



Κεφάλαιο 2.5:

Τύποι Δεδομένων, Τελεστές και Αριθμητικές Εκφράσεις

(Διαλέξεις 5-6)

Διδάσκων: Δημήτρης Ζεϊναλιπούρ

Περιεχόμενα



- Τύποι Δεδομένων
int, char, float, double
- Τελεστές
=, +, -, *, /, %
- Αριθμητικές εκφράσεις
a+b

Τύποι Δεδομένων



- Ένας τύπος δεδομένων είναι ένα **σύνολο τιμών** και ένα **σύνολο λειτουργιών** (πράξεων) που μπορούν να εφαρμοστούν σε αυτές τις τιμές
- βασικοί τύποι δεδομένων:
(char, int, float, double)
- σύνθετοι τύποι δεδομένων: (arrays, structs)
π.χ.

```
struct Time {  
    int hours; int minutes; int seconds;  
};
```

Τύποι δεδομένων - int



Τύπος int (ακέραιος – integer)

- Αναπαριστά ακέραιους αριθμούς,
- **Μέγεθος 4 bytes**, πεδίο τιμών $-2^{31} .. +(2^{31}-1)$
δηλαδή περίπου : από $[-2G .. +2G]$
 $[-(\text{δυο δισεκατομμύρια} .. +\text{δυο δισεκατομμύρια})]$
- **Πράξεις**: πρόσθεση (+), αφαίρεση (-),
πολλαπλασιασμός (*),
διαίρεση (/) κλπ.

Παραδείγματα κυριολεκτικών τιμών:

-2189456 0 50 +24562 -3245 13576313

Τύποι δεδομένων - float



Τύπος float (κινητής υποδιαστολής **απλής** ακρίβειας – floating point)

- Αναπαριστά τους πραγματικούς αριθμούς,
- **Μέγεθος 4 bytes,**
- Πράξεις: πρόσθεση (+), αφαίρεση (-), πολλαπλασιασμός (*), διαίρεση (/)

Τιμές μιας μεταβλητής float μπορεί να είναι:
π.χ 3.01, 110.8, -0.01, κλπ.

Τύποι δεδομένων - double



Τύπος double (κινητής υποδιαστολής διπλής ακρίβειας – double precision)

- Ίδιος τύπος με float αλλά με μεγαλύτερη ακρίβεια (περιέχει δηλ. διπλό αριθμό δεκαδικών ψηφίων απ' ότι η μεταβλητή float),
- **Μέγεθος 8 bytes.**

Τιμή μιας μεταβλητής double μπορεί να είναι:
π.χ 1.045623,

Τύποι δεδομένων - char



Τύπος char (χαρακτήρας – character)

- Αναπαριστά ατομικούς χαρακτήρες (A-Z, a-z, 0-9, !@\$%&#, ειδικά σύμβολα \n, κλπ.)
- Μέγεθος 1 byte.
- Κυριολεκτικές τιμές εσωκλείονται σε αποστρόφους, π.χ. 'A', 'a', '9', '"', ' ', '*', '\n', '\', κτλ
- Οι απόστροφοι δεν χρειάζονται όταν εισάγονται χαρακτήρες

Για παράδειγμα:

'a' είναι ο χαρακτήρας a
'b' είναι ο χαρακτήρας b
'9' είναι ο χαρακτήρας 9
'*' είναι ο χαρακτήρας *



char

- κάθε χαρακτήρας αντιστοιχεί σε ένα μοναδικό κωδικό
- C βασίζεται στο ASCII code
- αλφαβητικοί, ψηφιακοί, ειδικοί ('\n', '\t'...)
 - '0' ascii:48, '1' ascii:49,..., '9' ascii:57
 - 'A' ascii:65, ..., 'Z' ascii:90
 - 'a' ascii:97, ..., 'z' ascii:122

Ο Πίνακας ASCII



Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Spa	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	\$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DE

Βασικοί Τύποι Δεδομένων C

Περίληψη



<u>Τύπος</u>	<u>Μέγεθος</u>	<u>Πεδίο Τιμών</u>	<u>Μοναδικές</u>
char,	1 byte	'a'..'z' 'A'..'Z' '0'..'9'	2^8 ή 256
int,	4 bytes	$-2^{31}..2^{31}-1$	2^{32}
float,	4 bytes	$10^{-37}..10^{38}$	2^{32}
double	8 bytes	$10^{-307}..10^{308}$	2^{64}
*δείκτης	4 bytes	διευθύνσεις ($0..2^{32}-1$)	2^{32}

* Δεν θα μελετηθεί σε αυτό το μάθημα

Τύποι δεδομένων (Παράδειγμα)



```
/*program example2.c
this program adds three integer variables and displays the results*/
#include <stdio.h>    /*use of pre-processor*/
int main ( )
{
    /*declaration of variables*/
    int val1;
    int val2;
    int val3;
    int sum;
    /*assign values*/
    val1=1;
    val2=2;
    val3=3;
    /*compute sum*/
    sum=val1+val2+val3;
    /*display sum*/
    printf ("The sum of %d and %d and %d is %d\n", val1, val2, val3, sum);
    return 0;
}
```

Τύποι δεδομένων (Παράδειγμα)



```
/*program example3.c
this program adds three integer variables and displays the results*/
#include <stdio.h>    /*use of pre-processor*/
main ( )
{
    int val1=1, val2=2, val3=3, sum;
    /*compute sum*/
    sum=val1+val2+val3;
    printf ("The sum of %d and %d and %d is %d\n", val1, val2, val3, sum);
/*use of printf*/
}
```

The sum of 1 and 2 and 3 is 6

Επιλογή Τύπου Δεδομένων



- Μαθητές σε ένα σχολείο
- Βάρος, Μάζα
- Εμβαδό, Όγκος
- Όνομα
- Αριθμός Ταυτότητας

Επιλογή Τύπου Δεδομένων



- Μαθητές σε ένα σχολείο int
- Βάρος, Μάζα float, double
- Εμβαδό, Όγκος float, double
- Όνομα char (string)
- Αριθμός Ταυτότητας int

Αριθμητικές Εκφράσεις



- Σύνταξη: **a τ b** ή **τ a**
 - **τ** είναι ο **τελεστής (operator)**
 - **a, b** είναι **τελεσταίοι (operands)**
 - τελεσταίοι μπορεί να είναι
 - Σταθερές (π.χ. **KMS_PER_MILE * miles**)
 - Μεταβλητές (π.χ. **c = a + b**)
 - κλήση συνάρτησης που επιστρέφει αριθμό (π.χ **c = sum(a,b) + sum(b,a)**)
 - έκφραση (χρήση παρενθέσεων)

Αριθμητικοί Τελεστές (arithmetic operators)



Όνομα	Τελεστής	Παράδειγμα
Πρόσθεση	+	num1 + num2
Αφαίρεση	-	initial - spent
Πολλ/σμός	*	age * 6
Διαίρεση	/	sum / count
Υπόλοιπο	%	m % n

Τελεστής Ανάθεσης (=) (assignment operator)



- Σύνταξη:

μεταβλητή = έκφραση;

- `area = PI * radius * radius;`
 - `count = count + 1;`
 - `new_number = old_number;`
 - `average = total / count;`
-
- Η τιμή της έκφρασης αποθηκεύεται στην μεταβλητή

Τύπος Έκφρασης



- Ορίζεται από τους τύπους των τελεστών
- char, int, float, double

<u>Έκφραση</u>	<u>Αποτέλεσμα</u>	π.χ	
int τ int	=> int	5/2	2
double τ double	=> double	5.0/2.0	2.5
int τ double	=> double	5/2.0	2.5
double τ int	=> double	5.0/2	2.5
int τ char	=> int	5+'a'	102

↓
Ascii:97



Μετατροπή Εκφράσεων σε C

- $m^2 - n^2$
- $ax^2 + bx + c$
- $-b + 4ac$
- $\frac{2ab}{c+d}$
- $((-a)b) + ((-c)d)$



Μετατροπή Εκφράσεων σε C

Στην γλώσσα C

- $m^2 - n^2$ $m*m - n*n$ ή $(m*m)-(n*n)$
- $ax^2 + bx + c$ $a*x*x + b*x + c$
- $-b + 4ac$ $-b + 4 * a * c$
- $\frac{2ab}{c+d}$ $(2*a*b)/(c+d)$
- $((-a)b) + ((-c)d)$ $-a*b + -c*d$

Κανόνας Προτεραιότητας Τελεστών



- Τελεστές στην ίδια έκφραση αποτιμούνται σύμφωνα με τις προτεραιότητές τους.
- Οι προτεραιότητες των αριθμητικών τελεστών, σε φθίνουσα σειρά, είναι:



Αριθμητικοί τελεστές και προτεραιότητα

Τελεστές

Ψηλότερη Προτεραιότητα

()

* / %

+ -

=

Χαμηλότερη Προτεραιότητα

Προτεραιότητα

Πρώτο στην αποτίμηση. Εάν είναι εμφωλιασμένο (το εσώτατο πρώτα). Εάν είναι στο ίδιο επίπεδο, από **αριστερά στα δεξιά**

Δεύτερο στην αποτίμηση. Εάν υπάρχουν πολλά, από **αριστερά στα δεξιά**

Τρίτο στην αποτίμηση. Εάν υπάρχουν πολλά, από **αριστερά στα δεξιά**.

Αποτιμάται τελευταίο, από **δεξιά στα αριστερά**

Κανόνας Εταιρικότητας



- **Μοναδιαίοι τελεσταίοι** (π.χ. $-α$) στην ίδια υποέκφραση και με την ίδια προτεραιότητα αποτιμώνται από τα **δεξιά** προς τα **αριστερά** – (Ονομάζεται **δεξιά εταιρικότητα**)
- **Διαδικοί τελεσταίοι** ($α - β$) στην ίδια υποέκφραση και με την ίδια προτεραιότητα αποτιμώνται από τα **αριστερά** προς τα **δεξιά** (Ονομάζεται **αριστερή εταιρικότητα**)

Κανόνας Παρενθέσεων



- Κάθε έκφραση σε παρένθεση αποτιμείται ξεχωριστά. Φωλιασμένες παρενθετικές εκφράσεις, $((..))$, αποτιμώνται από τα μέσα προς τα έξω
- Όπου είμαστε αβέβαιοι χρησιμοποιούμε παρενθέσεις

Παράδειγμα Αποτίμησης



$$x * y * z + a / b - c * d$$



$$(((x * y) * z) + (a / b)) - (c * d)$$

- Και τα δυο είναι σωστά στην γλώσσα C.
- Ωστόσο η δεύτερη έκφραση εκφράζει πιο ξεκάθαρα την σειρά εκτέλεσης

Παράδειγμα Αποτίμησης



$$x = 5; y = 3; z = -4;$$

Έκφραση	Τιμή
$(x-y)*z$	-8
$x - (y * z)$	17
$x - y * z$	17



Εξάσκηση στην αποτίμηση εκφράσεων

Έστω οι ακέραιοι $a, b, c, d,$ & $e,$

όπου $a = 1, b = 2, c = 3, d = 4$

Τι βγάζουν τα παρακάτω;

$$a + b - c + d \quad \Rightarrow \quad 4$$

$$2\%3 = 2 \quad a * b / c \quad \Rightarrow \quad 0$$

$$4\%2 = 0 \quad 1 + \overline{a * b \% c} \quad \Rightarrow \quad 3$$

$$3/2 = 1 \quad a + d \% b - c \quad \Rightarrow \quad -2$$

$$d + c / b - a \quad \Rightarrow \quad 4$$



Υπόλοιπο %

- Η έκφραση $m \% n$ επιστρέφει το υπόλοιπο της διαίρεσης του m με το n .
- Το Modulus είναι ακέραιος τελεστής - και οι δύο τελεσταίοι **πρέπει να είναι ακέραιοι.**
- Π.χ :
 - $17 \% 5 = 2$
 - $6 \% 3 = 0$
 - $9 \% 2 = 1$
 - $5 \% 8 = 5$



Ακέραια διαίρεση

- Εάν και οι δύο τελεσταίοι είναι ακέραιο τότε θα πάρετε ακέραιο ως απάντηση. Το δεκαδικό τμήμα απορρίπτεται.
- Πχ. :
$$17 / 5 = 3$$
$$4 / 3 = 1$$
$$35 / 9 = 3$$
- Εάν ένας εκ των δυο τελεσταίους είναι float (ή double) θα πάρετε float (και αντίστοιχα double) ως αποτέλεσμα



Διαίρεση με το 0

- Δεν ορίζεται διαίρεση με 0 στα μαθηματικά.
- Αν επιτρέψετε διαίρεση με το 0 σε ένα πρόγραμμα θα προκληθεί **σφάλμα**.
- Η εκτέλεση του προγράμματος θα τερματιστεί απότομα – **runtime error**

- Θα μάθουμε πώς να αποφεύγουμε την διαίρεση με το 0

Μετατροπή Τύπων



Αυτόματη μετατροπή

- Σε ανάθεση (αυτόματα) η τιμή στα δεξιά του = μετατρέπεται στον τύπο της μεταβλητής στα αριστερά του =

```
int x = 3.14;    /* 3 */
```

```
float x = (2/3); /* 0.0 */
```

Ρητή Μετατροπή (Casting)

```
float x = (float) 2/3;    /* 2.0/3, 2.0/3.0, 0.66666 */
```

```
float x = (float) (2/3); /* 0.0 */
```