



ΤΜΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΠΟΛΙΤΙΣΜΙΚΟΥ ΠΕΡΙΒΑΛΛΟΝΤΟΣ
& ΝΕΩΝ ΤΕΧΝΟΛΟΓΙΩΝ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ

Αρχές Γλωσσών Προγραμματισμού (C++)

Διάλεξη 4η

Τελεστές - Casting

Γιάννης Θ. Ντόκας

Π.Δ. 407

Σκοπός της Διάλεξης

2

✓ Να αναλυθούν οι τύποι μεταβλητών, οι τελεστές και η μετατροπή ενός τύπου μεταβλητής σε άλλον τύπο στη C++.

Περιεχόμενα της Διάλεξης

3

- ✓ Τύποι μεταβλητών
- ✓ Τελεστές στη C++
- ✓ Ανάθεση Τιμών
- ✓ Πράξεις στη C++
- ✓ Προτεραιότητα τελεστών
- ✓ Casting

Τελεστές - Casting

4

Τύποι Δεδομένων

5

Τ
Ε
Λ
Σ
Τ
Ε
Σ

C
A
S
T
I
N
G

Name	Description	Size*	Range*
char	Character or small integer.	1byte	signed: -128 to 127 unsigned: 0 to 255
short int (short)	Short Integer.	2bytes	signed: -32768 to 32767 unsigned: 0 to 65535
int	Integer.	4bytes	signed: -2147483648 to 2147483647 unsigned: 0 to 4294967295
long int (long)	Long integer.	4bytes	signed: -2147483648 to 2147483647 unsigned: 0 to 4294967295
bool	Boolean value. It can take one of two values: true or false.	1byte	true or false
float	Floating point number.	4bytes	+/- 3.4e +/- 38 (~7 digits)
double	Double precision floating point number.	8bytes	+/- 1.7e +/- 308 (~15 digits)
long double	Long double precision floating point number.	8bytes	+/- 1.7e +/- 308 (~15 digits)
wchar_t	Wide character.	2 or 4 bytes	1 wide character

Τελεστές - Operators

6

✓ *Αριθμητικοί (arithmetic)*

+, -, /, *, \ ακέραιο πηλίκο, % ακέραιο υπόλοιπο

✓ *Αύξησης (increase)*

++

✓ *Μείωσης (decrease)*

--

✓ *Απόδοσης τιμής (compound assignment)*

=, +=, -=, *=, /=, %=

✓ *Συσχετιστικοί (relational and equality)*

<, >, <=, >=, ==, !=

✓ *Λογικοί (Logical)*

&& (and), || (or), ! (not)

Ανάθεση Τιμής (=)

7

Ανάθεση ή εκχώρηση ή καταχώρηση

- ✓ `year = 2012;`
- ✓ `int year; cin >> year;`
- ✓ `i = i + 1;`

Τελεστές

8

Τ
Ε
Λ
Ε
Σ
Τ
Ε
Σ

C
A
S
T
I
N
G

- ✓ Προσοχή: Διαίρεση ακεραίων δίνει ακέραιο
- ✓ Τελεστής %
- ✓ Λογικοί Τελεστές
- ✓ Τελεστές Μοναδιαίας Αύξησης - Μείωσης

Προτεραιότητες Τελεστών

9

Τ
Ε
Λ
Ε
Σ
Τ
Ε
Σ

C
A
S
T
I
N
G

Precedence	Operator	Description	Associativity	
1	::	Scope resolution	Left-to-right	
2	++ --	Suffix/postfix increment and decrement		
	()	Function call		
	[]	Array subscripting		
	. ->	Element selection by reference Element selection through pointer		
3	++ --	Prefix increment and decrement	Right-to-left	
	+ -	Unary plus and minus		
	! ~	Logical NOT and bitwise NOT		
	(type)	Type cast		
	*	Indirection (dereference)		
	&	Address-of		
	sizeof	Size-of		
	new, new[] delete, delete[]	Dynamic memory allocation Dynamic memory deallocation		
4	.* ->*	Pointer to member	Left-to-right	
5	* / %	Multiplication, division, and remainder		
6	+ -	Addition and subtraction		
7	<< >>	Bitwise left shift and right shift		
8	< <=	For relational operators < and ≤ respectively		
	> >=	For relational operators > and ≥ respectively		
9	== !=	For relational = and ≠ respectively		
10	&	Bitwise AND		
11	^	Bitwise XOR (exclusive or)		
12		Bitwise OR (inclusive or)		
13	&&	Logical AND		
14		Logical OR		
15	?:	Ternary conditional		Right-to-left
	=	Direct assignment (provided by default for C++ classes)		
	+= -=	Assignment by sum and difference		
	*= /= %=	Assignment by product, quotient, and remainder		
	<<= >>=	Assignment by bitwise left shift and right shift		
&= ^= =	Assignment by bitwise AND, XOR, and OR			
16	throw	Throw operator (for exceptions)	Left-to-right	
17	,	Comma		

Προτεραιότητες Τελεστών (συν.)

10

Τ
Ε
Λ
Ε
Σ
Τ
Ε
Σ

C
A
S
T
I
N
G

()	→
++ -- -	←
* / %	→
+ -	→
< <= > >=	→
== !=	→
&&	→
	→
=	←

Casting

11

Τ
Ε
Λ
Ε
Σ
Τ
Ε
Σ

C
A
S
T
I
N
G

```
int m, k=1.5;
float x;
m = 1.6 + 1.7;           // το m γίνεται 3, δηλ. το ακέραιο μέρος του 3.3
m = (int) 1.6 + (int) k; //το m γίνεται 1+1=2
x = (int) k*k;           // 1*1.5 = 1.5, άρα το x γίνεται 1.5
x = (int) (k*k);        // 1.5*1.5=2.25, άρα το x γίνεται 2.0
```

Παραδείγματα

12

```
int i=1, j=2, k=3, p;
```

```
float x=3, y=5;
```

```
i+j/x*y ??
```

```
k/j*++y ??
```

```
y, x++, x ??
```

```
j>i&&!x>=--y ??
```

Παραδείγματα

13

ΑΣΚΗΣΗ 3.2.

Υλοποιήστε πρόγραμμα το οποίο:

A) Να ζητά από το χρήστη δύο πραγματικούς αριθμούς στο διάστημα [10,100), πρώτα το μεγαλύτερο και μετά το μικρότερο. Αφού τους δώσει ο χρήστης (θεωρήστε, για τις εκτυπώσεις, ότι οι 2 αριθμοί έχουν διψήφιο ακέραιο μέρος και μονοψήφιο δεκαδικό μέρος), το πρόγραμμα:

B) Να απαντά πόσο είναι το ακέραιο μέρος και ποιο το δεκαδικό μέρος για καθέναν αναλυτικά.

Γ) Να εκτελεί και να εκτυπώνει τις 4 αριθμητικές πράξεις μεταξύ τους, καθώς και το ακέραιο πηλίκο και το υπόλοιπο της διαίρεσης των ακεραίων μερών τους. Θεωρήστε 2 δεκαδικά ψηφία για τις εκτυπώσεις των αποτελεσμάτων.

Ακολουθεί παράδειγμα επιθυμητής συμπεριφοράς της αλληλεπίδρασης προγράμματος – χρήστη:

```
Please enter the bigger of the two numbers:
40.7
Please enter the smaller of the two numbers:
10.3
The integer part of 40.7 is 40 and its fractional part is 0.7.
The integer part of 10.3 is 10 and its fractional part is 0.3.
40.7 + 10.3 = 51.00
40.7 - 10.3 = 30.40
40.7 x 10.3 = 419.21
40.7 / 10.3 = 3.95
40 / 10 = 4 and the remainder is 0
```