

ΔΕΥΤΕΡΗ ΔΙΑΛΕΞΗ

Επανάληψη

Κύρια συνάρτηση

```
int main() { ... }
```

Τύποι

ακέραιοι **int**, `std::size_t` από το `<cstddef>`

πραγματικός **double**

Δηλώσεις

τύπος όνομα_μεταβλητής;

τύπος όνομα_μεταβλητής{αρχική τιμή};

auto όνομα_μεταβλητής = αρχική τιμή;

Αριθμητικοί Τελεστές

*+, -, *, /, %*

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

++, --

Είσοδος/Εξοδος δεδομένων

Αφού κάνουμε **#include** το `<iostream>`:

- Το

```
std::cin >> a;
```

 διαβάζει τιμή στη μεταβλητή `a`.
- Το

```
std::cout << a;
```

 τυπώνει την τιμή της ποσότητας `a`.
- Το

```
std::cout << u8"Ελληνικό κείμενο";
```

 τυπώνει το κείμενο που γράφουμε εντός **διπλών** εισαγωγικών. Αν το κείμενο έχει ελληνικούς χαρακτήρες, βάζουμε **u8** πριν τα αρχικά εισαγωγικά.

Σταθερή ποσότητα

Μια ποσότητα που επιθυμούμε να πάρει τιμή που **να μην μπορεί να αλλάξει** κατά τη διάρκεια εκτέλεσης του προγράμματος, δηλώνεται με τη χρήση της λέξης **const**.

Αν είναι γνωστή η τιμή κατά τη μεταγλώττιση, μπορούμε να χρησιμοποιήσουμε τη λέξη **constexpr**. Είναι **απαραίτητο** να της δώσουμε **αρχική (και μόνιμη) τιμή** κατά τον ορισμό της.

Παράδειγμα

```
double const x{std::pow(2.3, 1.2)};
double constexpr pi{3.14159265358};
double constexpr twopi{2.0*pi};
```

Κανόνες σχηματισμού ονόματος

- Επιτρεπτοί χαρακτήρες: a-z, A-Z, 0-9, και _.
- Μήκος: οποιοδήποτε.
- Δεν επιτρέπεται να αρχίζει από αριθμητικό ψηφίο.
- Δεν επιτρέπεται να αποτελεί ενσωματωμένη λέξη της C++ (πίνακας 2.1 σημειώσεων).
- Κεφαλαία και πεζά γράμματα **είναι διαφορετικά**.

Ρητή μετατροπή από ένα τύπο σε άλλο

Τι θα τυπωθεί στον παρακάτω κώδικα;

```
int a{9}, b{2};  
std::cout << a/b << '\n';
```

Χωρίς να αλλάξουμε τον τύπο των **a,b**, πώς θα υπολογίσουμε το λόγο τους (και όχι το πηλίκο);

Ρητή (σε αντιδιαστολή με την αυτόματη) μετατροπή τιμής από ένα τύπο σε άλλο γίνεται με το **static_cast<>**:

```
static_cast<double>(a)/static_cast<double>(b);
```

Η μετατροπή αφορά **την τιμή** που έχει η μεταβλητή στο όρισμα. Ο **τύπος** της δεν αλλάζει.

Σύνταξη

```
static_cast<νέος_τύπος>(έκφραση);
```

Εντολή εκχώρησης τιμής

μεταβλητή = [γενική έκφραση] ;

Πρώτα εκτελούνται όλες οι πράξεις, κλήσεις συναρτήσεων κλπ. που εμφανίζονται στο δεξί μέλος.

Κατόπιν, το αποτέλεσμα **μετατρέπεται** (αν χρειάζεται) στον τύπο της **(υποχρεωτικά) μεταβλητής** του αριστερού μέλους και η τιμή που προκύπτει εκχωρείται σε αυτή.

Παραδείγματα

```
int a;
```

```
float b;
```

```
a = 2.5 * 3; // a ← 7
```

```
b = 1.00000001; // b ← 1.0f
```

Τύπος χαρακτήρα (**char**)

Ο τύπος **char** είναι κατάλληλος για την αναπαράσταση ποσοτήτων που οι δυνατές τιμές τους είναι χαρακτήρες.

Τιμές

Απλοί ή ειδικοί χαρακτήρες εντός απλών (') εισαγωγικών.

Δεν επιτρέπονται ελληνικοί χαρακτήρες.

Δήλωση

```
char c;  
char d{'a'};
```

Εκχώρηση

```
c = 'e';  
c = '\n';
```

Ισοδυναμία με ακέραιους

Κάθε χαρακτήρας είναι ακέραιος με τιμή που ορίζεται στο σύνολο χαρακτήρων της υλοποίησης (συνήθως το ASCII).

Ειδικόί Χαρακτήρες της C++

Ειδικός Χαρακτήρας	Περιγραφή
\'	Απόστροφος
\"	Εισαγωγικά
\?	Ερωτηματικό
\\	Ανάποδη κάθετος
\a	Κουδούνι
\b	Διαγραφή προηγούμενου χαρακτήρα
\f	Αλλαγή σελίδας
\n	Αλλαγή γραμμής
\r	Μετακίνηση στην αρχή της γραμμής
\t	Οριζόντιο tab
\v	Κατακόρυφο tab
\ooo	Χαρακτήρας με οκταδική αναπαράσταση ooo
\xhhh	Χαρακτήρας με δεκαεξαδική αναπαράσταση hhh
\unnnn	Ο χαρακτήρας unicode U+nnnn
\Unnnnnnnn	Ο χαρακτήρας unicode U+nnnnnnnn

Εμβέλεια

Μια ποσότητα (μεταβλητή, σταθερή, συνάρτηση, κλπ.) μπορεί

- να χρησιμοποιηθεί από το σημείο της δήλωσής της έως το } που κλείνει την περιοχή στην οποία δηλώθηκε.
- να οριστεί ξανά (ίσως με άλλο τύπο) σε «εσωτερικό» τμήμα.

Παράδειγμα

```
{  
  int a;  
  {  
    a = ... // Σωστό  
    int b = ... // Σωστό  
    double a = ... // Σωστό, νέο a  
  }  
  a = ... // Σωστό, είναι το αρχικό a  
  b = ... // Λάθος  
}  
a = ... // Λάθος  
b = ... // Λάθος
```

Μαθηματικές συναρτήσεις

Συνάρτηση	Επιστρεφόμενη τιμή
<code>double cos(double x)</code>	Συνημίτονο του x .
<code>double sin(double x)</code>	Ημίτονο του x .
<code>double tan(double x)</code>	Εφαπτομένη του x .
<code>double acos(double x)</code>	Τόξο συνημιτόνου του x .
<code>double asin(double x)</code>	Τόξο ημιτόνου του x .
<code>double atan(double x)</code>	Τόξο εφαπτομένης του x .
<code>double atan2(double x, double y)</code>	Τόξο εφαπτομένης $\tan^{-1}(x/y)$.
<code>double pow(double x, double a)</code>	Ύψωση σε δύναμη, x^a .
<code>double sqrt(double x)</code>	Η τετραγωνική ρίζα του x .
<code>double cbrt(double x)</code>	Η κυβική ρίζα του x .
<code>double hypot(double x, double y)</code>	$\sqrt{x^2 + y^2}$.
<code>double exp(double x)</code>	Εκθετικό του x (e^x).
<code>double log(double x)</code>	Φυσικός λογάριθμος του x ($\ln x$).
<code>double log2(double x)</code>	Δυαδικός λογάριθμος του x ($\log_2 x$).
<code>double log10(double x)</code>	Δεκαδικός λογάριθμος του x ($\log_{10} x$).
<code>double abs(double x)</code>	Απόλυτη τιμή του x .

Όλες χρειάζονται πρόθεμα `std::`. Προϋποθέτουν το `#include <cmath>`.
Οι τριγωνομετρικές συναρτήσεις δέχονται/επιστρέφουν τις γωνίες σε rad.

Παράδειγμα

Το $e^{-x} \cos(y)$ γράφεται `std::exp(-x) * std::cos(y)`.

Ασκήσεις

Δεύτερο κεφάλαιο

4,5,6,7,8,9,10,11,12,13,14,15,18,19