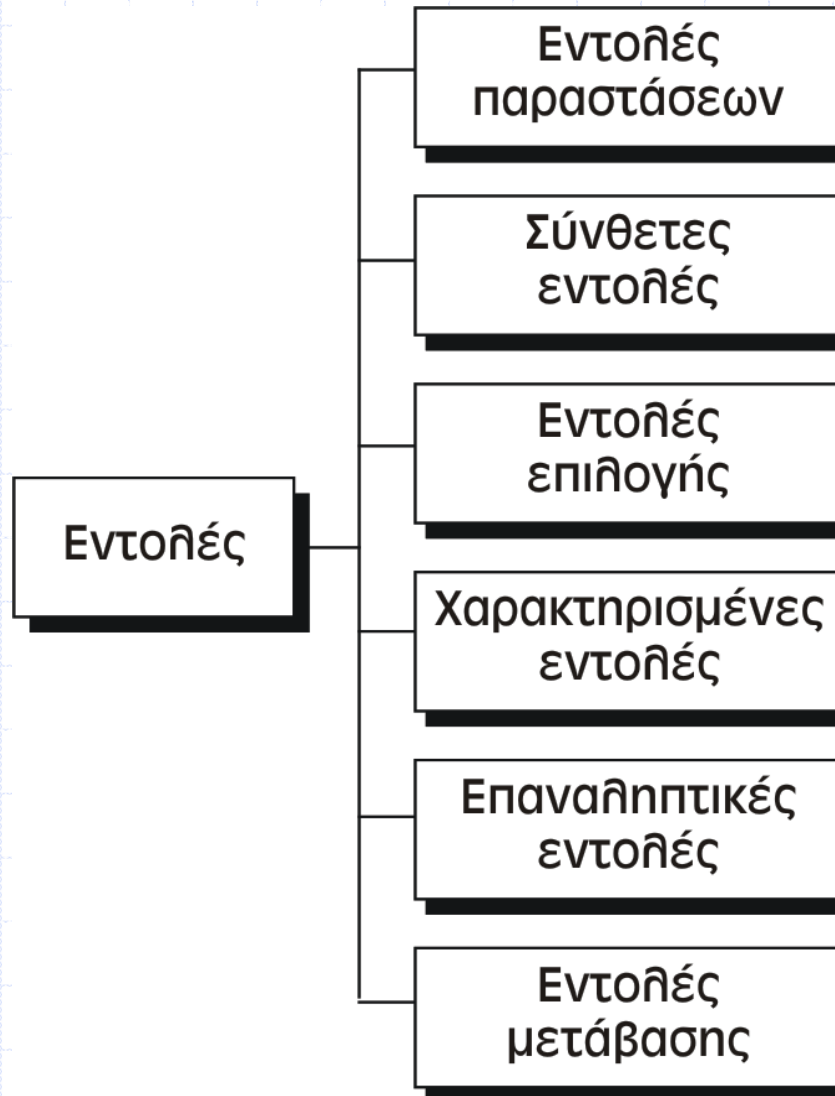
Τελεστής	Ορισμός	Παράδειγμα
!	NOT	!(Num1 < Num2)
&&	AND	(Num1 < 5) && (Num2 > 10)
	OR	(Num1 < 5) (Num2 > 10)

Τελεστές

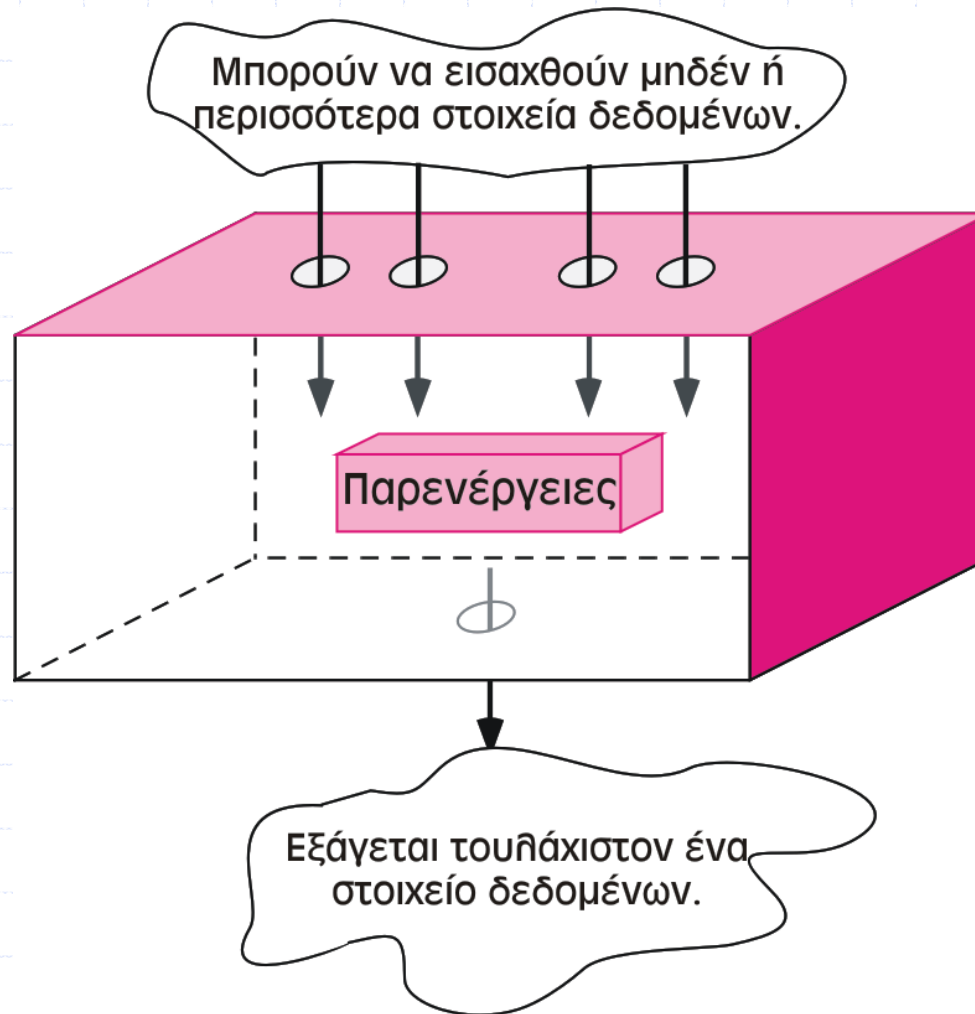
◆ Τελεστές ανάθεσης

Τελεστής	Παράδειγμα	Σημασία
=	num = 5 ;	Αποθήκευση του 5 στη μεταβλητή <code>num</code>
+=	num += 5 ;	Το ίδιο με <code>num = num + 5</code>
-=	num -= 5 ;	Το ίδιο με <code>num = num - 5</code>
*=	num *= 5 ;	Το ίδιο με <code>num = num * 5</code>
/=	num /= 5 ;	Το ίδιο με <code>num = num / 5</code>
&=	num %= 5 ;	Το ίδιο με <code>num = num % 5</code>

Εντολές



Συναρτήσεις



Συναρτήσεις

```
#include <stdio.h>
```

```
int multiply (int num1, int num2);
```

```
int main (void)  
{
```

```
    product = multiply (multiplier, multiplicand);
```

```
    return 0;  
}
```

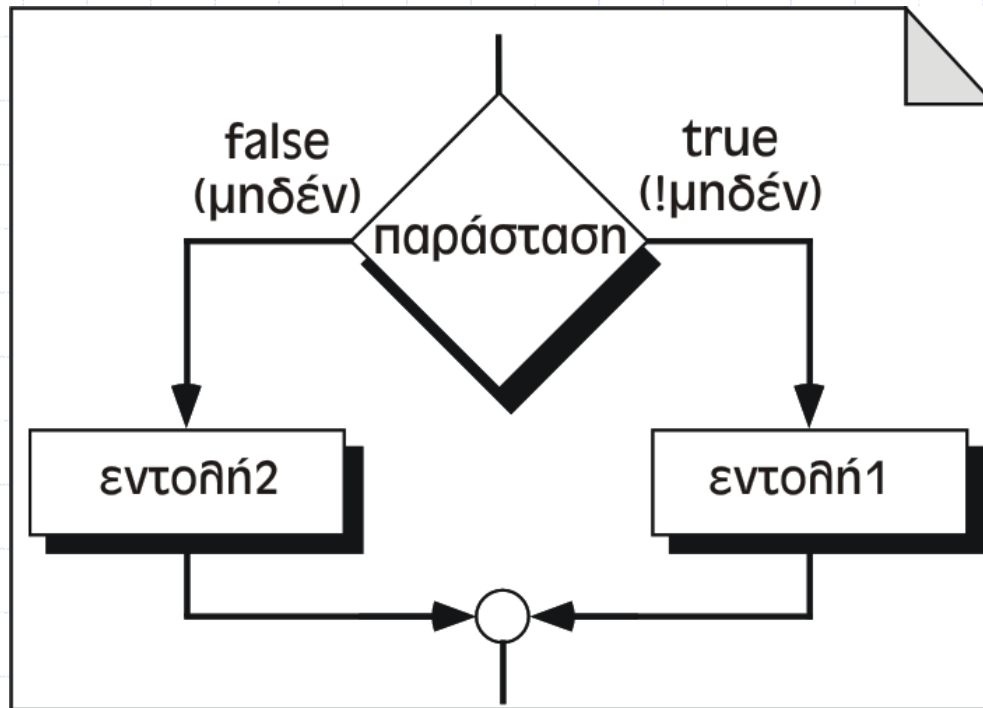
Δήλωση στην
καθοδική περιοχή

Η κλήση γίνεται στο
τμήμα των εντολών.

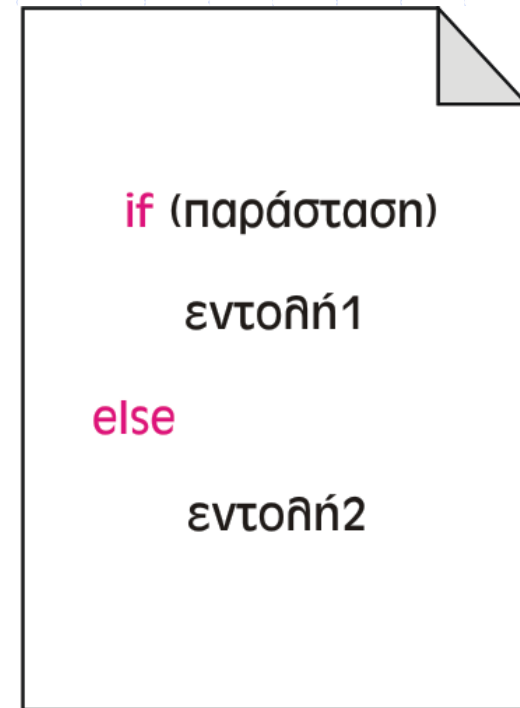
Ο ορισμός γίνεται μετά
την καλούσα συνάρτηση

```
int multiply (int num1,  
             int num2)  
{  
    return (num1 * num2);  
}
```


Επιλογή



α. Λογική ροή

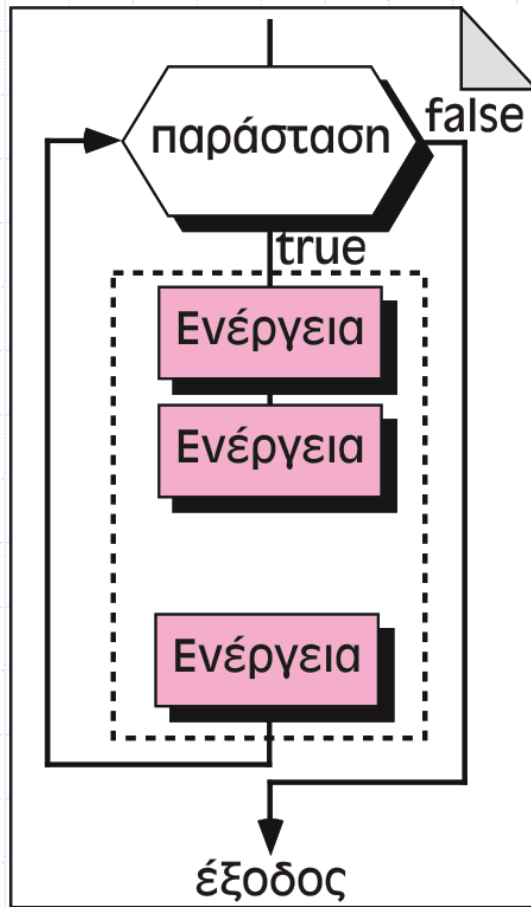


β. Κώδικας

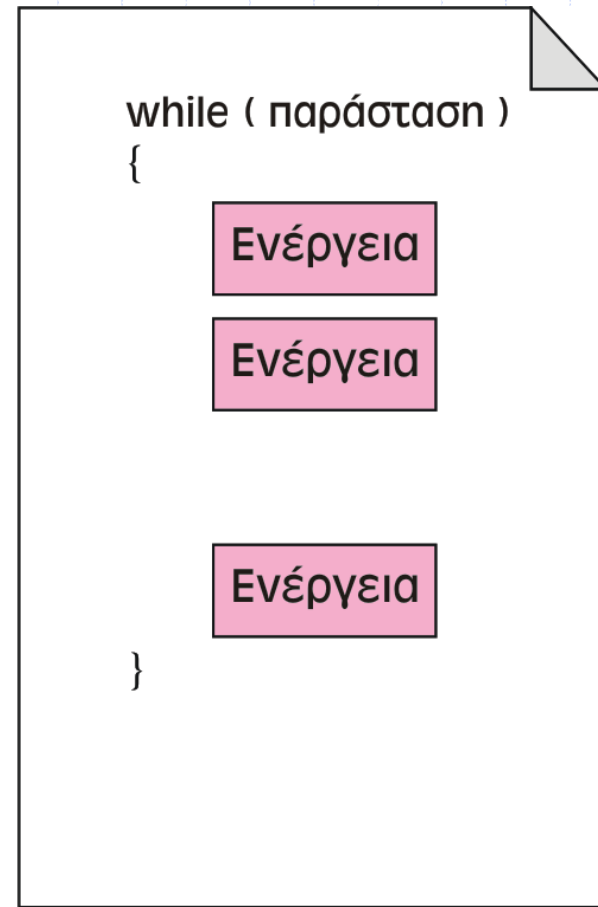
Επιλογή

```
switch ( παράσταση )  
{  
  case σταθερά-1 : εντολή  
                  εντολή  
  
  case σταθερά-2 : εντολή  
                  εντολή  
  
  case σταθερά-v : εντολή  
                  εντολή  
  
  default       : εντολή  
                  εντολή  
  
}
```

Επανάληψη

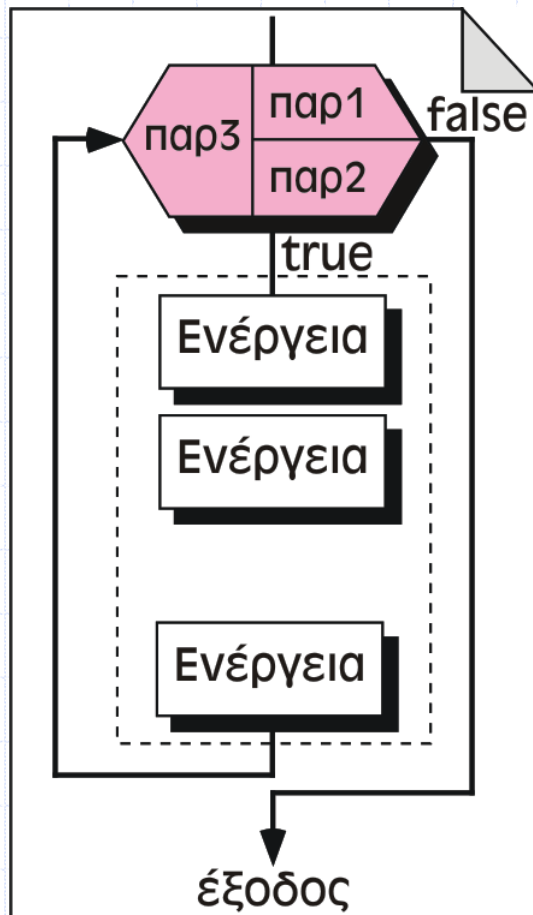


α. Διάγραμμα ροής

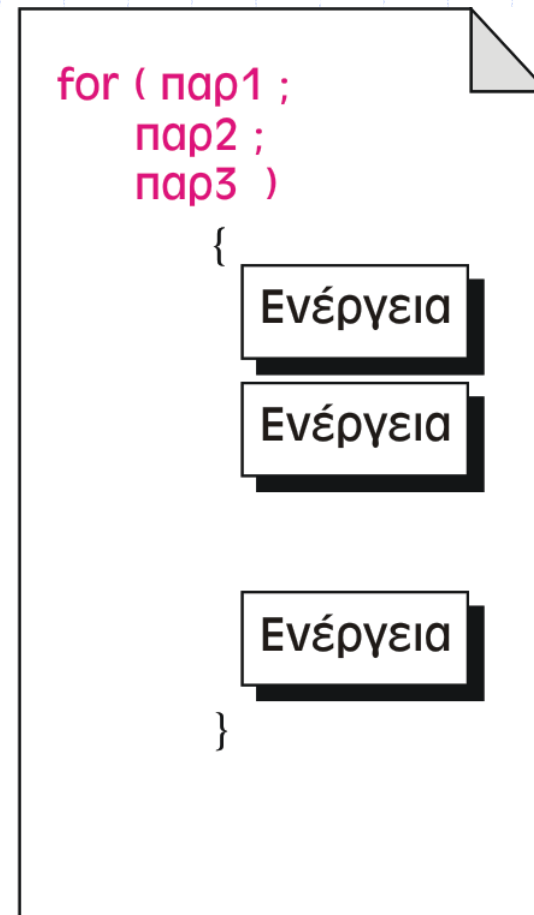


β. Κώδικας γλώσσας C

Επανάληψη

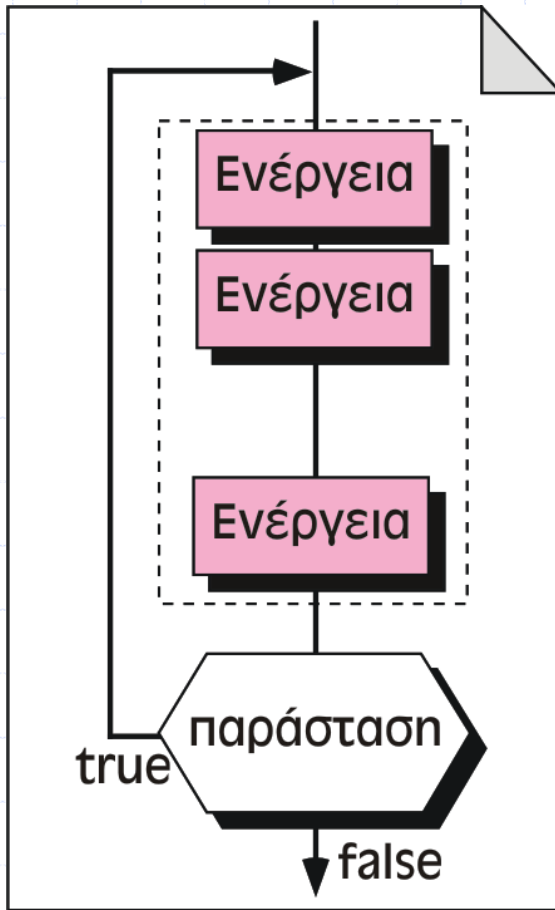


α. Διάγραμμα ροής

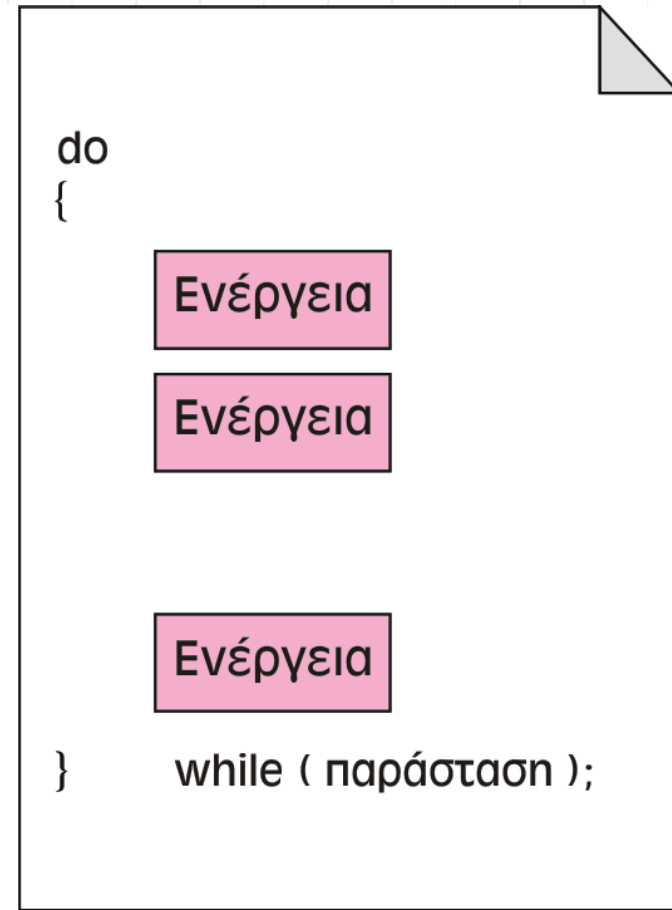


β. Κώδικας γλώσσας C

Επανάληψη



α. Διάγραμμα ροής

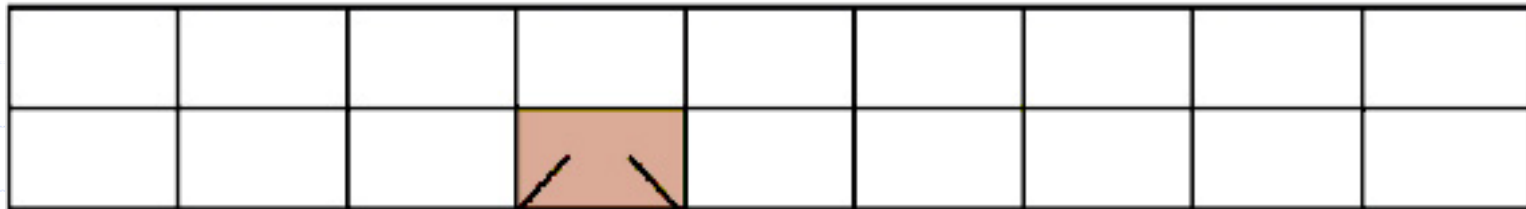


β. Κώδικας γλώσσας C

Δομές Δεδομένων

◆ Ομοιογενής Πίνακας

Scores



Scores (2 , 4) στη
FORTRAN όπου οι
δείκτες ξεκινούν
από το ένα

Scores [1] [3] στη C
και τα παράγωγα της,
όπου οι δείκτες ξεκινούν
από το μηδέν.

Ετερογενής Πίνακας

α. Η δήλωση του πίνακα

```
struct  
{ char Name [8];  
  int Age;  
  float SkillRating;  
} Employee;
```

β. Η νοητική διάταξη του πίνακα

