

# Προγραμματισμός I

## Αριθμητική

Δημήτρης Μιχαήλ



Τμήμα Πληροφορικής και Τηλεματικής  
Χαροκόπειο Πανεπιστήμιο

## Δεύτερο Πρόγραμμα

```
1  /* Second Simple Program: add 2 numbers */
2  #include <stdio.h>
3
4  int main() {
5      int integer1, integer2;
6      int sum;
7
8      printf("Enter first integer\n");
9      scanf("%d", &integer1);
10
11     printf("Enter second integer\n");
12     scanf("%d", &integer2);
13
14     sum = integer1 + integer2;
15     printf("Sum is %d\n", sum);
16
17     return 0;
18 }
```

## Η συνάρτηση `scanf()`

Ορισμένη στο `stdio.h` κάνει την αντίστροφη διαδικασία από την `printf()`.

Διαβάζει από τον χρήστη κάποια πληροφορία. Η εκτέλεση της παρακάτω γραμμής:

```
9 scanf("%d", &integer1);
```

σταματάει το πρόγραμμα μέχρι ο χρήστης να γράψει έναν αριθμό (ακέραιο) και να πατήσει ENTER.

# Η συνάρτηση `scanf()`

specifier	Έισοδος	Παράδειγμα
d	διάβασε ένα ακέραιο ως προσημασμένο δεκαδικό	392
c	χαρακτήρας	a
f	δεκαδικός αριθμός κινητής υποδιαστολής	392.65
x	δεκαεξαδικός αριθμός χωρίς πρόσημο	7fa
o	οκταδικός αριθμός	610
s	αλφαριθμητικό	sample
(space)	διαβάζει κενά	
If	διαβάζει έναν <b>double</b>	1.333
Lf	διαβάζει έναν <b>long double</b>	1.333

# Αριθμητική στην C

Τα περισσότερα προγράμματα σε C εκτελούν μαθηματικούς υπολογισμούς μεταξύ σταθερών και μεταβλητών, π.χ

```
1 int x, y;  
2  
3 x = 1;  
4 y = x + 100;
```

# Διαδικοί Αριθμητικοί τελεστές

	τελεστής	αλγεβρική έκφραση	C
πρόσθεση	+	$x + 7$	<b>x + 7</b>
αφαίρεση	-	$p - c$	<b>p - c</b>
πολλαπλασιασμός	*	$bm$	<b>b * m</b>
διαίρεση	/	$x/y$ ή $x \div y$	<b>x / y</b>
υπόλοιπο διαίρεσης	%	$r \text{ mod } s$	<b>r % s</b>

Οι παρενθέσεις χρησιμοποιούνται στην C με τον ίδιο περίπου τρόπο που χρησιμοποιούνται και στις αλγεβρικές εκφράσεις, π.χ

```
1 a = b * ( c + d );
```

# Προτεραιότητα Τελεστών

Ο τρόπος υπολογισμού μιας έκφρασης εξαρτάται από την προτεραιότητα των τελεστών:

- 1 παρενθέσεις:** ()  
Υπολογίζονται πρώτα, από τα αριστερά προς τα δεξιά.  
Εάν υπάρχουν ένθετες υπολογίζονται πρώτα οι εσωτερικές.
- 2 πολλαπλασιασμός, διαίρεση και υπόλοιπο:** \*, /, ή %  
Υπολογίζονται δεύτερα από αριστερά προς τα δεξιά.
- 3 πρόσθεση, αφαίρεση:** + ή -  
Εάν υπάρχουν πολλοί, υπολογίζονται από τα αριστερά προς τα δεξιά.
- 4 εκχώρησης:** =  
Από δεξιά προς τα αριστερά.

# Παραδείγματα Εκφράσεων

$$m = \frac{a + b + c + d + e}{5}$$

σε γλώσσα C

$$\mathbf{m = (a+b+c+d+e)/5;}$$



# Παραδείγματα Εκφράσεων

$$y = ax^2 + bx + c$$

σε γλώσσα C

$$\mathbf{y = a * x * x + b * x + c}$$

# Προτεραιότητα Τελεστών

$$z = p * r \% q + w / x - y$$

⑥ ① ② ④ ③ ⑤

# Μοναδιαίοι Αριθμητικοί τελεστές

- +  
πχ  
 $y = +5;$
- -  
πχ  
 $x = -y;$
- ++  
πχ  
 $x = ++y$   
ή  
 $x = y++$
- --  
πχ  
 $x = --y$   
ή  
 $x = y--$

Οι μοναδιαίοι αριθμητική τελεστές έχουν μεγαλύτερη προτεραιότητα από τους άλλους αριθμητικούς (εκτός από τις παρενθέσεις) και η σειρά υπολογισμού τους είναι από δεξιά προς τα αριστερά.

## Τελεστές ++ -

Ο τελεστής ++ αυξάνει μία μεταβλητή κατά 1 και ο - την μειώνει κατά 1.

```
1  int main() {
2      int x;
3
4      x = 1;
5      printf("%d\n", x); /* prints 1 */
6      x++;
7      printf("%d\n", x); /* prints 2 */
8      ++x;
9      printf("%d\n", x); /* prints 3 */
10     x--;
11     printf("%d\n", x); /* prints 2 */
12     --x;
13     printf("%d\n", x); /* prints 1 */
14 }
```

## Τελεστές ++ -

Οι εκφράσεις `x++` και `++x` έχουν διαφορά μόνο όταν υπάρχει χρήση της τιμής της έκφρασης (π.χ εκχώρηση σε άλλη μεταβλητή, κλήση συνάρτησης, κ.τ.λ.).

Σε αυτήν την περίπτωση η μεταβλητή `x` αυξάνει μετά ή πριν από την χρήση.

## Τελεστές ++ -

```
1  int main() {
2      int x, y;
3
4      x = 1;
5      y = x++;
6      printf("x=%d, y=%d\n", x, y);    /* prints x=2, y=1 */
7      y = ++x;
8      printf("x=%d, y=%d\n", x, y);    /* prints x=3, y=3 */
9      y = x--;
10     printf("x=%d, y=%d\n", x, y);    /* prints x=2, y=3 */
11     y = --x;
12     printf("x=%d, y=%d\n", x, y);    /* prints x=1, y=1 */
13     y = y + x--;
14     printf("x=%d, y=%d\n", x, y);    /* prints x=0, y=2 */
15     y = y + ++x;
16     printf("x=%d, y=%d\n", x, y);    /* prints x=1, y=3 */
17 }
```

## Τελεστές ++ -

Τι εκτυπώνει το παρακάτω πρόγραμμα;

```
1  int main() {
2      int x, y;
3
4      x = 3;
5      y = x++;
6
7      printf("%d %d\n", x++, ++y);
8      printf("%d %d\n", ++x, ++y);
9      printf("%d %d\n", y++ + ++x, --y + --x);
10     printf("%d %d\n", ++y + --x, --y - ++x);
11 }
```

# Προτεραιότητα Τελεστών

Ο τρόπος υπολογισμού μιας έκφρασης εξαρτάται από την προτεραιότητα των τελεστών:

- 1 παρενθέσεις:**  $()$ ,  $\text{expr}++$  ή  $\text{expr}--$   
Υπολογίζονται πρώτα, από τα αριστερά προς τα δεξιά.  
Εάν υπάρχουν ένθετες υπολογίζονται πρώτα οι εσωτερικές.
- 2 μοναδιαίοι αριθμητικοί τελεστές:**  
 $+$ ,  $-$ ,  $++\text{expr}$  ή  $--\text{expr}$  (prefix) Υπολογίζονται από δεξιά προς τα αριστερά.
- 3 πολλαπλασιασμός, διαίρεση και υπόλοιπο:**  
 $*$ ,  $/$ , ή  $\%$  Υπολογίζονται από αριστερά προς τα δεξιά.
- 4 πρόσθεση, αφαίρεση:**  $+$  ή  $-$   
Εάν υπάρχουν πολλοί, υπολογίζονται από τα αριστερά προς τα δεξιά.
- 5 εκχώρησης:**  $=$   
Από δεξιά προς τα αριστερά.



## Τελεστές Εκχώρησης

Ο τελεστής εκχώρησης = θέλει προσοχή γιατί εκτελείται από τα δεξιά προς τα αριστερά.

Επίσης η έκφραση  $x = 5$  επιστρέφει την τιμή που καταχωρήθηκε.

```
1 int main() {  
2     int x,y;  
3  
4     y = x = 5;  
5 }
```

Στο παραπάνω πρόγραμμα πρώτα εκτελείται το  $x=5$  και επιστρέφει την τιμή 5 οπότε και εκτελείται μετά το  $y = 5$ .

# Τελεστές Εκχώρησης

Η C παρέχει διαφόρους τελεστές εκχώρησης για την συντόμευση των εκφράσεων εκχώρησης.

τελεστής	πρόταση C	ισοδύναμη πρόταση C
<code>+=</code>	<code>c += 7</code>	<code>c = c + 7</code>
<code>-=</code>	<code>d -= 4</code>	<code>d = d - 4</code>
<code>*=</code>	<code>e *= 5</code>	<code>e = e * 5</code>
<code>/=</code>	<code>f /= 3</code>	<code>f = f / 3</code>
<code>%=</code>	<code>g %= 9</code>	<code>g = g % 9</code>

# Προτεραιότητα Τελεστών

Ο τρόπος υπολογισμού μιας έκφρασης εξαρτάται από την προτεραιότητα των τελεστών:

- 1 παρενθέσεις:**  $()$ ,  $\mathit{expr}++$  ή  $\mathit{expr}--$   
Υπολογίζονται πρώτα, από τα αριστερά προς τα δεξιά.  
Εάν υπάρχουν ένθετες υπολογίζονται πρώτα οι εσωτερικές.
- 2 μοναδιαίοι αριθμητικοί τελεστές:**  $+$ ,  $-$ ,  $++\mathit{expr}$  ή  $--\mathit{expr}$   
Υπολογίζονται από δεξιά προς τα αριστερά.
- 3 πολλαπλασιασμός, διαίρεση και υπόλοιπο:**  $*$ ,  $/$ , ή  $\%$   
Υπολογίζονται από αριστερά προς τα δεξιά.
- 4 πρόσθεση, αφαίρεση:**  $+$  ή  $-$   
Εάν υπάρχουν πολλοί, υπολογίζονται από τα αριστερά προς τα δεξιά.
- 5 εκχώρησης:**  $=$ ,  $+=$ ,  $-=$ ,  $*=$ ,  $/=$ ,  $\%=$   
Από δεξιά προς τα αριστερά.