



ΔΙΕΘΝΕΣ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΤΗΣ ΕΛΛΑΔΟΣ

Τμήμα Μηχανικών Πληροφορικής, Υπολογιστών και Τηλεπικοινωνιών  
Πανεπιστημιούπολη Σερρών

## Προγραμματισμός Ι (Θ)

Δρ. Δημήτρης Βαρσάμης  
Αναπληρωτής Καθηγητής

Οκτώβριος 2019

# ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ Ι (Θ)

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές
- 4 Λογικοί τελεστές
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

# Contents

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές
- 4 Λογικοί τελεστές
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

- Οι εκφράσεις είναι παραστάσεις με πράξεις και διεργασίες οι οποίες μας επιστρέφουν ένα αποτέλεσμα.
- Δύο είναι οι κύριες εκφράσεις:
  - ▶ Οι αριθμητικές εκφράσεις, οι οποίες επιστρέφουν αριθμητικό αποτέλεσμα
  - ▶ Οι λογικές εκφράσεις, οι οποίες επιστρέφουν λογικό αποτέλεσμα
- Μια έκφραση αποτελείται στην πιο απλή της μορφή από τον τελεστή και τους τελεστέους.
- Η σύνταξη της έκφρασης είναι  
τελεστήος τελεστής τελεστήος

- Τελεστέος μπορεί να είναι μια σταθερά, μια μεταβλητή, μια συνάρτηση ή μια έκφραση που να περιέχει όλα τα προηγούμενα.
- Τελεστής είναι ένα σύμβολο το οποίο επιτελεί μια πράξη μεταξύ των δυο τελεστέων.
- Θα πρέπει οι τελεστέοι και ο τελεστής να έχουν συμβατότητα, δηλαδή, να είναι όλα ιδίου τύπου. Δεν μπορεί να γίνει αριθμητική πράξη μεταξύ λογικών αποτελεσμάτων.
- Κάποιοι τελεστές είναι μοναδιαίοι, δηλαδή, επιδρούν σε ένα μόνο τελεστέο, είτε από δεξιά είτε από αριστερά.
- Η σύνταξη της έκφρασης είναι  
τελεστικής τελεστέος  
ή  
τελεστέος τελεστικής

- Η προτεραιότητα στις πράξεις ακολουθεί συγκεκριμένους κανόνες ανάλογα τον τύπο τελεστή που χρησιμοποιούμε
- Η προτεραιότητα στις πράξεις μπορεί να καθοριστεί με τη χρήση παρενθέσεων

- Για παράδειγμα, η έκφραση

$τελεστέος1 \ τελεστής1 \ τελεστέος2 \ τελεστής2 \ τελεστέος3$

δεν μας δίνει το ίδιο αποτέλεσμα με την έκφραση

$τελεστέος1 \ τελεστέος2 \ (τελεστέος3 \ τελεστής2 \ τελεστέος4)$

εάν ο  $τελεστής1$  έχει μεγαλύτερη προτεραιότητα από τον  $τελεστής2$

# Contents

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές
- 4 Λογικοί τελεστές
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

# Αριθμητικοί τελεστές

- Αριθμητικοί Τελεστές στην C

Τελεστής	πράξη
+	πρόσθεση
-	αφαίρεση
*	πολλαπλασιασμός
/	πηλίκο ακέραιας διαίρεσης εάν οι τελεστέοι είναι ακέραιοι ή πηλίκο εάν ένας από τους τελεστέους είναι πραγματικός
%	υπόλοιπο ακέραιας διαίρεσης
++	μοναδιαία αύξηση ακεραίου (επιδρά σε ένα τελεστέο)
--	μοναδιαία μείωση ακεραίου (επιδρά σε ένα τελεστέο)



# Αριθμητικοί τελεστές

- Η προτεραιότητα στις αριθμητικές πράξεις ορίζεται από τα μαθηματικά.
- Οι πράξεις ίδιας προτεραιότητας εκτελούνται από αριστερά προς τα δεξιά

# Αριθμητικοί τελεστές

Προτεραιότητα στις αριθμητικές πράξεις

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int x, y, z;
5 |     x=7/5*3+4;
6 |     y=7%5/2*7;
7 |     z=3*(5+8*2);
8 |     printf(" x= %d\n y= %d\n z= %d\n", x, y, z);
9 |     return 0;
10| }
```

# Αριθμητικοί τελεστές

Προτεραιότητα στις αριθμητικές πράξεις

$$x = 7$$

$$y = 7$$

$$z = 63$$

# Αριθμητικοί τελεστές

Πράξεις μεταξύ αριθμών διαφόρων τύπων

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int x,y;
5 |     float z,w;
6 |     x=7/5;
7 |     y=7%5;
8 |     z=7.0/5.0;
9 |     w=7/5;
10 |    printf(" x= %d\n y= %d\n z= %f\n w= %f\n",x,y,z
    |        ,w);
11 |    return 0;
12 | }
```

Προσοχή στη χρήση του τελεστή /

# Αριθμητικοί τελεστές

Πράξεις μεταξύ αριθμών διαφόρων τύπων. Προσοχή στη χρήση του τελεστή /

```
x= 1
```

```
y= 2
```

```
z= 1.400000
```

```
w= 1.000000
```

# Αριθμητικοί τελεστές I

Πράξεις μεταξύ μοναδιαίων τελεστών αύξησης ή μείωσης

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int x=10,y=20;
5 |     printf("x= %d y= %d\n",x,y);
6 |     ++x;
7 |     printf("x= %d y= %d\n",x,y);
8 |     y = --x;
9 |     printf("x= %d y= %d\n",x,y);
10 |    y = x-- + y;
11 |    printf("x= %d y= %d\n",x,y);
12 |    y = y - x++;
13 |    printf("x= %d y= %d\n",x,y);
14 |    return 0;
```

# Αριθμητικοί τελεστές II

15 || }

# Αριθμητικοί τελεστές

Πράξεις μεταξύ μοναδιαίων τελεστών αύξησης ή μείωσης

x= 10 y= 20

x= 11 y= 20

x= 10 y= 10

x= 9 y= 20

x= 10 y= 11



# Αριθμητικοί τελεστές I

Προσωρινή αλλαγή τύπου μεταβλητής

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int x=7,y=5;
5 |     float z,w;
6 |     z=(float)x/y;
7 |     w=x/(y*1.0);
8 |     printf(" x= %d\n y= %d\n z= %f\n w= %f\n",x,y,z
   |         ,w);
9 |     system("PAUSE");
10 |     return 0;
11 | }
```

# Αριθμητικοί τελεστές II

- Μπορούμε να αλλάξουμε προσωρινά τον τύπο μιας μεταβλητής με την βοήθεια του `tyrecasting`
- Χρησιμοποιούμε τον τύπο που θέλουμε μέσα σε παρενθέσεις και επιδρά στην μεταβλητή ή έκφραση που ακολουθεί.

# Αριθμητικοί τελεστές

Προσωρινή αλλαγή τύπου μεταβλητής

```
x= 7
```

```
y= 5
```

```
z= 1.400000
```

```
w= 1.400000
```

# Contents

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές**
- 4 Λογικοί τελεστές
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

# Συγκριτικοί τελεστές

- Συγκριτικοί Τελεστές στην C

Τελεστής	πράξη
>	
<	
>=	
<=	
==	έλεγχος ισότητας
!=	έλεγχος μη ισότητας

# Συγκριτικοί τελεστές

- Οι συγκριτικές πράξεις στην C επιστρέφουν 1 όταν το αποτέλεσμα της πράξης είναι ΑΛΗΘΗΣ και 0 όταν το αποτέλεσμα της πράξης είναι ΨΕΥΔΗΣ.
- Το γεγονός ότι τα λογικά αποτελέσματα στην C είναι 0 ή 1 μπορεί να προκαλέσει σύγχυση σε συνδυασμό με αριθμητικές πράξεις και σε μη ορθά αποτελέσματα.
- Πρώτα εκτελούνται οι αριθμητικές πράξεις και έπειτα οι συγκριτικές πράξεις.

# Συγκριτικοί τελεστές

Προτεραιότητα στις συγκριτικές πράξεις

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int x, y, z, w;
5 |     x=7>=6;
6 |     y=7%5>3;
7 |     z=3+5!=5-8;
8 |     w=3<5>=2;
9 |     printf(" x= %d\n y= %d\n z= %d\n w= %d\n", x, y
    |         , z, w);
10 |     return 0;
11 | }
```

Προσοχή στις πράξεις στην γραμμή 8

# Συγκριτικοί τελεστές

Προτεραιότητα στις συγκριτικές πράξεις

x= 1

y= 0

z= 1

w= 0



# Contents

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές
- 4 Λογικοί τελεστές**
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

# Λογικοί τελεστές

- Λογικοί Τελεστές στην C

Τελεστής	πράξη
&&	σύζευξη
	διάζευξη
!	άρνηση

- Οι λογικές πράξεις στην C επιστρέφουν 1 όταν το αποτέλεσμα της πράξης είναι ΑΛΗΘΗΣ και 0 όταν το αποτέλεσμα της πράξης είναι ΨΕΥΔΗΣ.
- Το γεγονός ότι τα λογικά αποτελέσματα στην C είναι 0 ή 1 μπορεί να προκαλέσει σύγχυση σε συνδυασμό με αριθμητικές πράξεις και σε μη ορθά αποτελέσματα.
- Πρώτα εκτελούνται οι αριθμητικές πράξεις έπειτα οι συγκριτικές πράξεις και τέλος οι λογικές πράξεις.

# Λογικοί τελεστές

- Πίνακας Αληθείας

$p$	$q$	$p \& \& q$	$p \mid \mid q$	$!p$
1	1	1	1	0
1	0	0	1	0
0	1	0	1	1
0	0	0	0	1

# Λογικοί τελεστές

Προτεραιότητα στις λογικές πράξεις

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int x, y, z, w, v;
5 |     x=7>=6&&8<9;
6 |     y=7%5>3||5!=8;
7 |     z=!(3+5!=5-8);
8 |     w=3<5>=2;
9 |     v=3<5&&5>=2;
10 |    printf(" x= %d\n y= %d\n z= %d\n w= %d\n v= %
    |         d\n", x, y, z, w, v);
11 |    return 0;
12 | }
```

Προσοχή στις πράξεις στην γραμμή 8

# Λογικοί τελεστές

Προτεραιότητα στις λογικές πράξεις

x= 1

y= 1

z= 0

w= 0

v= 1

# Contents

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές
- 4 Λογικοί τελεστές
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

# Τελεστές Ανάθεσης

- Τελεστές Ανάθεσης στην C

Τελεστής	Παράδειγμα	πράξη
=	a=b	a=b
+=	a+=b	a=a+b
-=	a-=b	a=a-b
*=	a*=b	a=a*b
/=	a/=b	a=a/b
%=	a%=b	a=a%b

# Τελεστές Ανάθεσης I

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | int main() {
4 |     int a = 5, c;
5 |     c = a;
6 |     printf("c = %d \n", c);
7 |     c += a;
8 |     printf("c = %d \n", c);
9 |     c -= a;
10 |    printf("c = %d \n", c);
11 |    c *= a;
12 |    printf("c = %d \n", c);
13 |    c /= a;
14 |    printf("c = %d \n", c);
15 |    c %= a;
```



## Τελεστές Ανάθεσης II

```
16 | printf("c = %d \n", c);  
17 | return 0;  
18 | }
```

# Τελεστές Ανάθεσης

$c = 5$

$c = 10$

$c = 5$

$c = 25$

$c = 5$

$c = 0$

# Contents

- 1 Εκφράσεις
- 2 Αριθμητικοί τελεστές
- 3 Συγκριτικοί τελεστές
- 4 Λογικοί τελεστές
- 5 Τελεστές Ανάθεσης
- 6 Μαθηματικές Συναρτήσεις

# Μαθηματικές Συναρτήσεις I

## Βιβλιοθήκη `math.h`

- `double cos(double x)`  
Returns the cosine of a radian angle  $x$ .
- `double sin(double x)`  
Returns the sine of a radian angle  $x$ .
- `double exp(double x)`  
Returns the value of  $e$  raised to the  $x^{\text{th}}$  power.
- `double log(double x)`  
Returns the natural logarithm (base- $e$  logarithm) of  $x$ .
- `double log10(double x)`  
Returns the common logarithm (base-10 logarithm) of  $x$ .
- `double pow(double x, double y)`  
Returns  $x$  raised to the power of  $y$ .

# Μαθηματικές Συναρτήσεις II

- `double sqrt(double x)`  
Returns the square root of  $x$ .
- `double ceil(double x)`  
Returns the smallest integer value greater than or equal to  $x$ .
- `double fabs(double x)`  
Returns the absolute value of  $x$ .
- `double floor(double x)`  
Returns the largest integer value less than or equal to  $x$ .

# Μαθηματικές Συναρτήσεις

Τετραγωνική ρίζα, Ύψωση σε δύναμη, Λογάριθμος, e

```
1 | #include <stdio.h>
2 | #include <stdlib.h>
3 | #include <math.h>
4 | int main() {
5 |     int x;
6 |     float y, z;
7 |     x=pow(4,2);
8 |     y=sqrt(2);
9 |     z=log(exp(2));
10 |    printf(" x= %d\n y= %f\n z= %f\n",x,y,z);
11 |    return 0;
12 | }
```

$x = 16$

$y = 1.414214$

$z = 2.000000$