

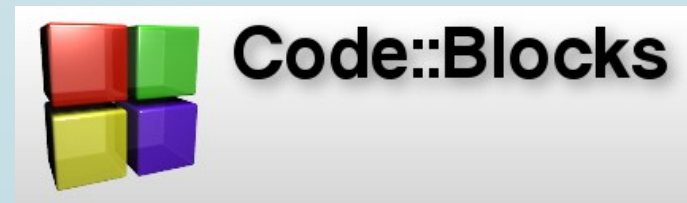
Εισαγωγή στη γλώσσα προγραμματισμού C++

Περιβάλλον Εργασίας

Άριστος Πασιάς

Περιβάλλον Εργασίας

1. Χρήση απλού κειμενογράφου και Μεταγλωττιστή
2. Ολοκληρωμένα Περιβάλλοντα Εργασίας (Integrated Development Environments)
 - ▶ Επιτρέπουν το γράψιμο προγράμματος, τον αυτόματο έλεγχο λαθών, την μεταγλώττιση, παραγωγή εκτελέσιμου αρχείου και εκτέλεση του προγράμματος.
 - ▶ DevC++, MinGW studio, **Code Blocks**, Visual Studio, Eclipse, Netbeans και άλλα



Εγκατάσταση του Code::Blocks

3



Code::Blocks

Code::Blocks - The IDE with all the features you need, having a consistent I

Home Features Downloads Forums Wiki

Main

Please select a setup package depending on your platform:

- Windows XP / Vista / 7 / 8.x / 10
- Linux 32 and 64-bit
- Mac OS X

• Home
• Features
• Screenshots



Windows XP / Vista / 7 / 8.x / 10:

File

codeblocks-17.12-setup.exe

codeblocks-17.12-setup-nonadmin.exe

codeblocks-17.12-nosetup.zip

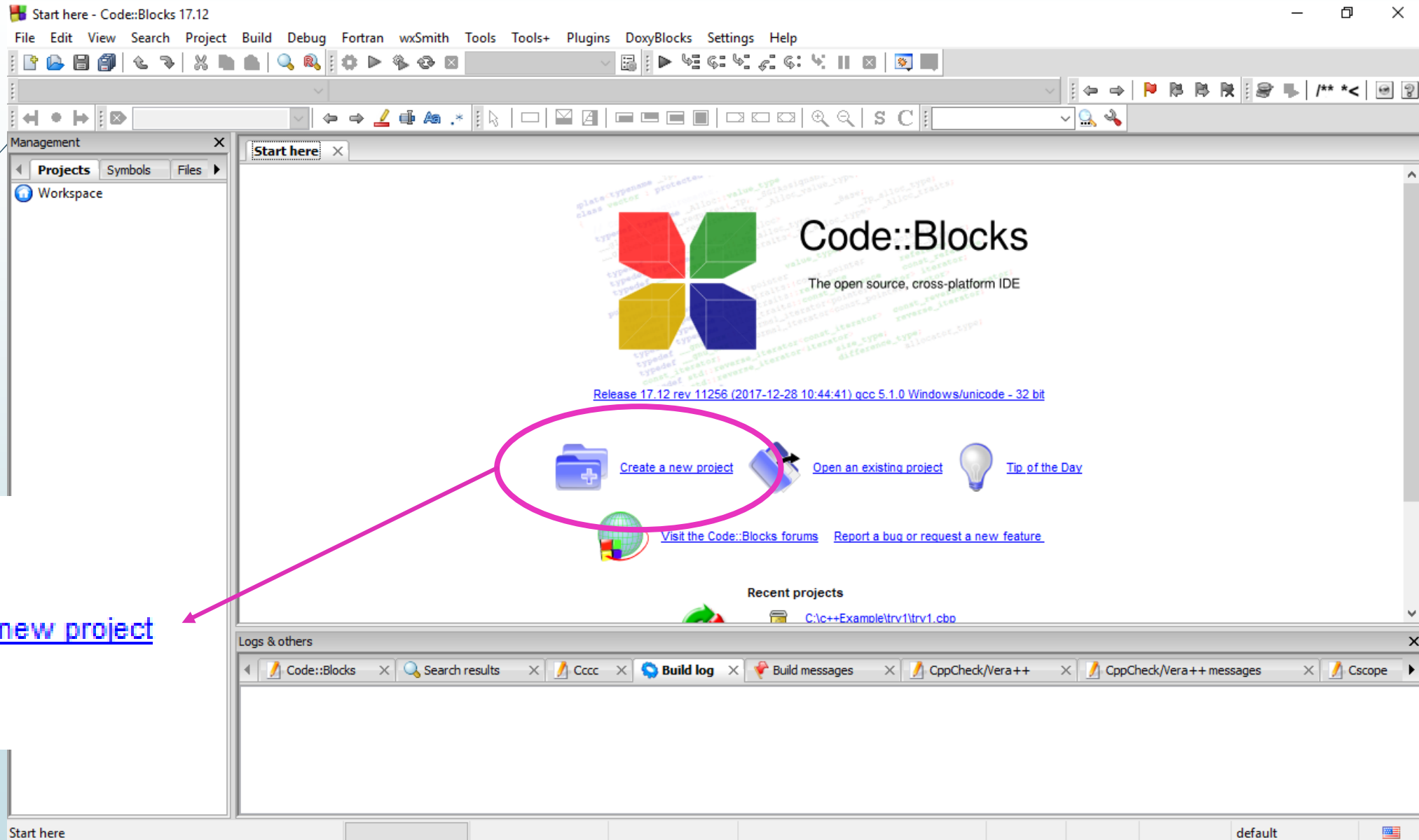
codeblocks-17.12mingw-setup.exe

codeblocks-17.12mingw-nosetup.zip

codeblocks-17.12mingw_fortran-setup.exe

Πρώτη επαφή

4



[Create a new project](#)

Πρώτη επαφή

5

New from template

Projects
Build targets
Files
Custom
User templates

Category: <All categories>

ARM Project AVR Project Arduino Project Code::Blocks plugin
Console application D application Direct/X project Dynamic Link Library
Empty project FLTK project Fortran DLL Fortran application
Fortran library GLFW project GLUT project GTK+ project

Go
Cancel

View as
 Large icons
 List

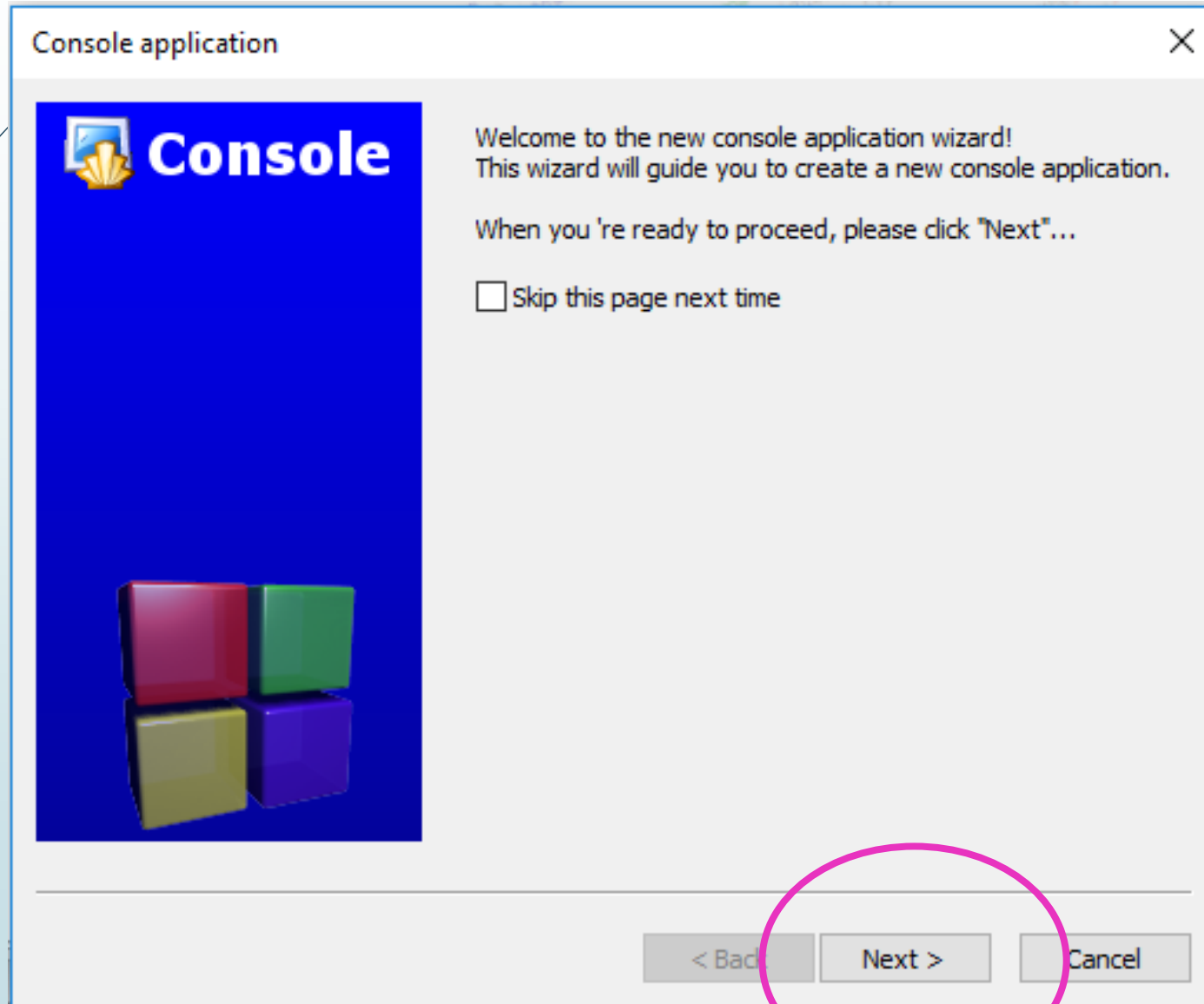
TIP: Try right-clicking an item

1. Select a wizard type first on the left
2. Select a specific wizard from the main window (filter by categories if needed)
3. Press Go

Console application

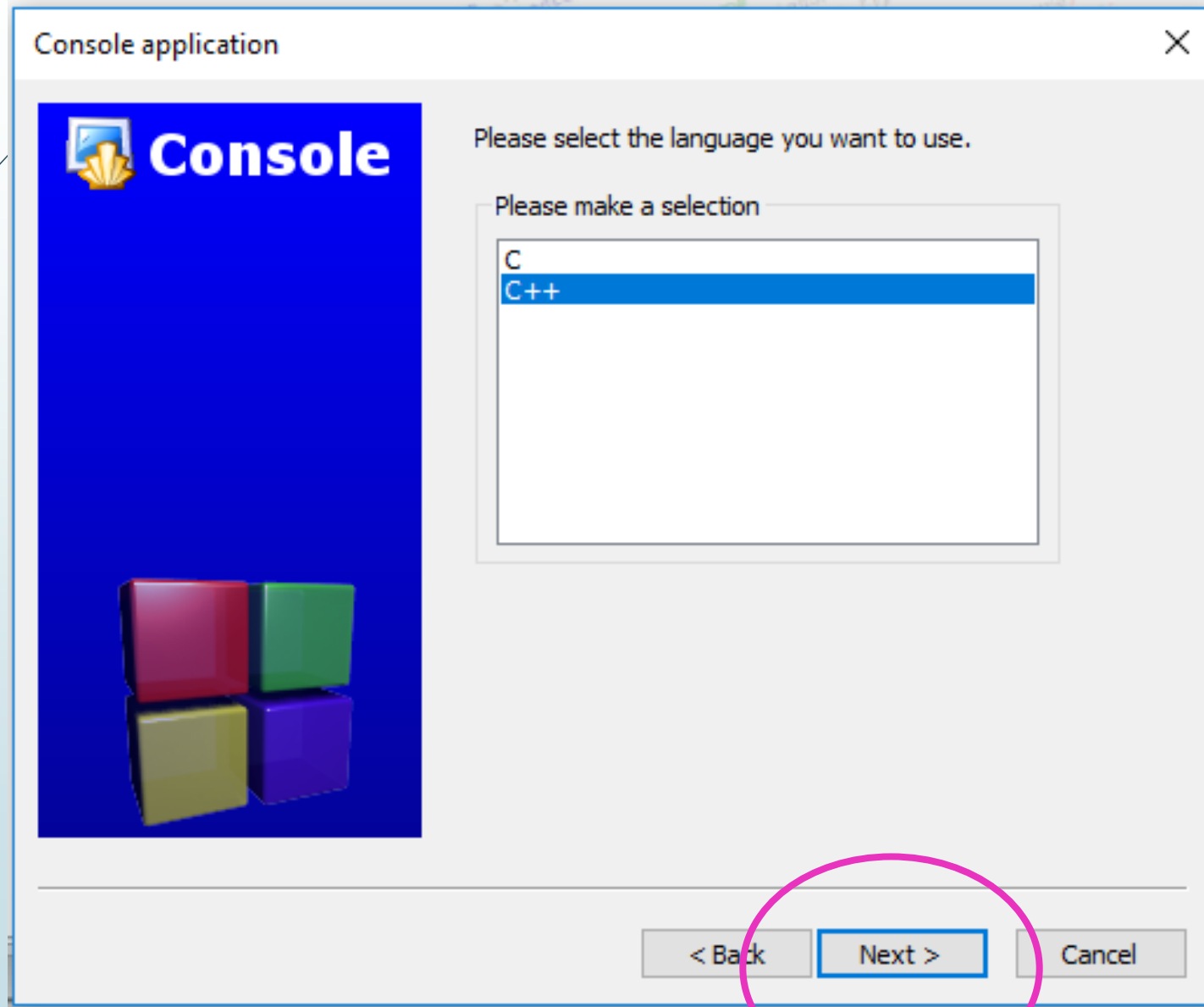
Πρώτη επαφή

6



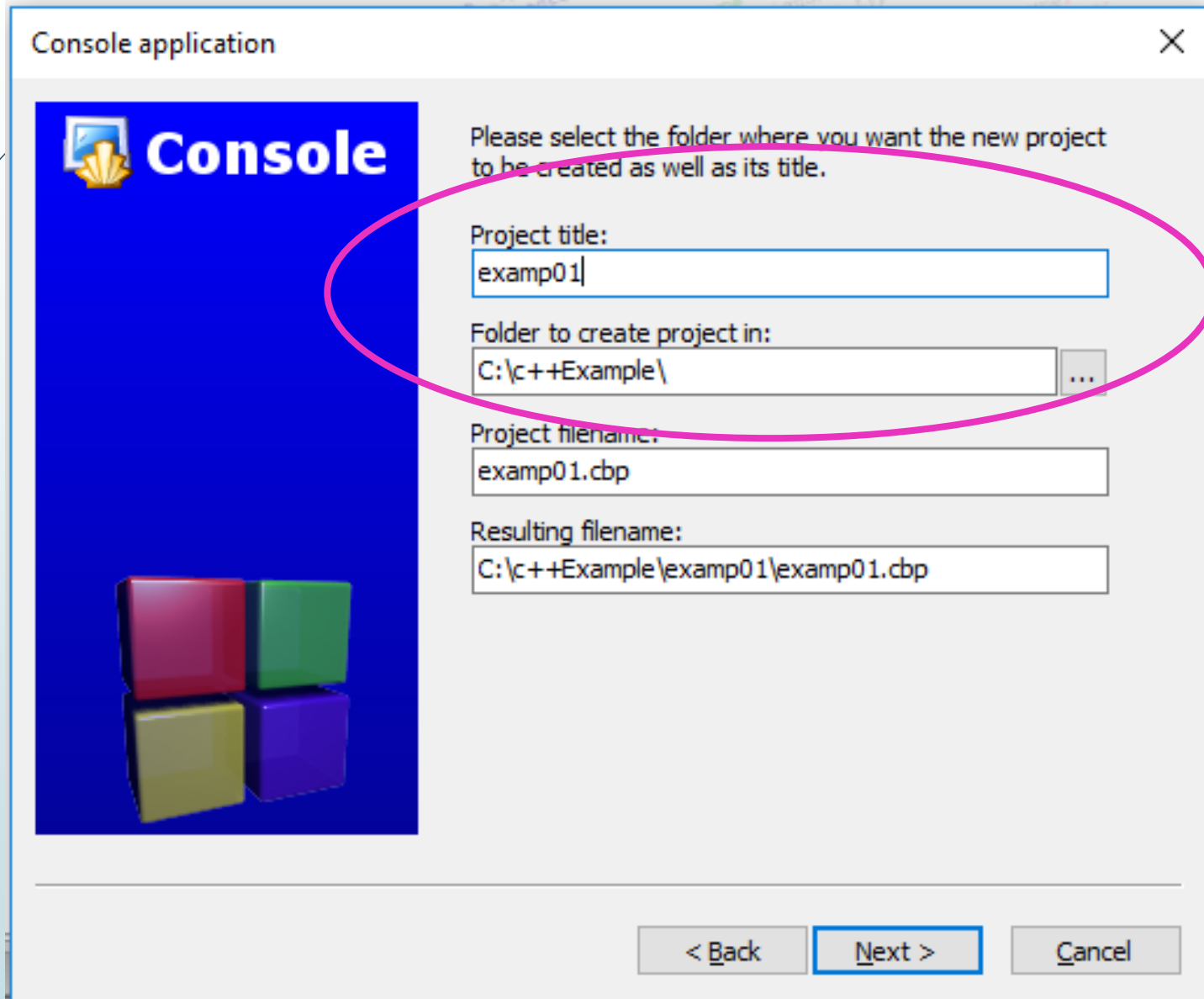
Πρώτη επαφή

7



Πρώτη επαφή

8



Console application

Please select the folder where you want the new project to be created as well as its title.

Project title:
examp01

Folder to create project in:
C:\c++Example\

Project filename:
examp01.cbp


Resulting filename:
C:\c++Example\examp01\examp01.cbp

< Back Next > Cancel

Πρώτη επαφή

9

Console application

 **Console**

Please select the compiler to use and which configurations you want enabled in your project.

Compiler: GNU GCC Compiler

Create "Debug" configuration: Debug

"Debug" options

Output dir.: bin\Debug\
Objects output dir.: obj\Debug\

 Create "Release" configuration: Release

"Release" options

Output dir.: bin\Release\
Objects output dir.: obj\Release\

< Back Finish Cancel

Πρώτη επαφή

10

The screenshot shows the Code::Blocks IDE interface. The main window is titled "[examp01] - Code::Blocks 17.12". The menu bar includes File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, and Help. The toolbar contains various icons for file operations, editing, and debugging. The left sidebar is divided into three tabs: Projects, Symbols, and Files. The Projects tab is active, showing a workspace named "examp01" with a sub-folder "Sources". The main editor area is currently blank. The bottom status bar shows "Welcome to Code::Blocks!" and "default".

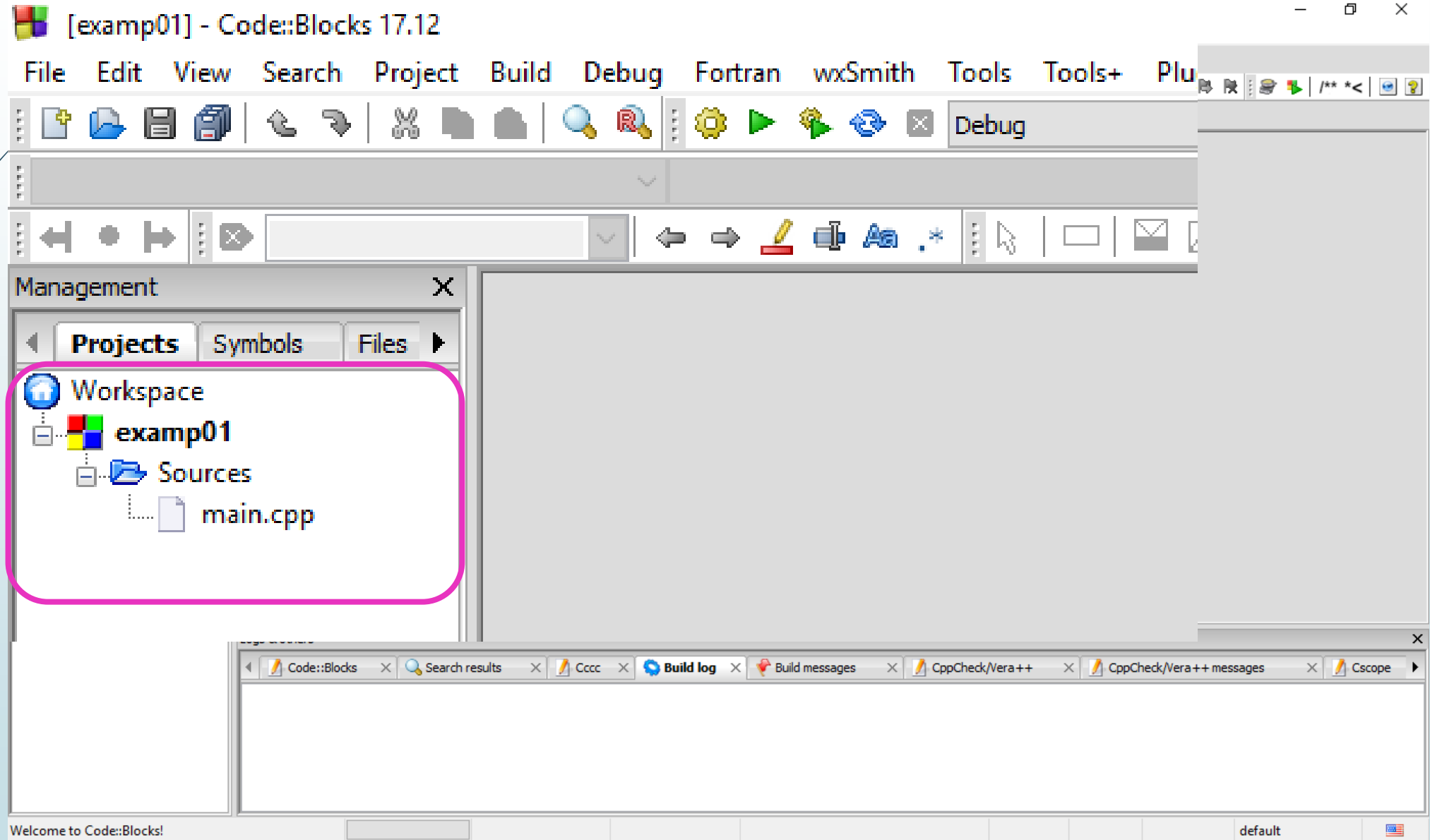
Management (Διαχείριση)
Καρτέλα **Projects** (Έργα) δείχνει όλα τα έργα που είναι ανοικτά σε CodeBlocks.
Καρτέλα **Symbols** (Σύμβολα) δείχνει σύμβολα, μεταβλητές κλπ.
Καρτέλα **Files**

Editor (Συντάκτης) Περιοχή σύνταξης του κώδικα

Logs & Others (Αρχεία καταγραφής και άλλα)
Αποτελέσματα αναζήτησης, μηνύματα καταγραφής ενός μεταγλωττιστή κλπ.

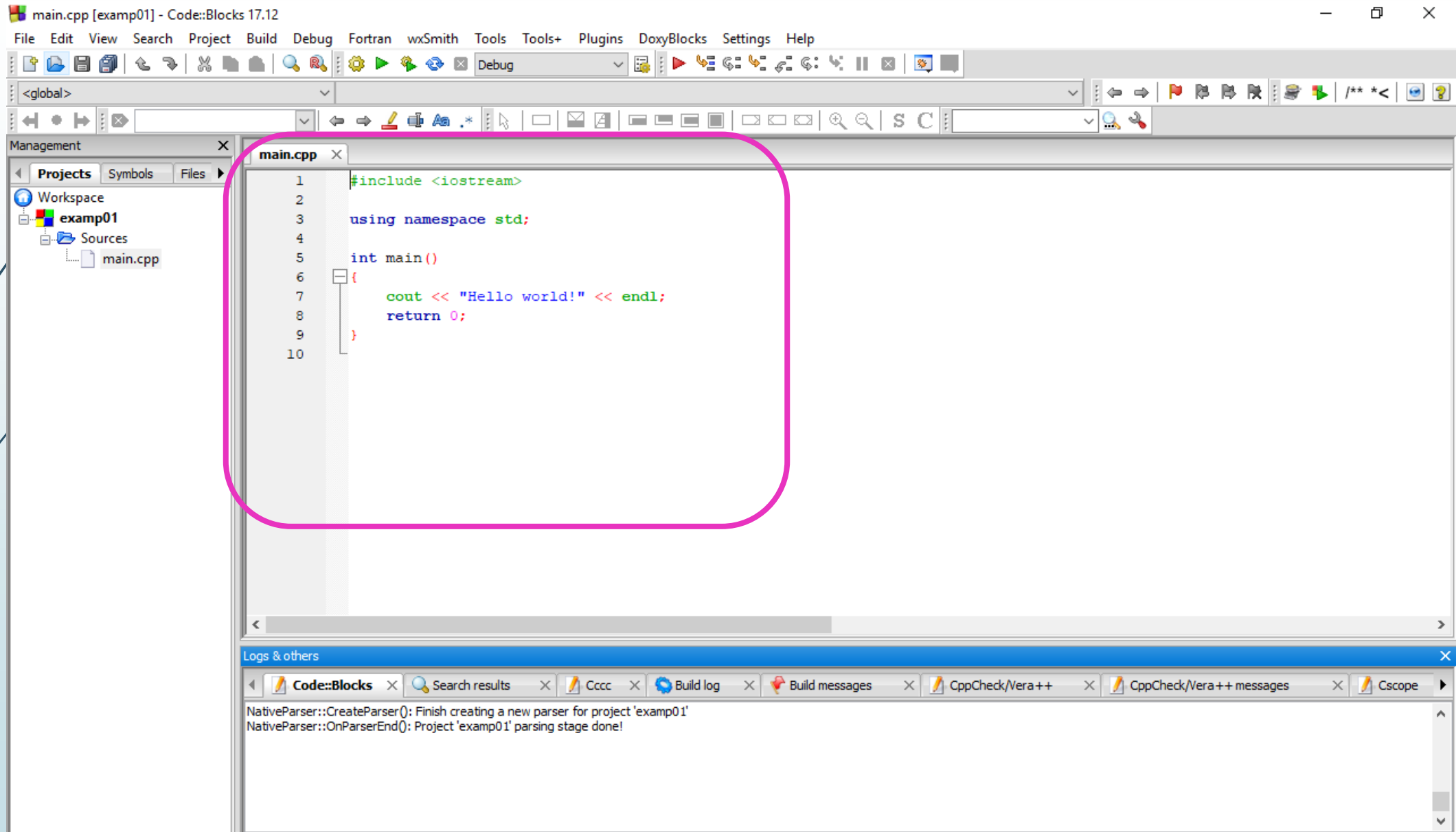
Πρώτη επαφή

11



Πρώτη επαφή

12



The screenshot shows the Code::Blocks IDE interface. The main window displays the source code for `main.cpp`, which is highlighted with a pink rounded rectangle. The code is as follows:

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     cout << "Hello world!" << endl;
8     return 0;
9 }
10
```

The left sidebar shows the project structure for `examp01`, including a `Sources` folder containing `main.cpp`. The bottom status bar shows the `Logs & others` panel with the following messages:

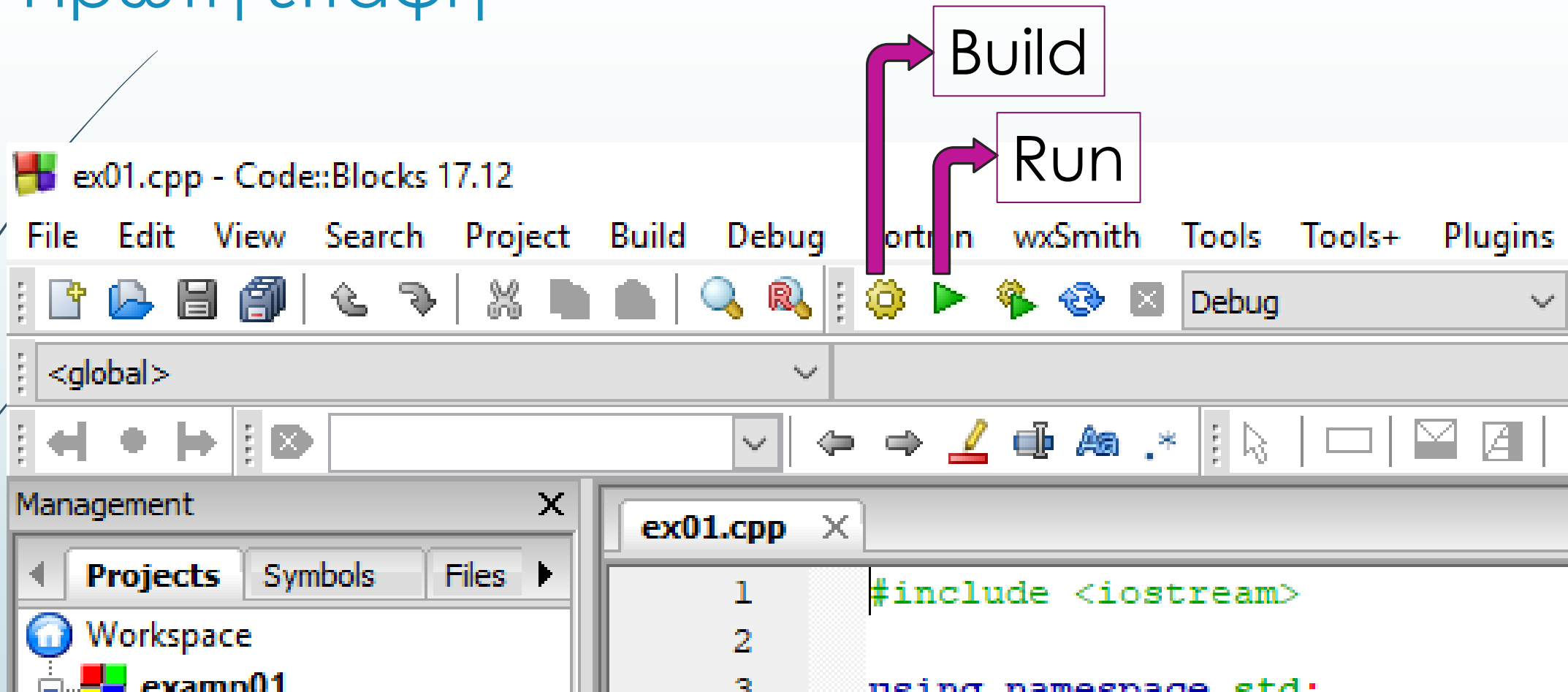
```
Code::Blocks x Search results x Cccc x Build log x Build messages x CppCheck/Vera++ x CppCheck/Vera++ messages x Cscope
NativeParser::CreateParser(): Finish creating a new parser for project 'examp01'
NativeParser::OnParserEnd(): Project 'examp01' parsing stage done!
```

Πρώτη επαφή

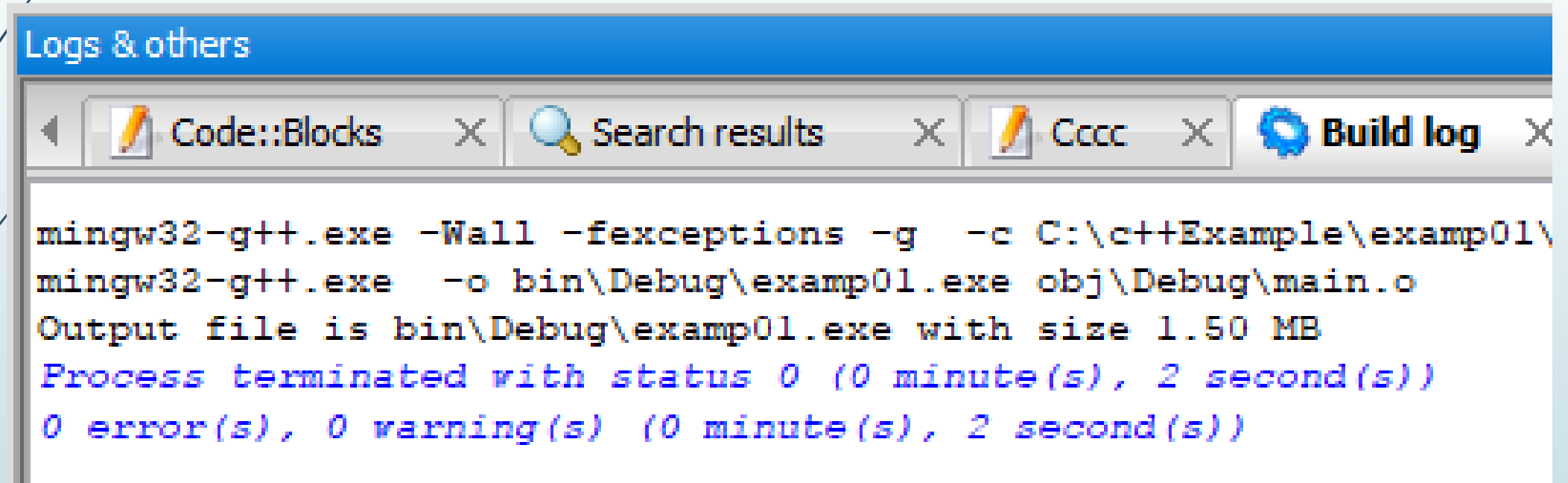
13

```
main.cpp X
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      cout << "Hello world!" << endl;
8      return 0;
9  }
10
```

Πρώτη επαφή



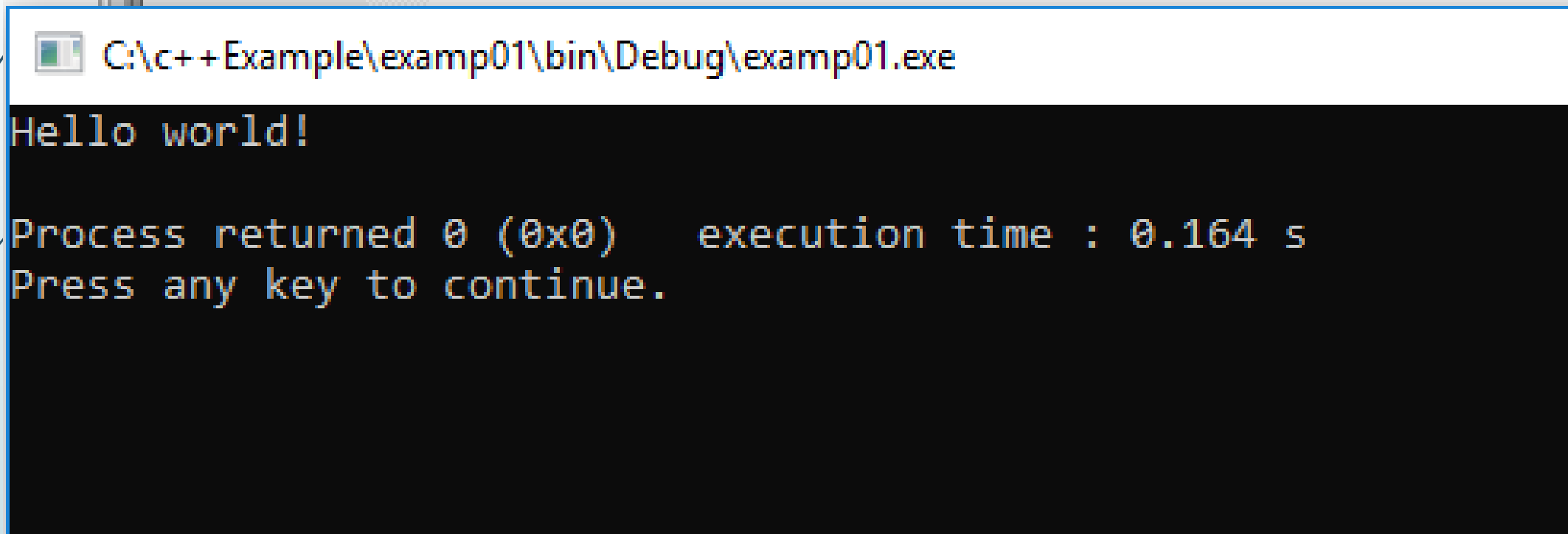
Πρώτη επαφή



The screenshot shows the 'Logs & others' window in Visual Studio. The window title bar includes 'Code::Blocks', 'Search results', 'Cccc', and 'Build log'. The main content area displays the following build output:

```
mingw32-g++.exe -Wall -fexceptions -g -c C:\c++Example\examp01\  
mingw32-g++.exe -o bin\Debug\examp01.exe obj\Debug\main.o  
Output file is bin\Debug\examp01.exe with size 1.50 MB  
Process terminated with status 0 (0 minute(s), 2 second(s))  
0 error(s), 0 warning(s) (0 minute(s), 2 second(s))
```

Πρώτη επαφή



A screenshot of a Windows command prompt window. The title bar shows the file path: C:\c++Example\examp01\bin\Debug\examp01.exe. The command prompt displays the output of a program: "Hello world!". Below this, it shows "Process returned 0 (0x0) execution time : 0.164 s" and "Press any key to continue.".

```
C:\c++Example\examp01\bin\Debug\examp01.exe  
Hello world!  
Process returned 0 (0x0) execution time : 0.164 s  
Press any key to continue.
```

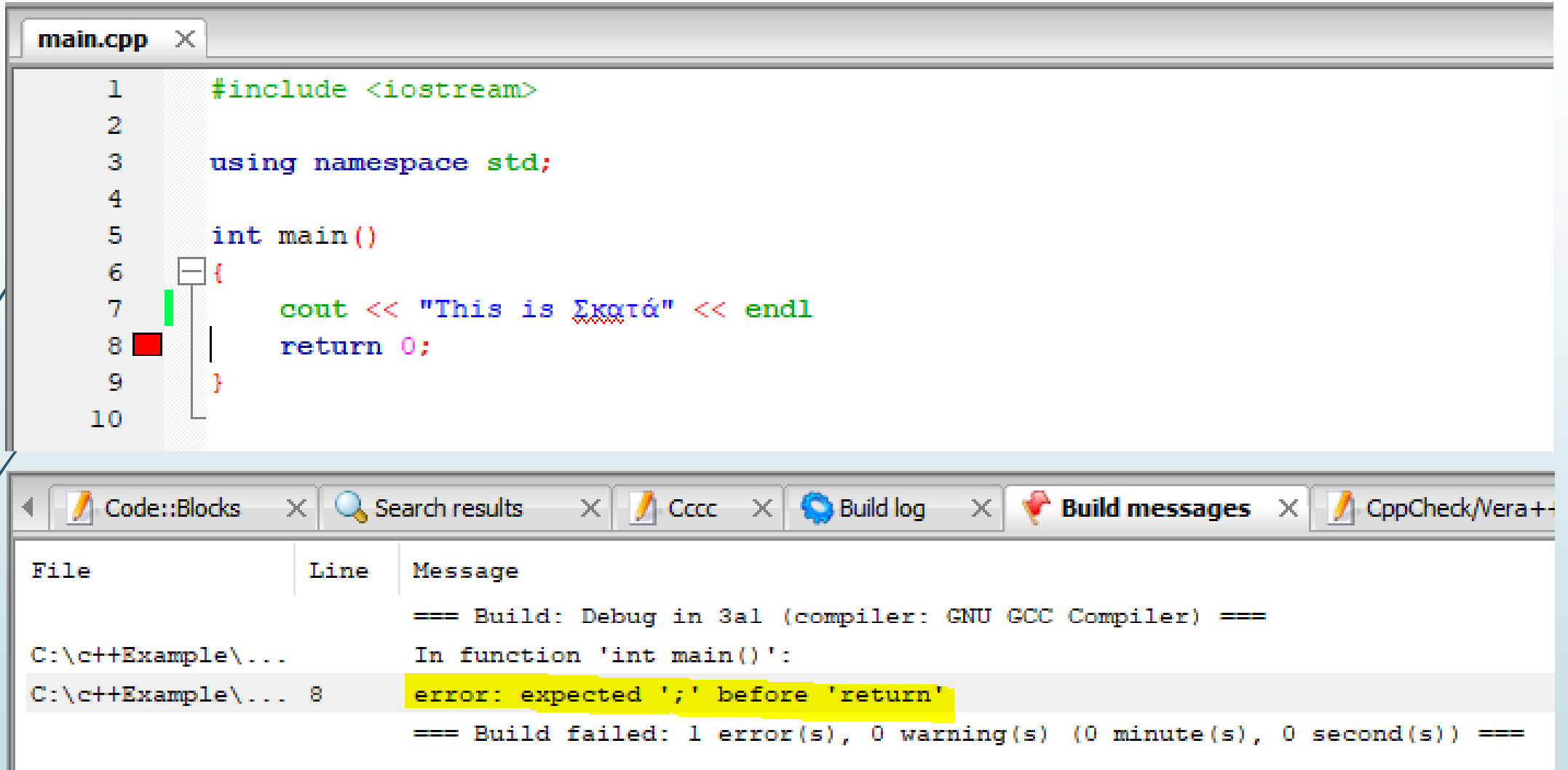
Το Πρώτο πρόγραμμα

17

```
// Αυτή η γραμμή είναι σχόλιο.  
/* Έναρξη Σχολιου πολλών γραμμών  
Η γραμμή "#include" Εισάγει μια βιβλιοθήκη με όνομα iostream  
using namespace std;  
Είναι ο χώρος τυπικών ονομάτων εισόδων / εξόδων cout, cin  
Κλείσιμο σχολίων */  
  
#include <iostream>  
  
using namespace std;  
  
/* Η int main() Ορίζει τη βασική συνάρτηση. Πρέπει να υπάρχει σε κάθε πρόγραμμα C++  
επιστρέφει ένα ακέραιο αριθμό. (int). Δεν δέχεται ορίσματα.  
Ανάμεσα στις αγκύλες { } περιλαμβάνονται οι εντολές που εκτελούνται  
με την κλήση της. */  
int main()  
{  
    cout << "Lets see the main program!" << endl; // console out , end of line  
    return 0;  
}
```

Το Πρώτο πρόγραμμα – μηνύματα μεταγλωττιστή

18



```
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      cout << "This is Σκατά" << endl;
8      return 0;
9  }
```

Code::Blocks Search results Cccc Build log Build messages CppCheck/Vera++

File	Line	Message
		=== Build: Debug in 3a1 (compiler: GNU GCC Compiler) ===
C:\c++Example\...		In function 'int main()':
C:\c++Example\...	8	error: expected ';' before 'return'
		=== Build failed: 1 error(s), 0 warning(s) (0 minute(s), 0 second(s)) ===

Το Πρώτο πρόγραμμα – Ελληνικοί χαρακτήρες

19

```
1  #include <iostream>
2
3  using namespace std;
4  #include <stdio.h>
5  #include <stdlib.h>
6
7  int main()
8  {
9      system("chcp 1253");
10     cout << "Τώρα μπορώ να εμφανίσω Ελληνικούς χαρακτήρες" << endl;
11     return 0;
12 }
13
```

C:\c++\Example\3a1\bin\Debug\3a1.exe

Active code page: 1253

Τώρα μπορώ να εμφανίσω Ελληνικούς χαρακτήρες

Process returned 0 (0x0) execution time : 0.170 s

Press any key to continue.

Εντολή εξόδου cout

20

```
1 // More for the COUT
2 #include <iostream>
3 using namespace std;
4
5 #include <stdio.h>
6 #include<stdlib.h>
7
8 int main()
9 {
10     system("chcp 1253");
11
12     cout << "Β' Τεχνική σχολή Λεμεσού";
13     cout << "Κλάδος Ηλεκτρολογίας";
14     cout << "Ψηφιακή Τεχνολογία και προγραμματισμός";
15     cout << "ΘΗΨ1";
16     return 0;
17 }
18
```

C:\c++Example\3a1\bin\Debug\3a1.exe

Active code page: 1253

Β' Τεχνική σχολή ΛεμεσούΚλάδος ΗλεκτρολογίαςΨηφιακή Τεχνολογία και προγραμματισμόςΘΗΨ1

Process returned 0 (0x0) execution time : 0.389 s

Press any key to continue.

Εντολή εξόδου cout

21

```
1 // More for the COUT
2 #include <iostream>
3 using namespace std;
4
5 #include <stdio.h>
6 #include<stdlib.h>
7
8 int main()
9 {
10     system("chcp 1253");
11
12     cout << "B' Τεχνική σχολή Λεμεσού ";
13     cout << "Κλάδος Ηλεκτρολογίας ";
14     cout << "Ψηφιακή Τεχνολογία και προγραμματισμός ";
15     cout << "ΘΗΨ1" << endl;
16
17     cout << "B' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1" << endl;
18
19     cout << "B' Τεχνική σχολή Λεμεσού " << "Κλάδος Ηλεκτρολογίας "
20         << "Ψηφιακή Τεχνολογία και προγραμματισμός " << "ΘΗΨ1" << endl;
21     return 0;
22 }
```

```
C:\c++\Example\3a1\bin\Debug\3a1.exe
Active code page: 1253
B' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1
B' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1
B' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1

Process returned 0 (0x0)   execution time : 0.349 s
Press any key to continue.
```

Εντολή εξόδου cout //ex01a

22

```
// More for the COUT
#include <iostream>
using namespace std;

#include <stdio.h>
#include<stdlib.h>

int main()
{
    system("chcp 1253");

    cout << "Β' Τεχνική σχολή Λεμεσού ";
    cout << "Κλάδος Ηλεκτρολογίας ";
    cout << "Ψηφιακή Τεχνολογία και προγραμματισμός ";
    cout << "ΘΗΨ1" << endl;

    cout << "Β' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1" << endl;

    cout << "Β' Τεχνική σχολή Λεμεσού " << endl << "Κλάδος Ηλεκτρολογίας " << endl
        << "Ψηφιακή Τεχνολογία και προγραμματισμός " << endl << "ΘΗΨ1" << endl;
    return 0;
}
```

```
C:\c++\Example\3a1\bin\Debug\3a1.exe
Active code page: 1253
Β' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1
Β' Τεχνική σχολή Λεμεσού Κλάδος Ηλεκτρολογίας Ψηφιακή Τεχνολογία και προγραμματισμός ΘΗΨ1
Β' Τεχνική σχολή Λεμεσού
Κλάδος Ηλεκτρολογίας
Ψηφιακή Τεχνολογία και προγραμματισμός
ΘΗΨ1

Process returned 0 (0x0)   execution time : 0.386 s
Press any key to continue.
```

Μεταβλητές & Σταθερές

23

- Συχνά κατά την διάρκεια εκτέλεσης ενός προγράμματος χρειάζεται να αποθηκεύσουμε δεδομένα και ενδιάμεσα αποτελέσματα. Όλα αυτά αποθηκεύονται στη μνήμη ως μεταβλητές ή ως σταθερές. Μια σταθερά έχει σταθερή τιμή κατά την διαδικασία εκτέλεσης του προγράμματος, ενώ μια μεταβλητή μπορεί να αλλάξει τιμή.
- Κάθε σταθερά ή μεταβλητή πριν χρησιμοποιηθεί πρέπει να δηλωθεί κατάλληλα.
- Κάθε μεταβλητή ή σταθερά χαρακτηρίζεται από ένα όνομα και μια τιμή.
- Τα ονόματα των μεταβλητών και των σταθερών πρέπει να υπακούουν σε κάποιους κανόνες.
 - Μπορούμε να χρησιμοποιήσουμε γράμματα, αριθμούς και τον χαρακτήρα `_`
 - Ο πρώτος χαρακτήρας πρέπει να είναι λατινικό γράμμα
 - Είναι case sensitive
 - Δεν μπορούμε να χρησιμοποιήσουμε τις δεσμευμένες λέξεις ως ονόματα. (`main`, `cout`, `cin` ...)

Τύποι Μεταβλητών & Σταθερών

24

- Οι μεταβλητές και σταθερές διακρίνονται και από τον τύπο δεδομένων που περιέχουν.

► Τύπος	Διάκριση	Bytes	Εμβέλεια
► Char	Χαρακτήρας	1	-128 ως 127
► Short	ακέραιος αρ.	2	-32.768 ως 32.768
► Int	ακέραιος αρ.	4	-2.147.483.648 ως 2.147.483.647
► long long	ακέραιος αρ.	8	$-2^{63}+1$ ως $2^{63}+1$
► float	πραγματικός αρ.	4	$1,2 \cdot 10^{-38}$ ως $3,4 \cdot 10^{38}$
► double	πραγματικός αρ.	8	$2,2 \cdot 10^{-308}$ ως $1,8 \cdot 10^{308}$
► long double	πραγματικός αρ.	16	$3,4 \cdot 10^{-4932}$ ως $1,2 \cdot 10^{4932}$
► bool	λογικές τιμές	1	true / false

Δήλωση μεταβλητής

- `Int num_1;` // Δήλωση μεταβλητής ακέραιου τύπου με όνομα num_1
- `double Div;` //Δήλωση μεταβλητής δεκαδικού τύπου με όνομα Div
- `Bool star = false;` //Δήλωση λογικής μεταβλητής με ταυτόχρονη ανάθεση αρχικής τιμής
- `Char alfa='A', bita='B', gama='G' ;` //Δήλωση τριών μεταβλητών τύπου χαρακτήρα στην ίδια εντολή (Διαχωρίζονται με ,) με ταυτόχρονη εκχώρηση τιμών.
- `Int find=100, lose=2;`
- `Int aris=10+4;`
- `float number_1=25.12 , number_2=128,24;`
- `Float last_n = number_1+number_2;`

Δήλωση σταθεράς

- `const float persent=0.2;`
- `#define pososto =0.15;`

Εντολή εισόδου cin //ex02a

26

```
1 // More for the COUT
2 #include <iostream>
3
4 using namespace std;
5
6 int main()
7 {
8     int n1,n2,n3,sumi;
9     double d1,d2,d3,sumd;
10    bool test;
11    cin>>n1>>n2>>n3;
12    cin>>d1>>d2>>d3;
13    cin>>test;
14
15    sumi=n1+n2+n3;
16    sumd=d1+d2+d3;
17
18
19    cout<<"sum of n = "<<sumi<<endl;
20    cout<<"sum of d = "<<sumd<<endl;
21    cout<<"the logical = "<<test<<endl;
22    return 0;
23 }
```

C:\c++Example\3a1\bin\Debug\3a1.exe

```
1 3 7
2.3 2.3 2.3
true
sum of n = 11
sum of d = 6.9
the logical = 0
```

```
Process returned 0 (0x0)   execution time : 26.879 s
Press any key to continue.
```

Τελεστές

27

► Τελεστής ανάθεσης (εκχώρησης) τιμής =

► `int num_1;`

► `num_1=12`

► Αριθμητικοί τελεστές :

► + Πρόσθεση

► - Αφαίρεση

► * Πολλαπλασιασμός

► / Διαίρεση

► % Υπόλοιπο ακεραίας διαίρεσης

► ++ Αυξητικός τελεστής

► -- Αφαιρετικός τελεστής

► Τελεστής προσαρμογής τύπου

► `int n;`

► `float pi = 3.14;`

► `n=(int) pi // ή n=int(pi)`

Προτεραιότητα αριθμητικών τελεστών

()

*, /, %

+, -

```
int x = 10;
```

```
x++; // τώρα η τιμή x έχει την τιμή 11
```

```
x--; // και πάλι την τιμή 10
```

```
x += 20; (είναι το ίδιο με την εντολή x = x + 20, Δλδ 30)
```

```
x /= 10; (το ίδιο με x = x / 10, αποτέλεσμα 3).
```

```
int x = 10; // το x έχει την τιμή 10
```

```
int y = x++; // y έχει την τιμή 10, αλλά το x την τιμή 11
```

```
x = 10; // το x έχει πάλι την τιμή 10
```

```
int z = ++x; // το z έχει την τιμή 11, όπως και το x
```

Τελεστές

28

➤ Σχεσιακοί τελεστές :

- == Ισότητα
- != Ανισότητα
- > Μεγαλύτερο από
- < Μικρότερο από
- >= Μεγαλύτερο από ή ίσο με
- <= Μικρότερο από ή ίσο με

➤ Λογικοί τελεστές :

- && AND
- || OR
- ! NOT

```
(alpha == true) && (beta == 1)
```

Παράδειγμα //ex03a

29

```
1  #include <iostream>
2  #include<iomanip>
3  using namespace std;
4
5  int main()
6  {
7      int a1,a2,a3;
8      float mo;
9      cout << "Input three Integers: " << endl;
10     cin >>a1>>a2>>a3;
11     mo= (a1+a2+a3)/3;
12     cout <<"0 Mesos Oros = "<<fixed<<setprecision(2)<<mo;
13     return 0;
14 }
15
```

C:\c++Example\1ex05\bin\Debug\1ex05.exe

Input three Integers:

5 6 8

0 Mesos Oros = 6.00

Process returned 0 (0x0) execution time : 7.269 s

Press any key to continue.

Παράδειγμα //ex03b

30

```
1  #include <iostream>
2  #include<iomanip>
3  using namespace std;
4
5  int main()
6  {
7      int a1,a2,a3;
8      float mo;
9      cout << "Input three Integers: " << endl;
10     cin >>a1>>a2>>a3;
11     mo= float (a1+a2+a3)/3;
12     cout <<"0 Mesos Oros = "<<fixed<<setprecision(2)<<mo;
13     return 0;
14 }
15
```

C:\c++Example\1ex05\bin\Debug\1ex05.exe

```
Input three Integers:
5 6 8
0 Mesos Oros = 6.33
Process returned 0 (0x0)   execution time : 5.119 s
Press any key to continue.
```

Συναρτήσεις

31

- Οι συναρτήσεις είναι μικρά τμήματα κώδικα που εκτελούν προκαθορισμένη λειτουργία κάθε φορά που τα καλούμε στη διαδικασία εκτέλεσης ενός προγράμματος.
 - Η C++ διαθέτει πλήθος έτοιμων συναρτήσεων. Μπορείτε να δείτε αυτές τις συναρτήσεις στη σελίδα: <http://www.cplusplus.com>
 - Για να χρησιμοποιήσουμε μια έτοιμη συνάρτηση πρέπει να δηλώσουμε την βιβλιοθήκη στην οποία ανήκει.
- `sqrt(x)` `sqrt(9)=3`
- `abs(x)` `abs(-2) =2`
- Μπορούμε όμως να ορίσουμε και δικές μας συναρτήσεις τις οποίες μπορούμε να χρησιμοποιούμε όπως τις έτοιμες.

■ Π.χ.

```
Int athrisma (int a , int b);
```

```
{
```

```
  Int c=a + b;
```

```
  return c;
```

```
} ...
```

```
Int y
```

```
y=athrisma(3,6);
```

Συναρτήσεις – Παράδειγμα ex04

32

```
1  #include <iostream>
2  using namespace std;
3  int athrisma (int a, int b)
4  {
5      int c = a+b;
6      return (c);
7  }
8  int main ()
9  {
10     int y;
11     y = athrisma (6, 5);
12     cout << "The result is " << y;
13     return (0);
14 }
15
```

Ορισμός συνάρτησης που δέχεται ως παράμετρος δυο ακέραιους και επιστρέφει το άθροισμα τους σε μια νέα παράμετρο

Η εκτέλεση του προγράμματος ξεκινά πάντα από την Main η οποία καλεί όλες τις άλλες συναρτήσεις και τελειώνει με την επιστροφή από την main της τιμής 0

**Teacher at the beginning
of the school year**



**Teacher at the end
of the school year**

