

# ΕΠΛ 033: ΕΙΣΑΓΩΓΗ ΣΤΟΝ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟ ΓΙΑ ΜΗΧΑΝΙΚΟΥΣ

Μάριος Belk, Τμήμα Πληροφορικής, Πανεπιστήμιο Κύπρου

Email: [belk@cs.ucy.ac.cy](mailto:belk@cs.ucy.ac.cy)



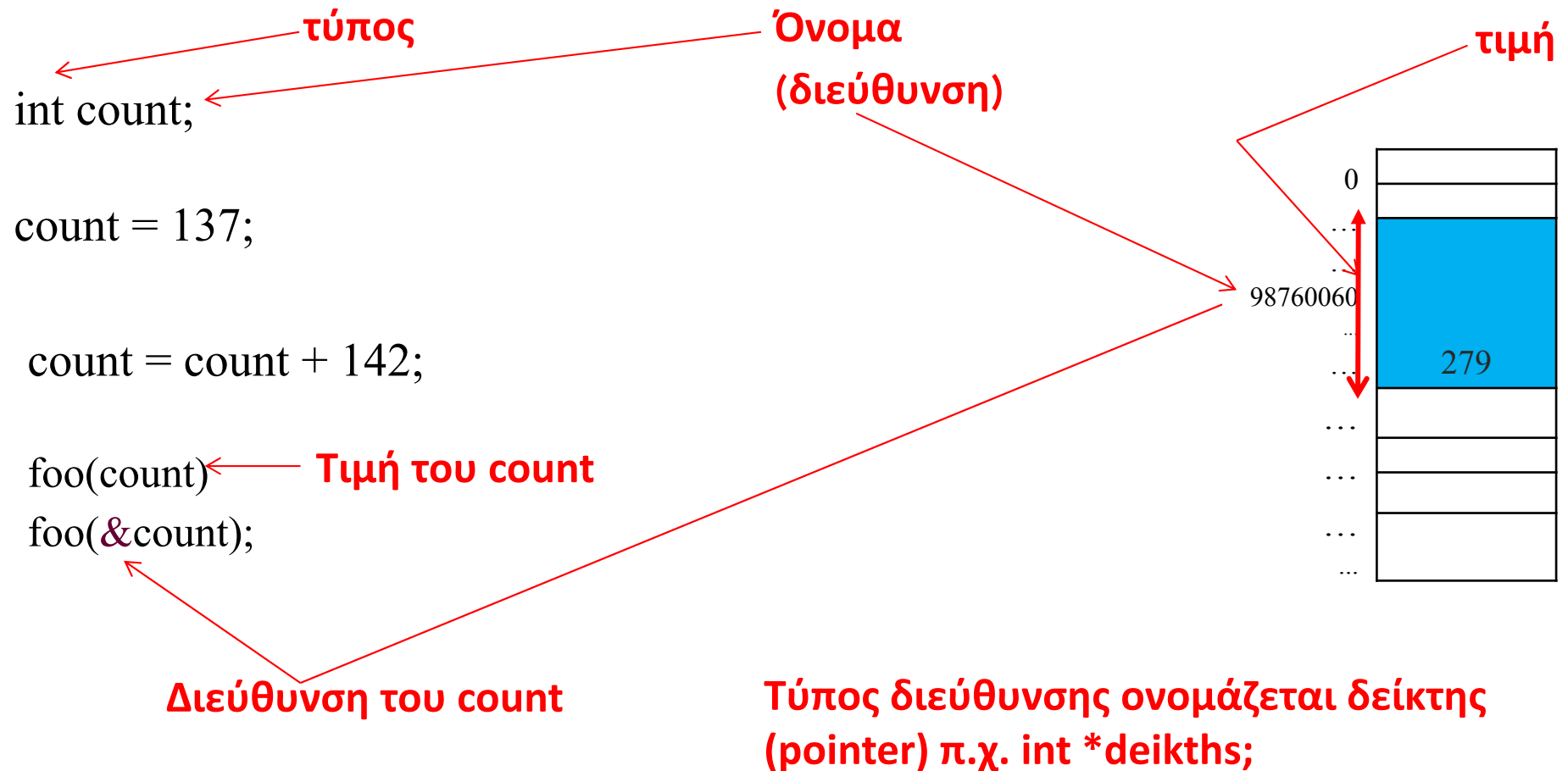
# ΜΕΤΑΒΛΗΤΕΣ

# Σημασία Μεταβλητής (variable)

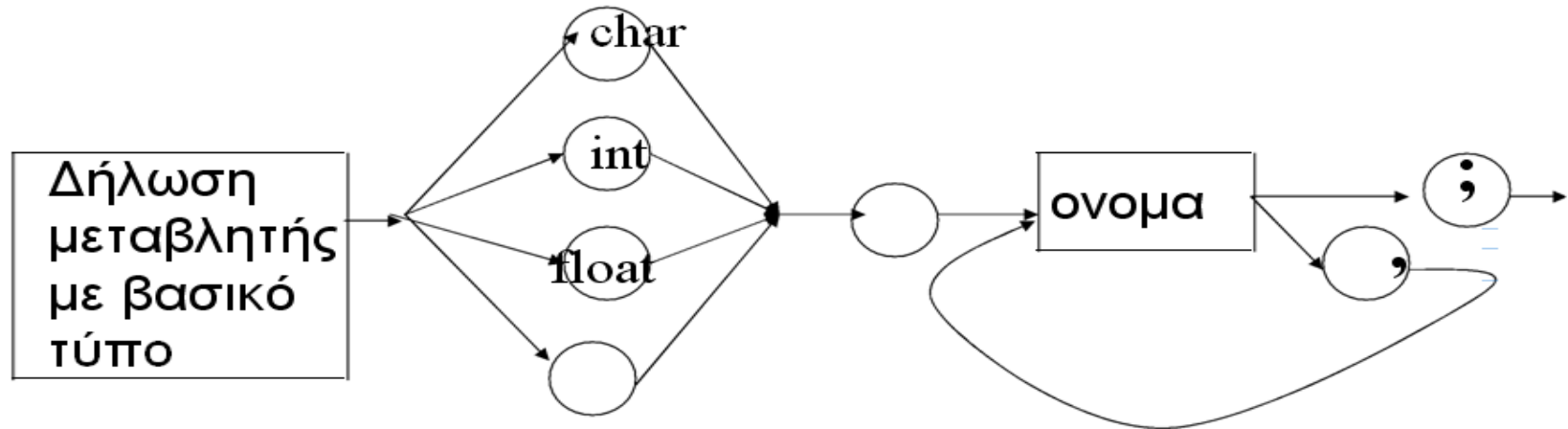
- Αποθήκευση και ανάγνωση τιμών
  - Αριθμών, χαρακτήρων, κ.ά
- Κάθε μεταβλητή χαρακτηρίζεται από:
  - **τύπο**
    - βασικοί: int, char, float, double
    - μέγεθος: 4B, 1B, 4B, 8B (κάθε κυψελίδα 1 Byte)
  - **(συμβολικό) όνομα**   
    - το όνομα αντιστοιχεί σε συγκεκριμένη διεύθυνση στην μνήμη
  - **τιμή**
- Μια μεταβλητή πρέπει να **δηλωθεί** πριν χρησιμοποιηθεί

**Εξαρτάται από το λειτουργικό σύστημα**

# Σημασία Μεταβλητής (συν.)



# Δήλωση Μεταβλητής



Παραδείγματα (σωστό η λάθος?):

- ▣ **char** \$foo; (σωστό)
- ▣ **char** \_\_\_\_f; (σωστό)
- ▣ **int** count, sum; (σωστό)
- ▣ **float** area, **double** volume; (λάθος)
- ▣ **int** index; cost; (λάθος)
- ▣ **double** charge; **float** angle, income; (σωστό)

# Σημασία Σταθεράς (constant)

- Σταθερές έχουν
  - τιμή
  - όνομα **όχι όμως διεύθυνση**
  - προ-επεξεργαστής αντικαθιστά το όνομα με την τιμή
- Διευκολύνει
  - τροποποιήσεις
  - διάβασμα ενός προγράμματος (PI ή 3.1453)
- Μια σταθερά πρέπει να δηλωθεί πριν χρησιμοποιηθεί
- **Δεν** μπορεί να αλλάξει τιμή κατά την εκτέλεση του προγράμματος

# Δήλωση Σταθεράς

6

- Δήλωση Σταθεράς: `#define` → όνομα → τιμή
- `#define PI 3.1453`
- `#define YES 1`
- `#define NO 1`
- `#define TRUE 1`
- `#define PISQUARE PI*PI`
- `#define ERROR_SPACE_MESSAGE "error:run out of space"`

# Τύποι δεδομένων

7

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: (πίνακες, δομές)



# int (ακέραιος)

8

- Αναπαριστά ακεραίους αριθμούς
- Μέγεθος 4 bytes, πεδίο τιμών  $-2^{31}.. +2^{31}-1$ 
  - ( $\approx -2$  δις –  $2$  δις)
- Παραδείγματα κυριολεκτικών τιμών:
  - -2189456 0 50 +24562 -3245 13576313
- Πράξεις: πρόσθεση (+), αφαίρεση (-), πολλαπλασιασμός (\*), διαίρεση (/), υπόλοιπο (%) ....

# float (κινητής υποδιαστολής μονής ακριβείας)

9

- Αναπαριστά τους πραγματικούς αριθμούς
- Μέγεθος: 4 bytes,  $1 \times 10^{38}$ ,  $1 \times 10^{-37}$
- Πράξεις: πρόσθεση (+), αφαίρεση (-), πολλαπλασιασμός (\*), διαίρεση (/)
- Σημειογραφία για κυριολεκτικές τιμές:
  - ▣  $1.258 \times 10^6 = \mathbf{1258000.0} = \mathbf{1.258e6} = \mathbf{1.258E6}$
  - ▣  $8.9 \times 10^{-4} = \mathbf{0.00089} = \mathbf{8.9e-4} = \mathbf{8.9E-4}$

# double (κινητής υποδιαστολής διπλής ακριβείας)

10

- Ίδιος τύπος με float αλλά με μεγαλύτερη ακρίβεια
- Μέγεθος 8 bytes

## Valid double Constants

3.14159

0.005

12345.0

15.0e-04 (0.0015)

2.345e2 (234.5)

1.15e-3 (0.00115)

12e+5 (1200000.0)

## Invalid double Constants

150 (no decimal point)

.12345e (missing exponent)

15e-0.3 (0.3 is invalid exponent)

12.5e.3 (.3 is invalid exponent)

34,500.99 (comma is not allowed)

# Χαρακτήρες (char)

11

- Χειρισμός χαρακτήρων (ατομικών)
  - A-Z, a-z, 0-9, !@\$%&#, ειδικά σύμβολα \n
  - Μέγεθος 1 byte
  - Εσωκλείονται σε απλές (μονές) αποστρόφους
    - 'A', 'a', '9', '"', ' ', '\*', '\n', '\", κτλ
    - Κατά την είσοδο από το πληκτρολόγιο δε χρειάζονται απόστροφοι
- Παραδείγματα
  - 'a' είναι ο χαρακτήρας a
  - 'b' είναι ο χαρακτήρας b
  - '9' είναι ο χαρακτήρας 9
  - '\*' είναι ο χαρακτήρας \*

# Χαρακτήρες

12

- Αντιστοιχούν σε ένα μοναδικό κωδικό
- Στη C ο κωδικός αναφέρεται στον πίνακα ASCII
  - ▣ 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

# Πίνακας ASCII

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

# Οι τύποι δεδομένων περιληπτικά

14

Τύπος	Μέγεθος	Πεδίο Τιμών
char,	1byte	'α'..'z' 'Α'..'Ζ' '0'..'9'
int,	4 bytes	$-2^{31}..2^{31}$
float,	4 bytes	$10^{-37}..10^{38}$
double	8 bytes	$10^{-307}..10^{308}$

# Παράδειγμα

15

```
/*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;
}
```



# Επιλογή του σωστού τύπου δεδομένων

16

Μαθητές σε ένα σχολείο

- int

Βάρος, Μάζα

- float / double

Εμβαδό, Όγκος

- float / double

Χαρακτήρας

- char

Αριθμός ταυτότητας

- int

# Ερωτήσεις Ελέγχου



Τι είναι:

- Μεταβλητές
- Τύπος Δεδομένων
- Ψευδοκώδικας
- Πηγαίος Κώδικας
- Μεταγλωττιστής
- Εκτελέσιμο αρχείο