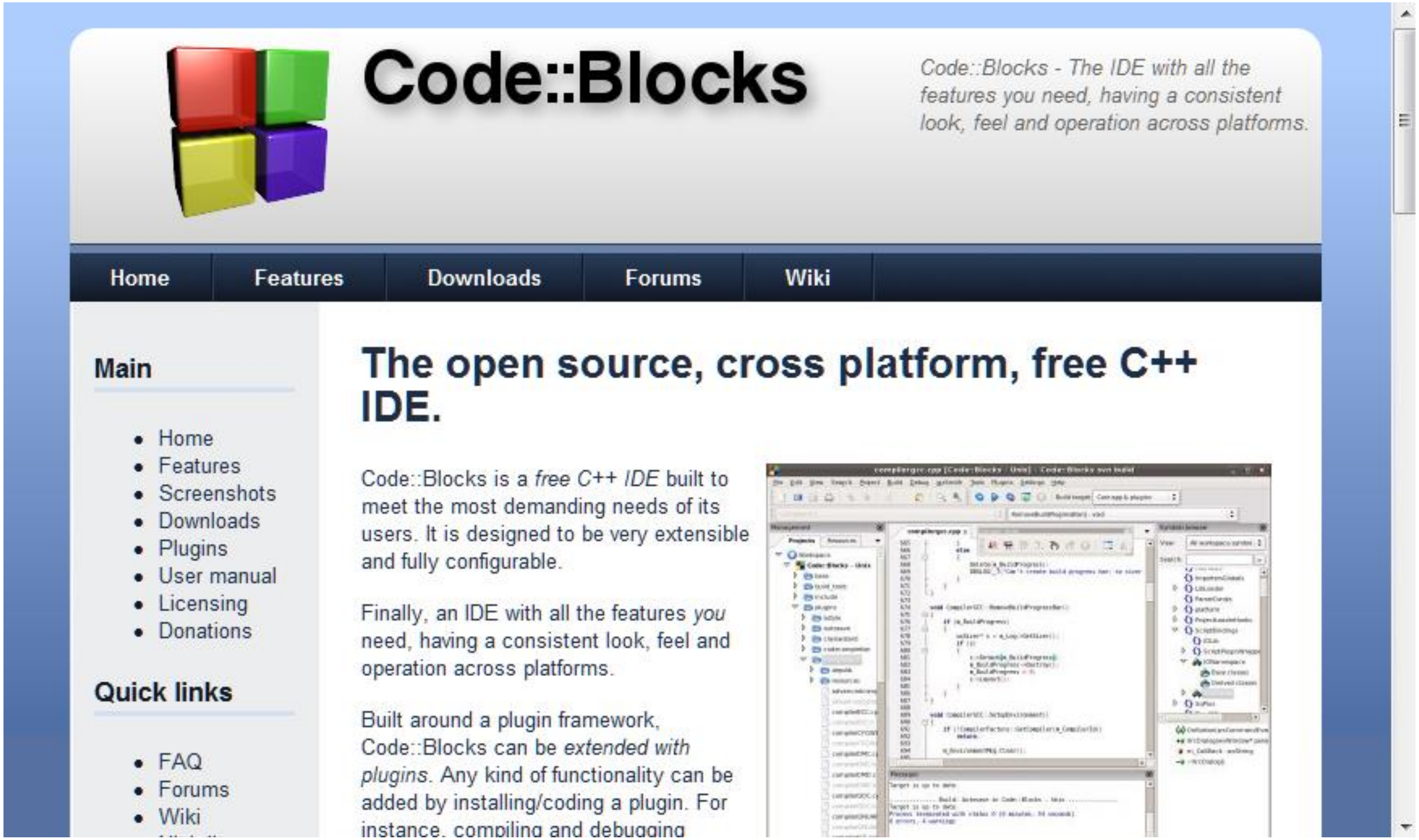
Code::Blocks使用介绍

一、下载安装程序

http://www.codeblocks.org/



Code::Blocks

Code::Blocks - The IDE with all the features you need, having a consistent look, feel and operation across platforms.

Main

- Home
- Features
- Screenshots
- Downloads
- Plugins
- User manual
- Licensing
- Donations

Quick links

- FAQ
- Forums
- Wiki

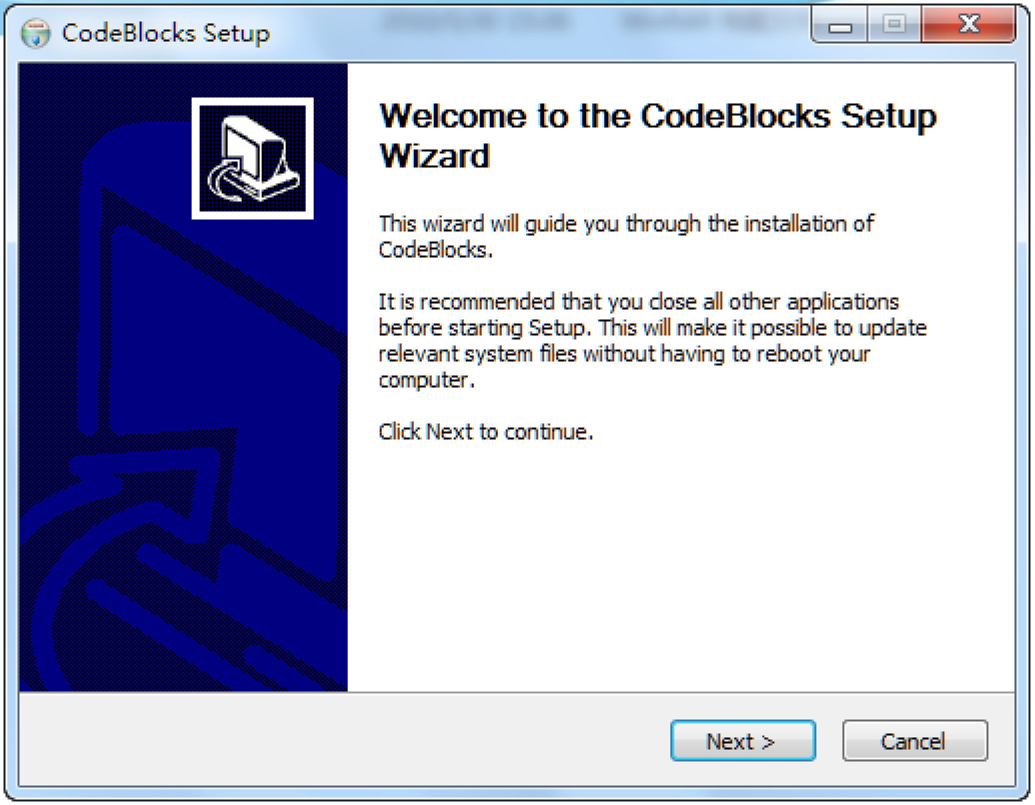
The open source, cross platform, free C++ IDE.

Code::Blocks is a *free C++ IDE* built to meet the most demanding needs of its users. It is designed to be very extensible and fully configurable.

Finally, an IDE with all the features *you* need, having a consistent look, feel and operation across platforms.

Built around a plugin framework, Code::Blocks can be *extended with plugins*. Any kind of functionality can be added by installing/coding a plugin. For instance, compiling and debugging

二、安装程序



Welcome to the CodeBlocks Setup Wizard

This wizard will guide you through the installation of CodeBlocks.

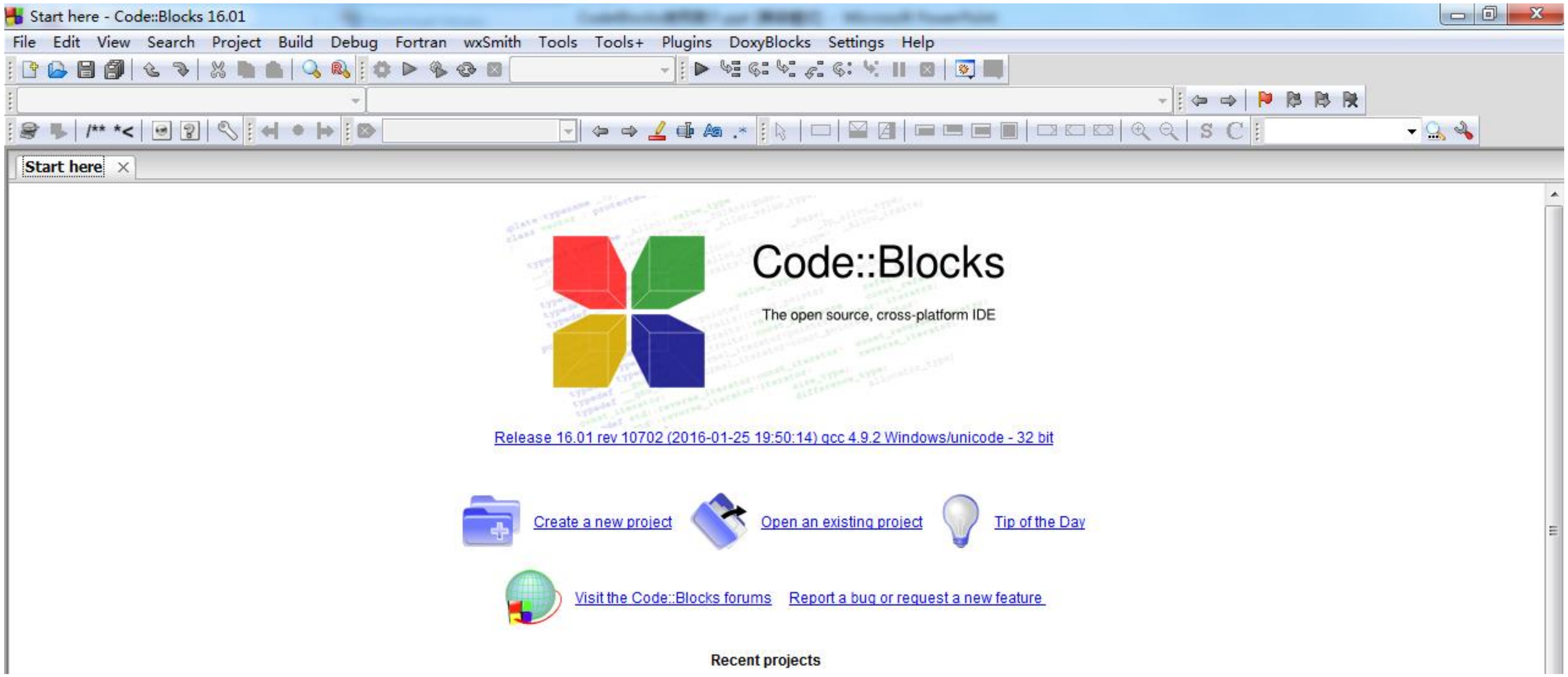
It is recommended that you close all other applications before starting Setup. This will make it possible to update relevant system files without having to reboot your computer.

Click Next to continue.

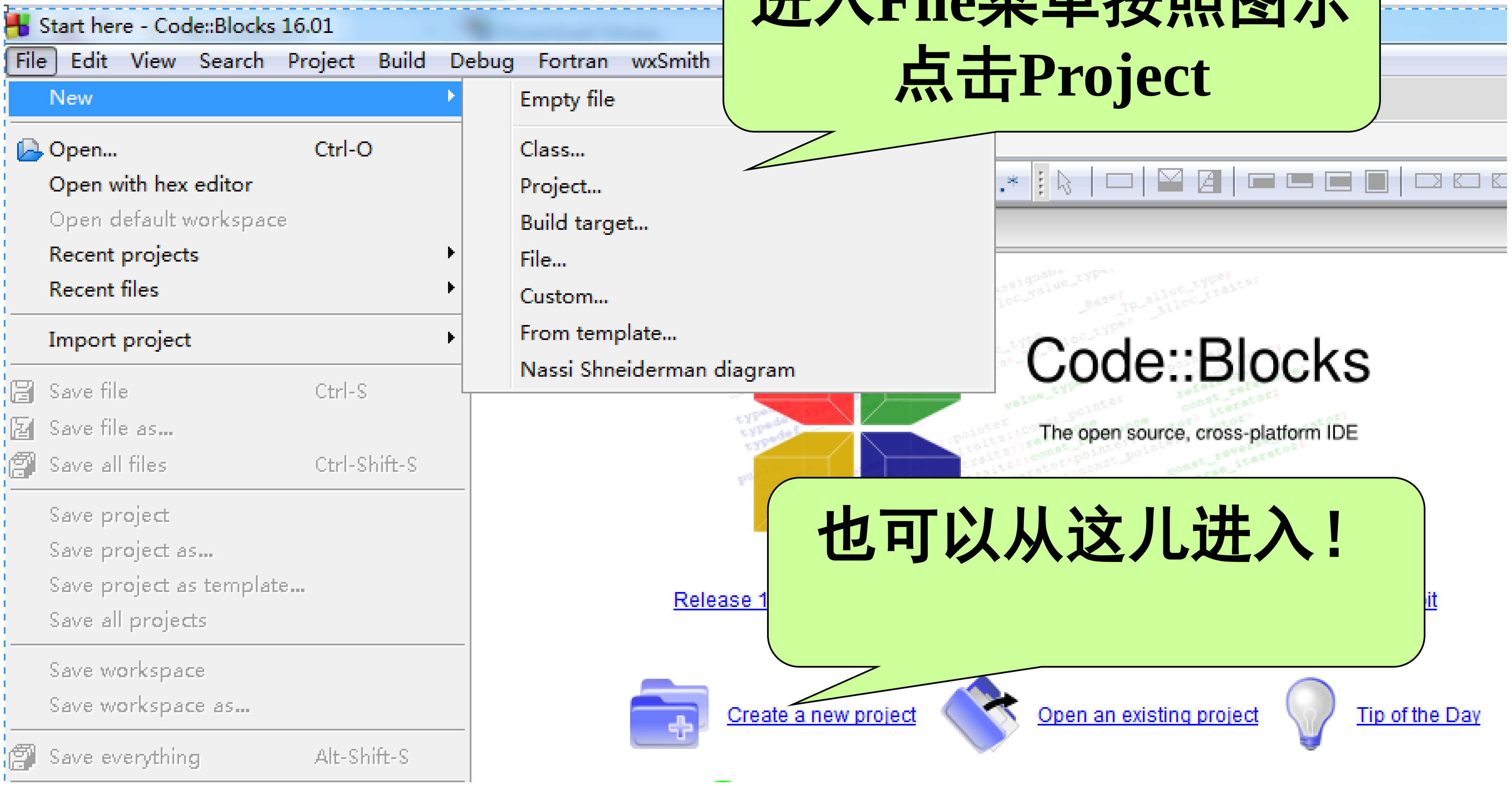
常用版本为：codeblocks-16.01mingw-setup。

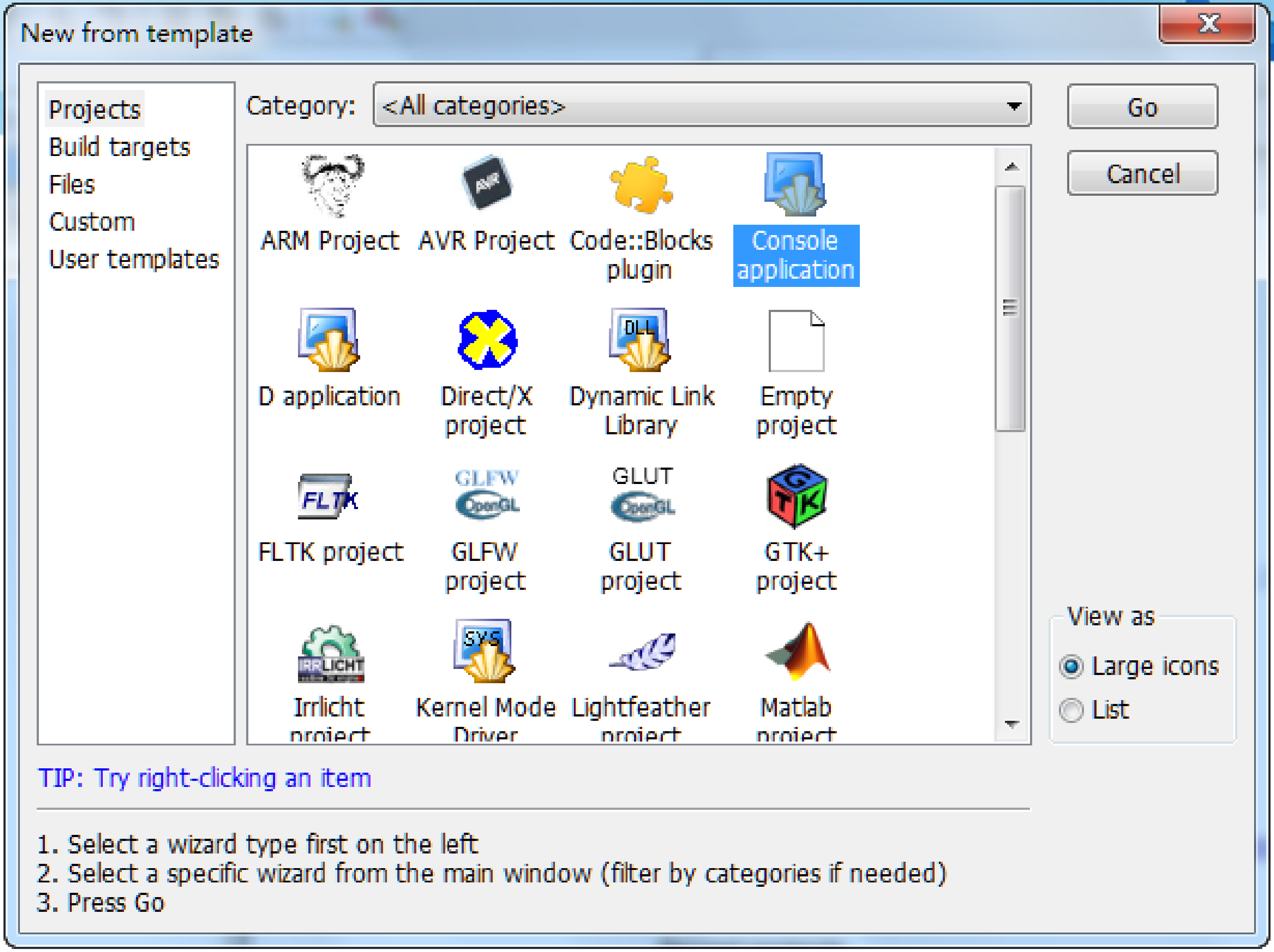
安装时请按安装程序窗口的提示一步步点击，直到安装完成。

启动后的窗口


Start here - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Start here
Code::Blocks
The open source, cross-platform IDE
Release 16.01 rev 10702 (2016-01-25 19:50:14) gcc 4.9.2 Windows/unicode - 32 bit
Create a new project
Open an existing project
Tip of the Day
Visit the Code::Blocks forums
Report a bug or request a new feature
Recent projects

三、建立工程


Start here - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith
New
Open... Ctrl-O
Open with hex editor
Open default workspace
Recent projects
Recent files
Import project
Save file Ctrl-S
Save file as...
Save all files Ctrl-Shift-S
Save project
Save project as...
Save project as template...
Save all projects
Save workspace
Save workspace as...
Save everything Alt-Shift-S
Empty file
Class...
Project...
Build target...
File...
Custom...
From template...
Nassi Shneiderman diagram
Code::Blocks
The open source, cross-platform IDE
Release 1
Create a new project
Open an existing project
Tip of the Day
进入File菜单按照图示点击Project
也可以从这儿进入！



New from template

Projects
Build targets
Files
Custom
User templates

Category: <All categories>

ARM Project, AVR Project, Code::Blocks plugin, Console application, D application, Direct/X project, Dynamic Link Library, Empty project, FLTK project, GLFW project, GLUT project, GTK+ project, Irrlicht project, Kernel Mode Driver, Lightfeather project, Matlab project

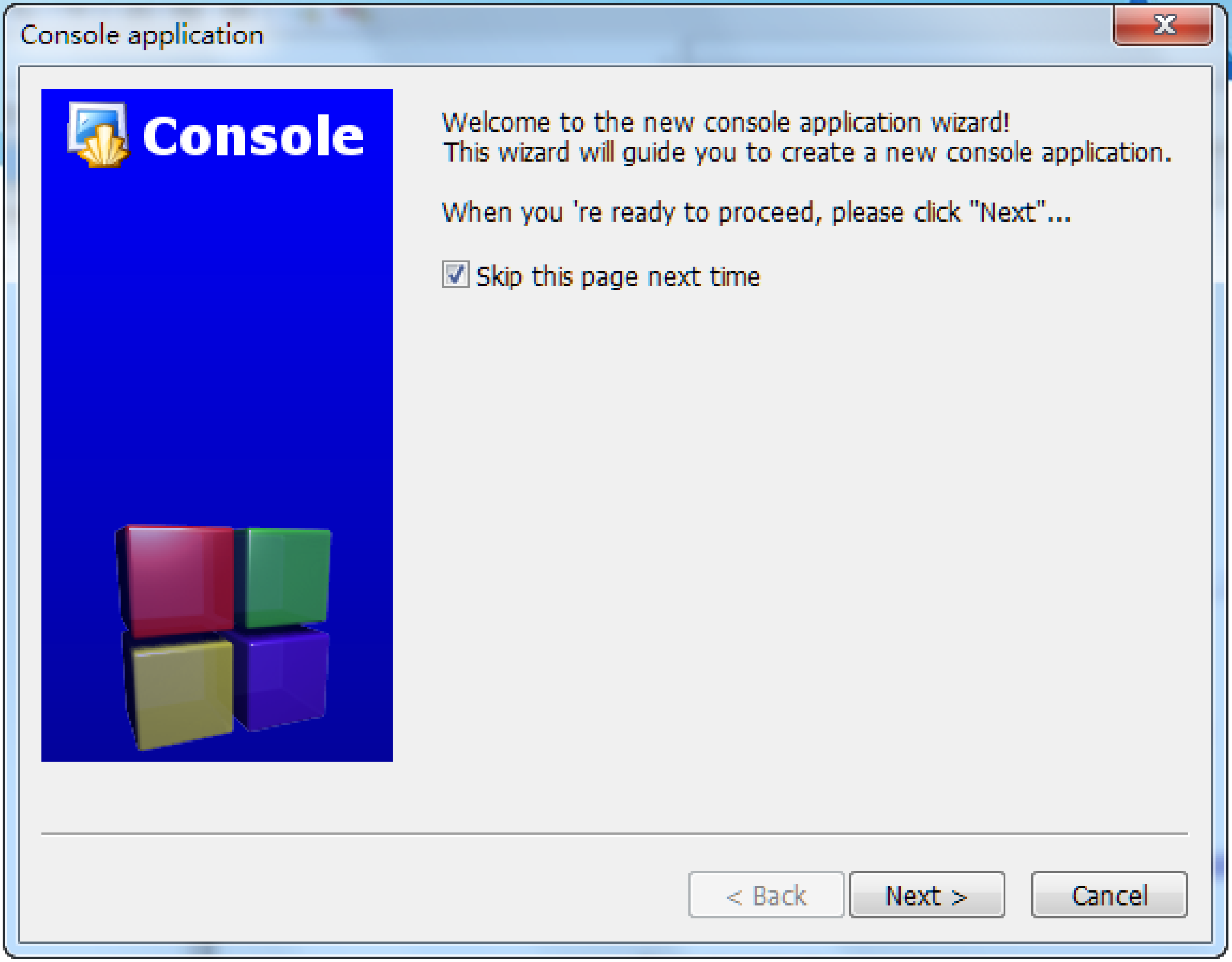
Go
Cancel

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

选择Console application(控制台应用程序)。

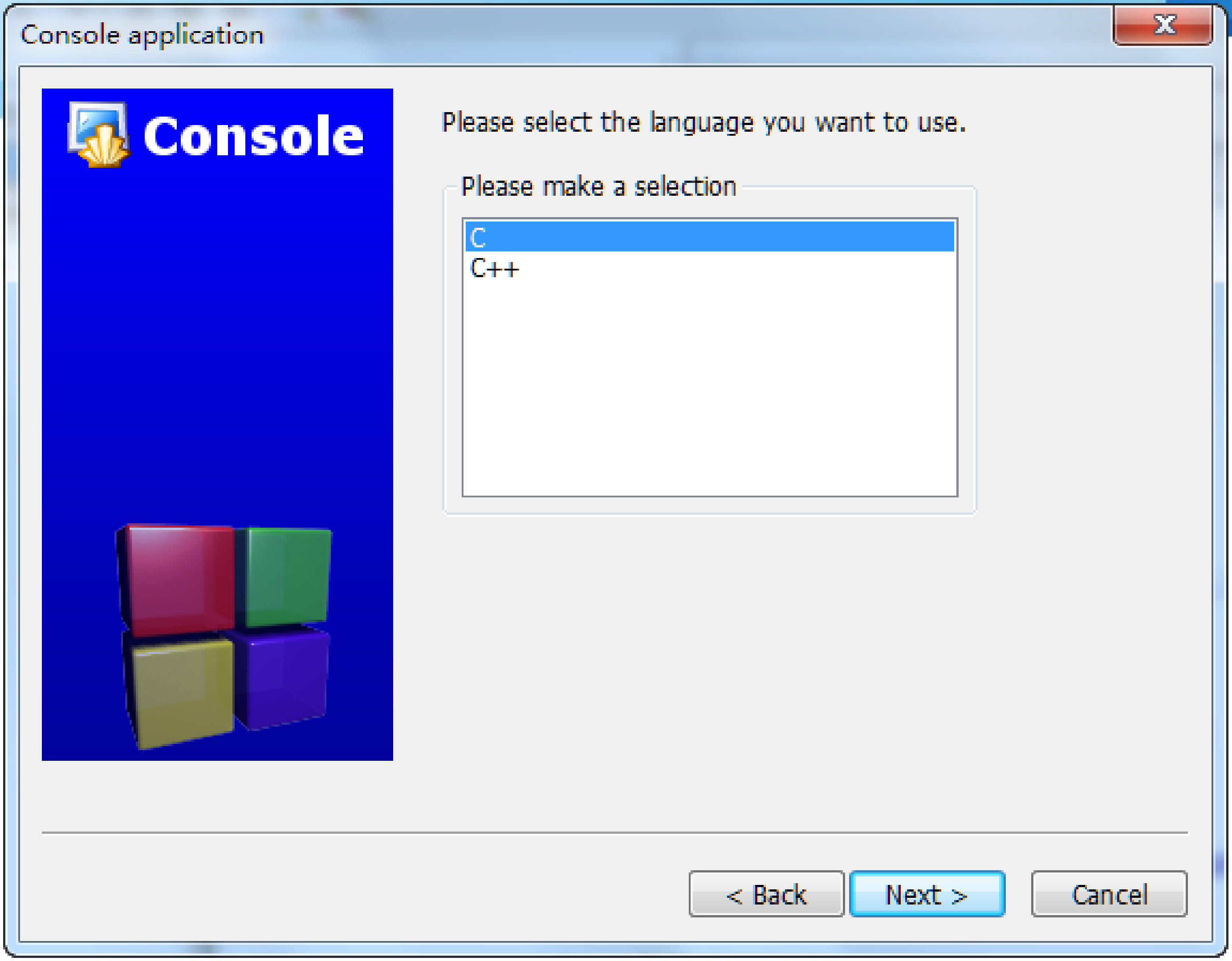

Console application

Welcome to the new console application wizard!
This wizard will guide you to create a new console application.

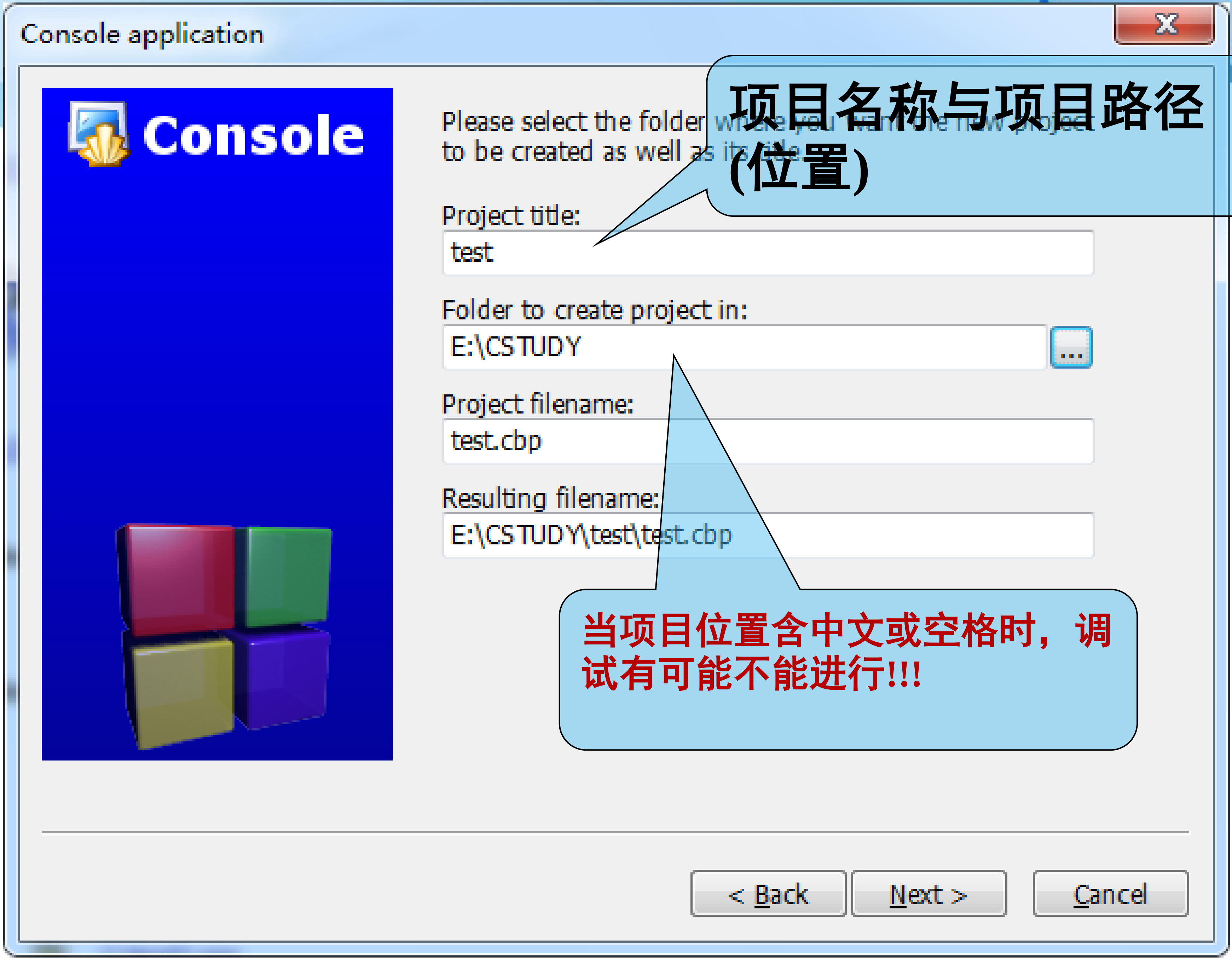
When you 're ready to proceed, please click "Next"...

[✓] Skip this page next time

为避免下次出现此窗口，选上“Skip this page next time”。



如果是学习C语言，则选择“C”。



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

Project title: test

Folder to create project in: E:\CSTUDY

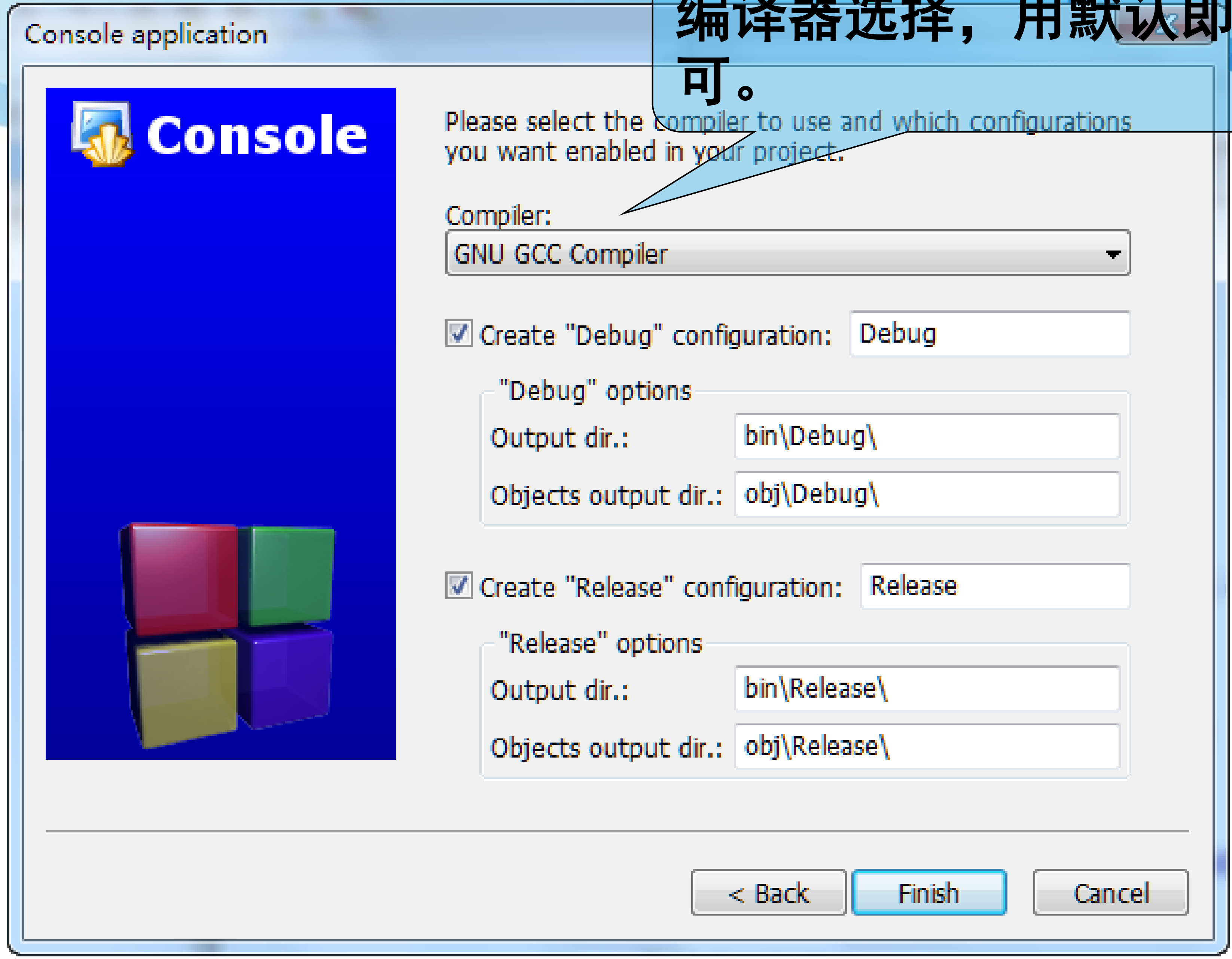
Project filename: test.cbp

Resulting filename: E:\CSTUDY\test\test.cbp

项目名称与项目路径(位置)

当项目位置含中文或空格时，调试有可能不能进行!!!

编译器选择，用默认即可。



Console application

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\

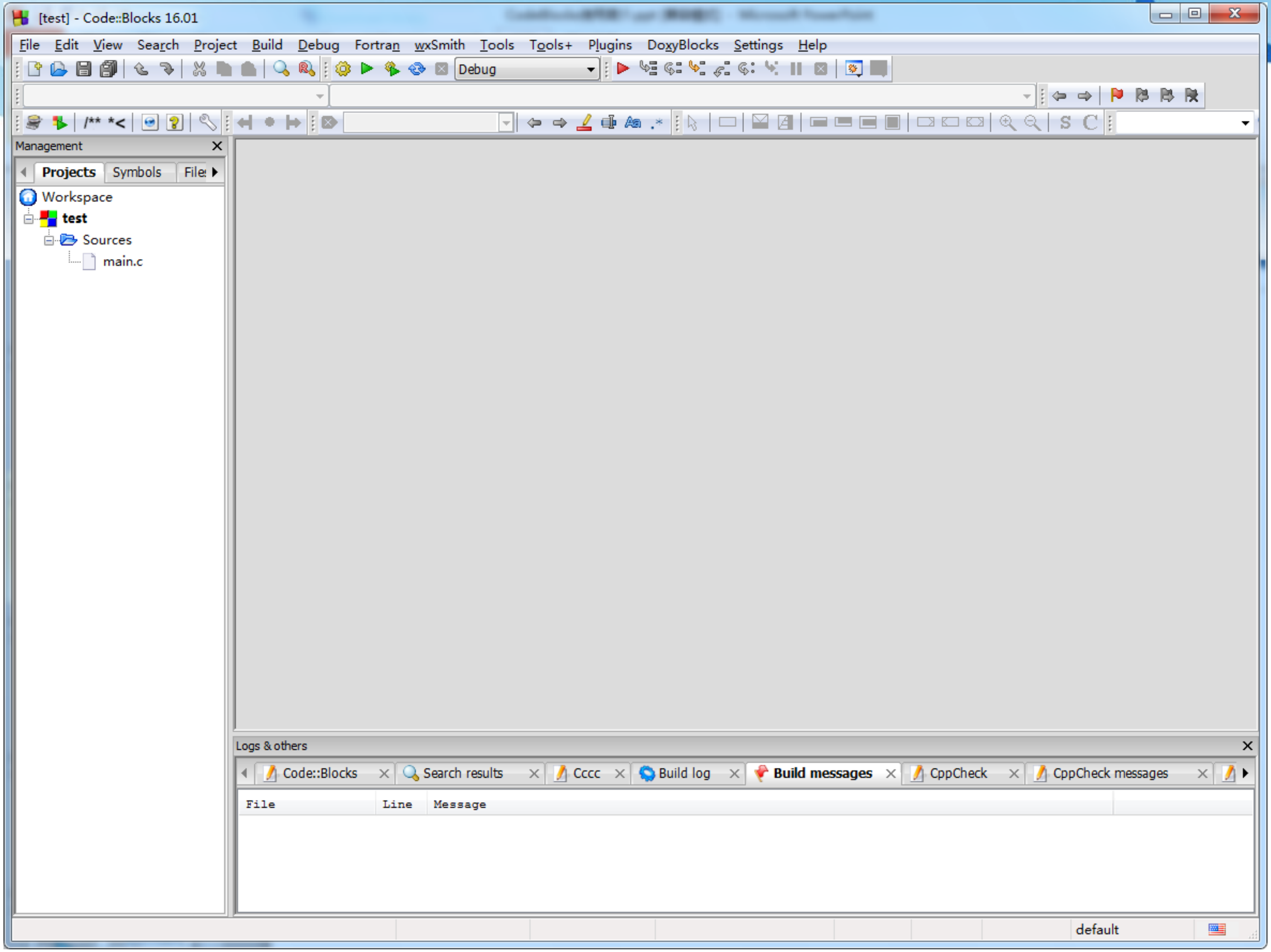
Objects output dir.: obj\Debug\

[✓] Create "Release" configuration: Release

"Release" options

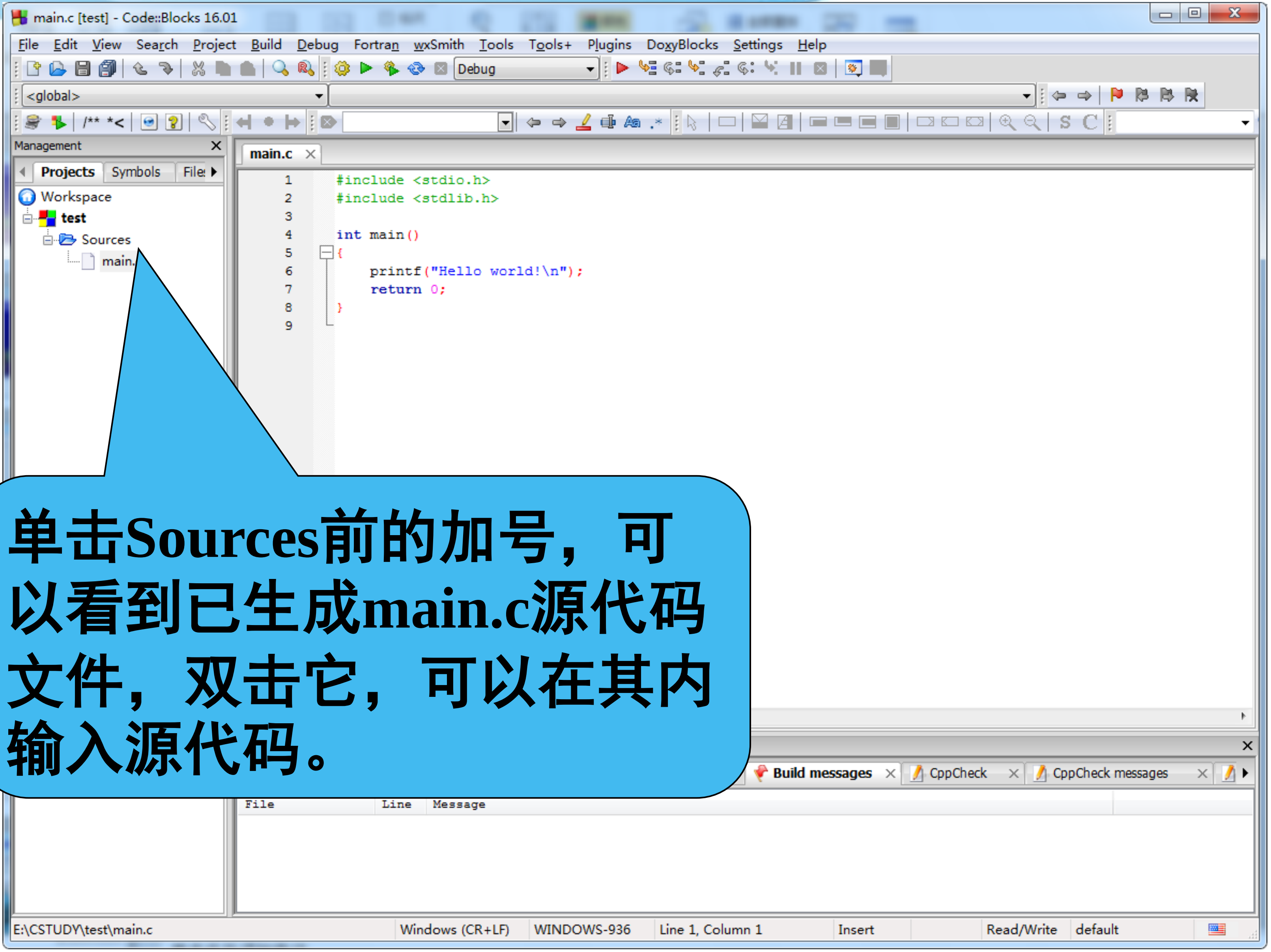
Output dir.: bin\Release\

Objects output dir.: obj\Release\


[test] - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
Management
Projects Symbols Files
Workspace
test
Sources
main.c
Logs & others
Code::Blocks Search results Cccc Build log Build messages CppCheck CppCheck messages
File Line Message
default

工程建立好后的界面。

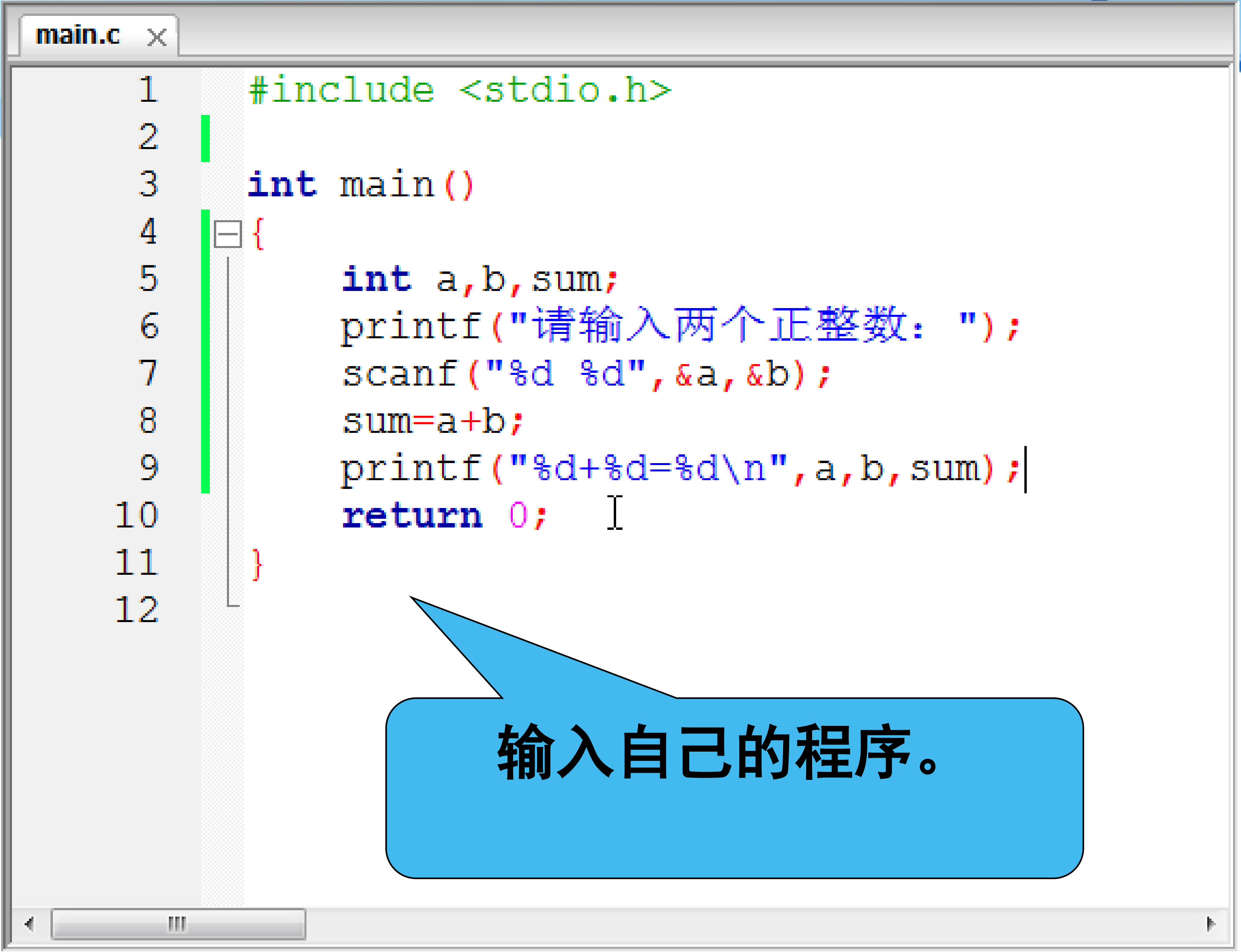
四、输入源程序(代码)


main.c [test] - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
Management
Projects Symbols Files
Workspace
test
Sources
main.c
Build messages
CppCheck
CppCheck messages
File Line Message

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

int main()
{
    printf("Hello world!\n");
    return 0;
}
```

E:\CSTUDY\test\main.c Windows (CR+LF) WINDOWS-936 Line 1, Column 1 Insert Read/Write default

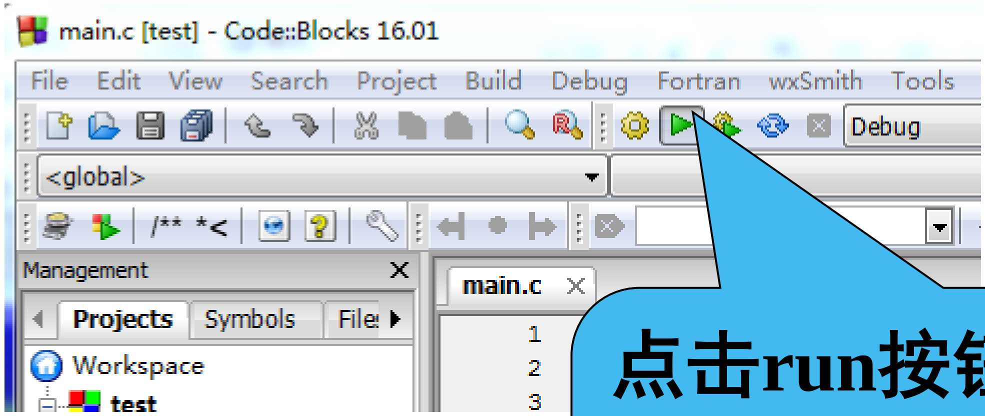
单击Sources前的加号，可以看到已生成main.c源代码文件，双击它，可以在其内输入源代码。



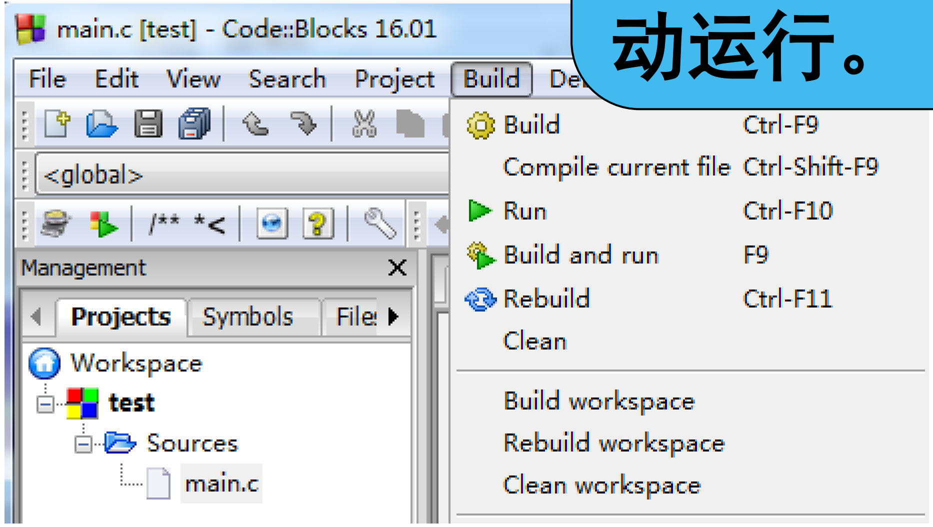
```c
#include <stdio.h>

int main()
{
    int a,b,sum;
    printf("请输入两个正整数: ");
    scanf("%d %d",&a,&b);
    sum=a+b;
    printf("%d+%d=%d\n",a,b,sum);
    return 0;
}
```

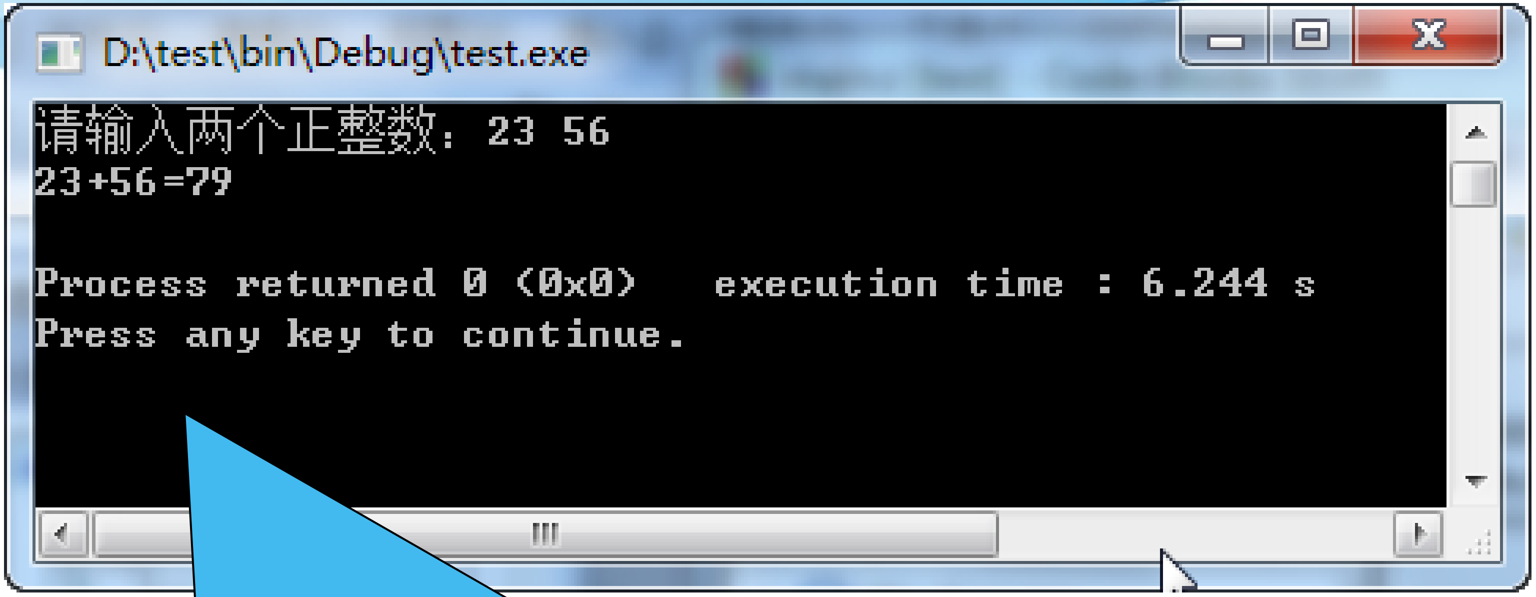
输入自己的程序。

五、运行程序


main.c [test] - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools
Debug
<global>
Management
Projects Symbols Files
Workspace
test
main.c

点击run按钮或进入下图所示的的Build菜单，程序开始编译，然后会自动运行。



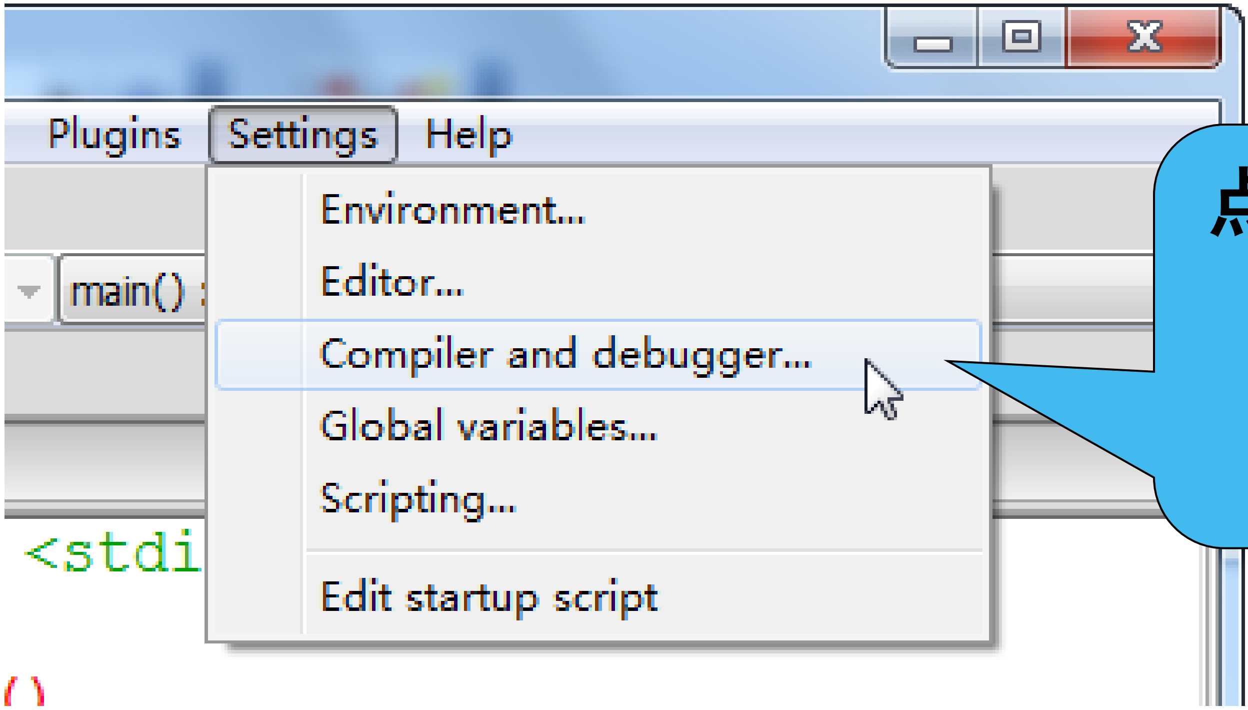


```
请输入两个正整数: 23 56
23+56=79

Process returned 0 (0x0)   execution time : 6.244 s
Press any key to continue.
```

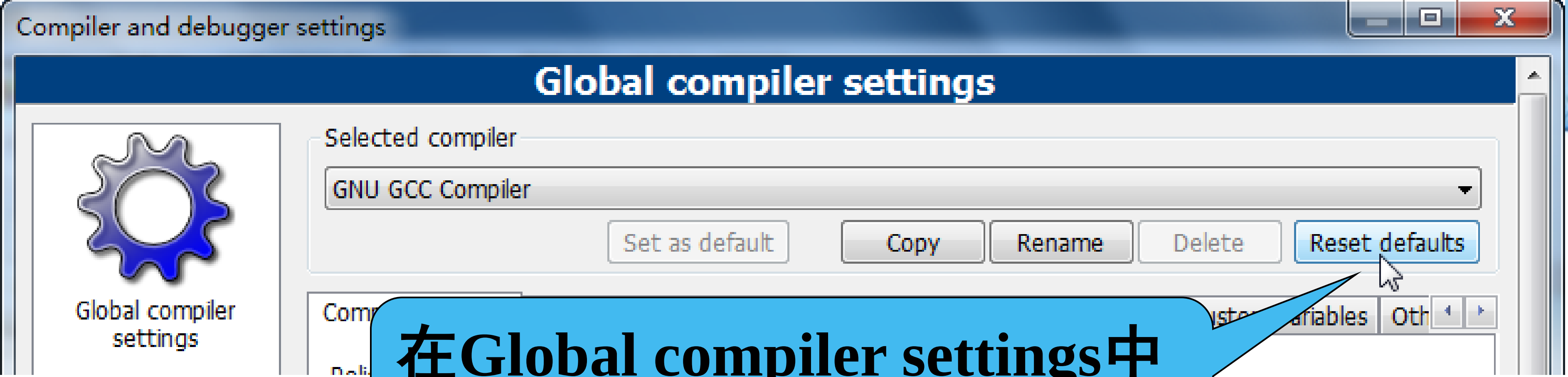
这是刚才程序代码的运行结果，最后两行不是该程序的运行结果，是CodeBlocks添加的结果，可以看到该程序的返回值和执行时长。并说明按任意键就能回到编辑程序界面。

六、关于设置

问题1：有时安装完成后，建立工程并写了程序开始编译运行，但CodeBlocks没有反应。通常是编译器路径不对的问题。大家可以这样修改。



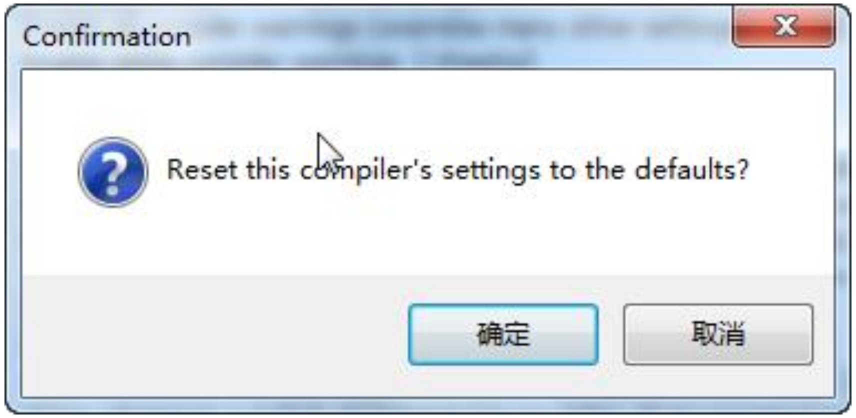
点击Compiler and debugger…


Compiler and debugger settings

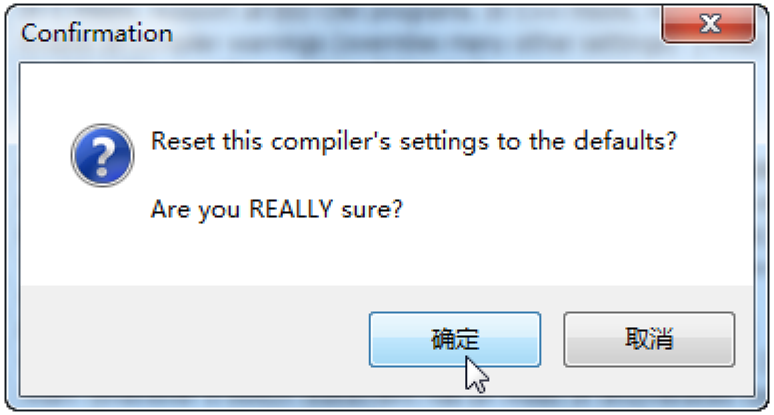
Global compiler settings

Selected compiler: GNU GCC Compiler

在Global compiler settings中点击Reset default


Confirmation

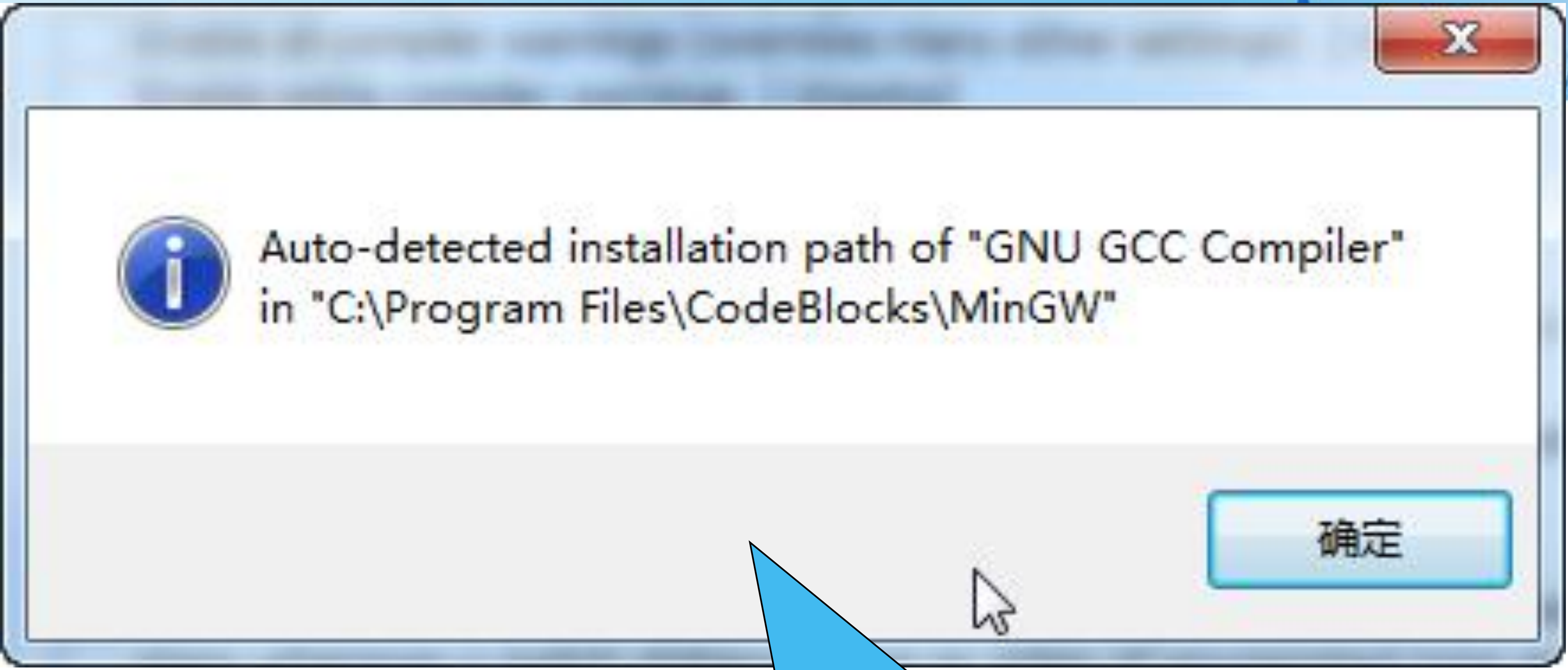
Reset this compiler's settings to the defaults?


Confirmation

Reset this compiler's settings to the defaults?

Are you REALLY sure?

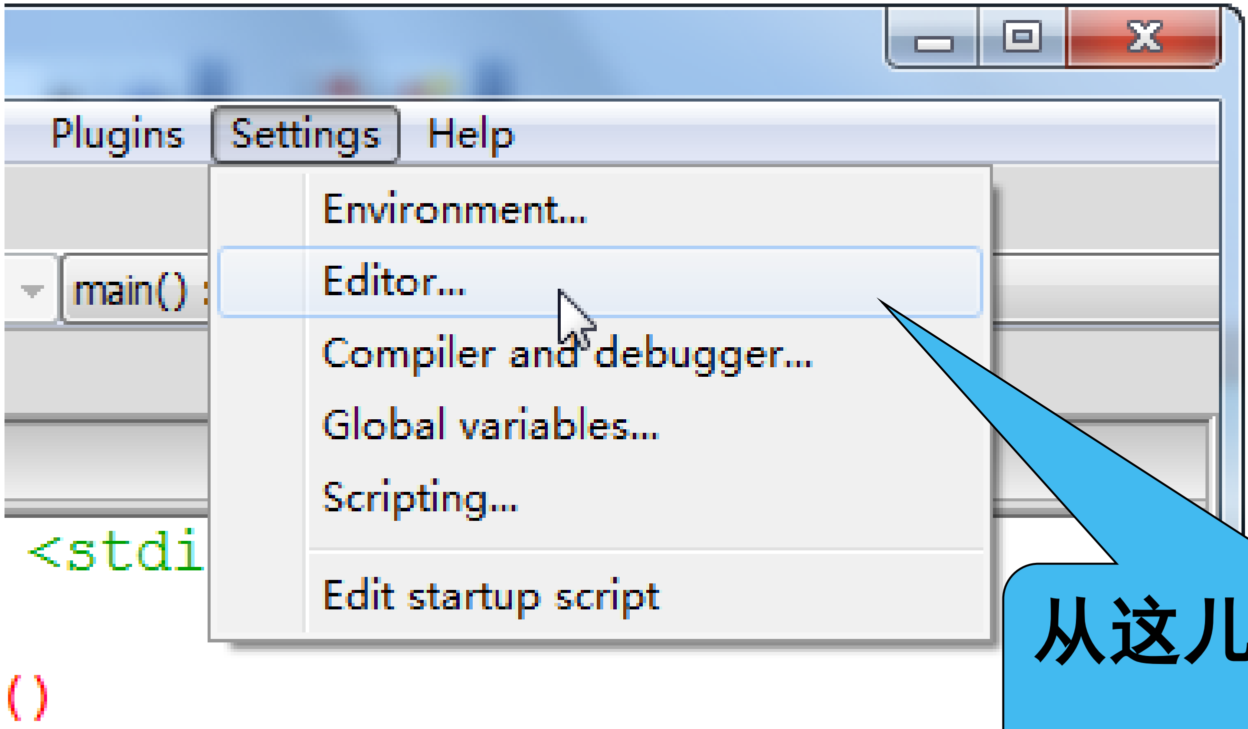
均点击确定。



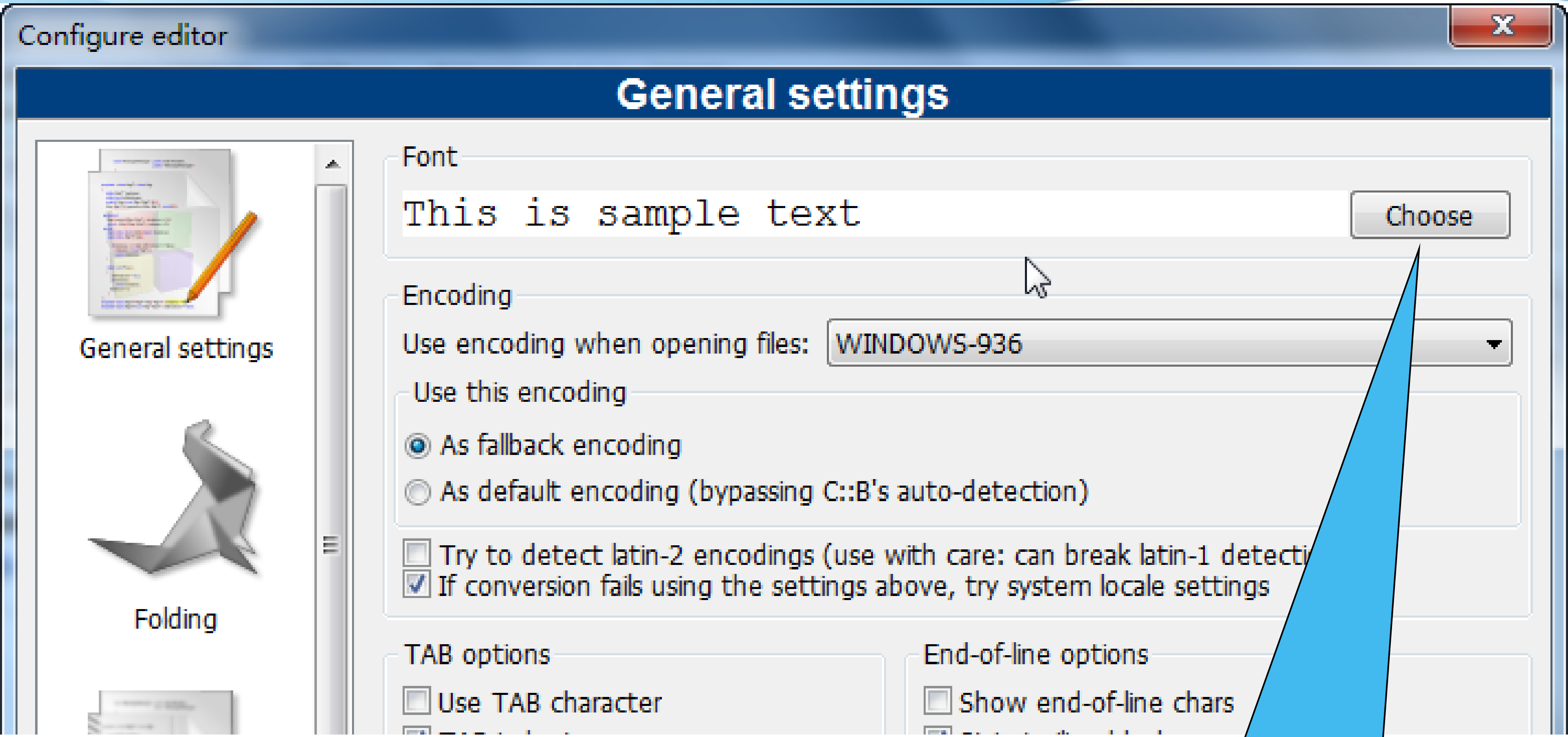
Auto-detected installation path of "GNU GCC Compiler" in "C:\Program Files\CodeBlocks\MinGW"

这时它会自动探测默认编译器的位置，从而就可以使用该编译器。

问题2：如何设置编辑器字体及大小？



从这儿进入


Configure editor

General settings

Font

This is sample text

Choose

Encoding

Use encoding when opening files: WINDOWS-936

Use this encoding

[✓ As fallback encoding]

As default encoding (bypassing C::B's auto-detection)

☐ Try to detect latin-2 encodings (use with care: can break latin-1 detecti

☑ If conversion fails using the settings above, try system locale settings

TAB options

☐ Use TAB character

End-of-line options

☐ Show end-of-line chars

点击Choose进入即可进行设置。

七、程序的调试

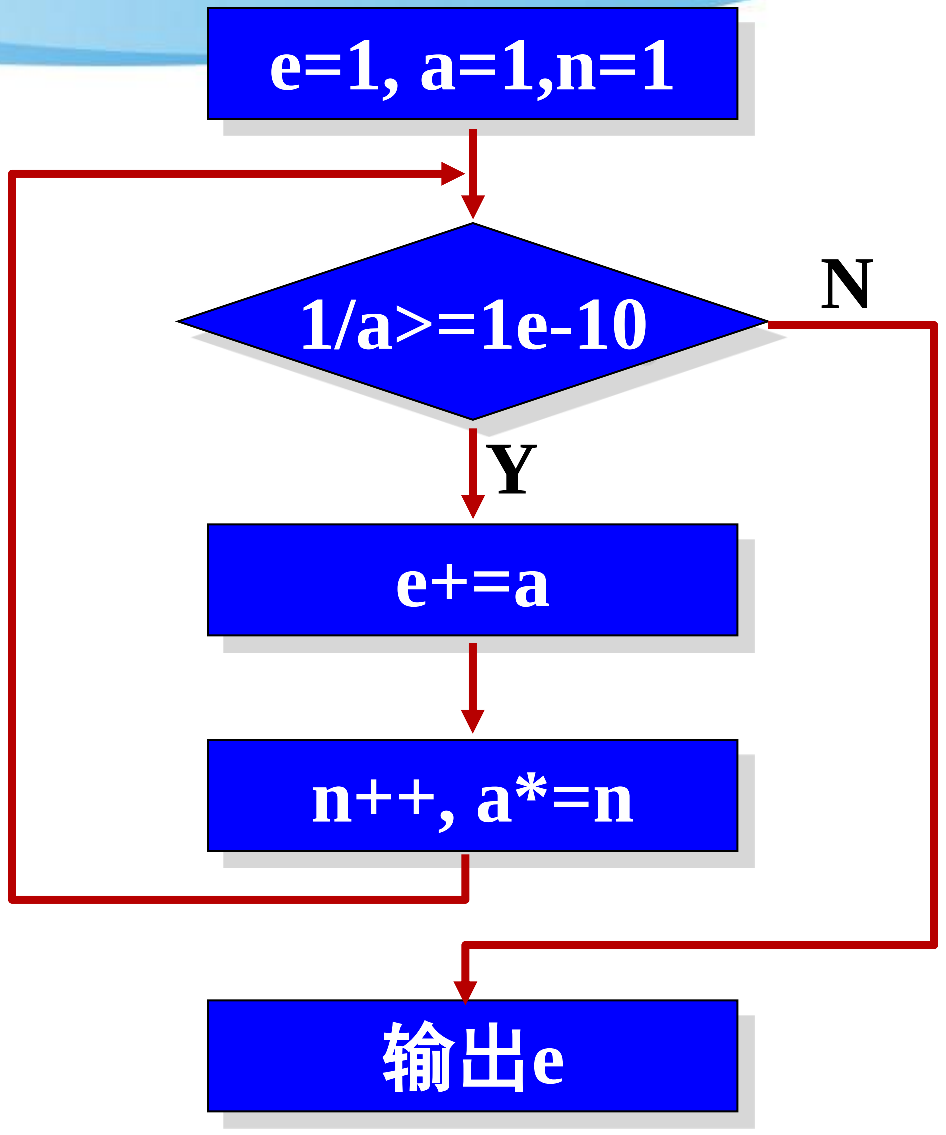
以下例说明在Code::Blocks中关于程序调试的简单方法。

例 由级数知识，$e \approx 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \cdots + \frac{1}{n!}$,

由此编写程序求e，直到右式中最后一项小于10^{-10}。

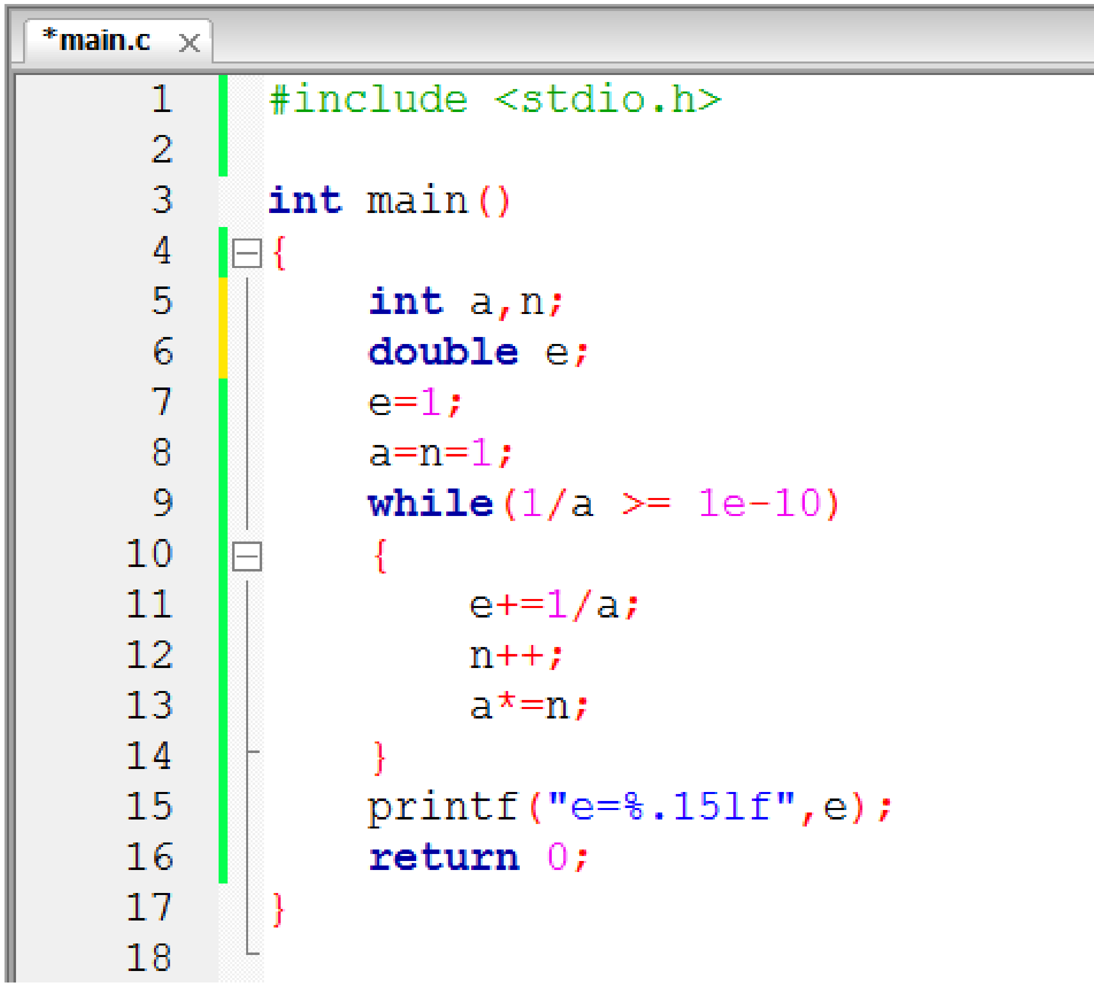
分析：由上面右式，看出是多项连加，并且项的生成有规律，所以考虑用循环实现连加。

如果把1/1!看作第1项，用变量a表示当前要加的数的分母，当a是第k项分母时，则第k+1项分母可表达为：a*=k+1。由此得到下面流程图。


e=1, a=1,n=1
1/a>=1e-10
N
Y
e+=a
n++, a*=n
输出e

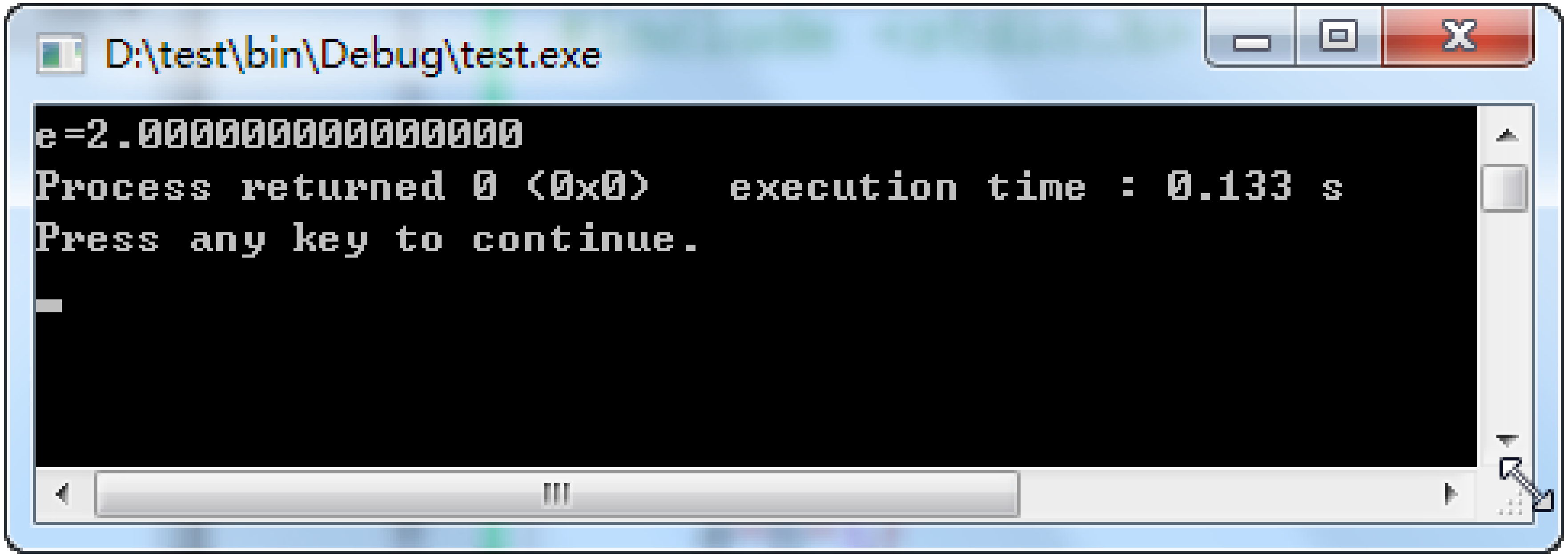
程序如下：

```
#include <stdio.h>
int main()
{
    int a,n;
    double e;
    e=1;
    a=n=1;
    while(1/a >= 1e-10)
    {
        e+=1/a;
        n++;
        a*=n;
    }
    printf("e=%.15lf",e);
    return 0;
}
```



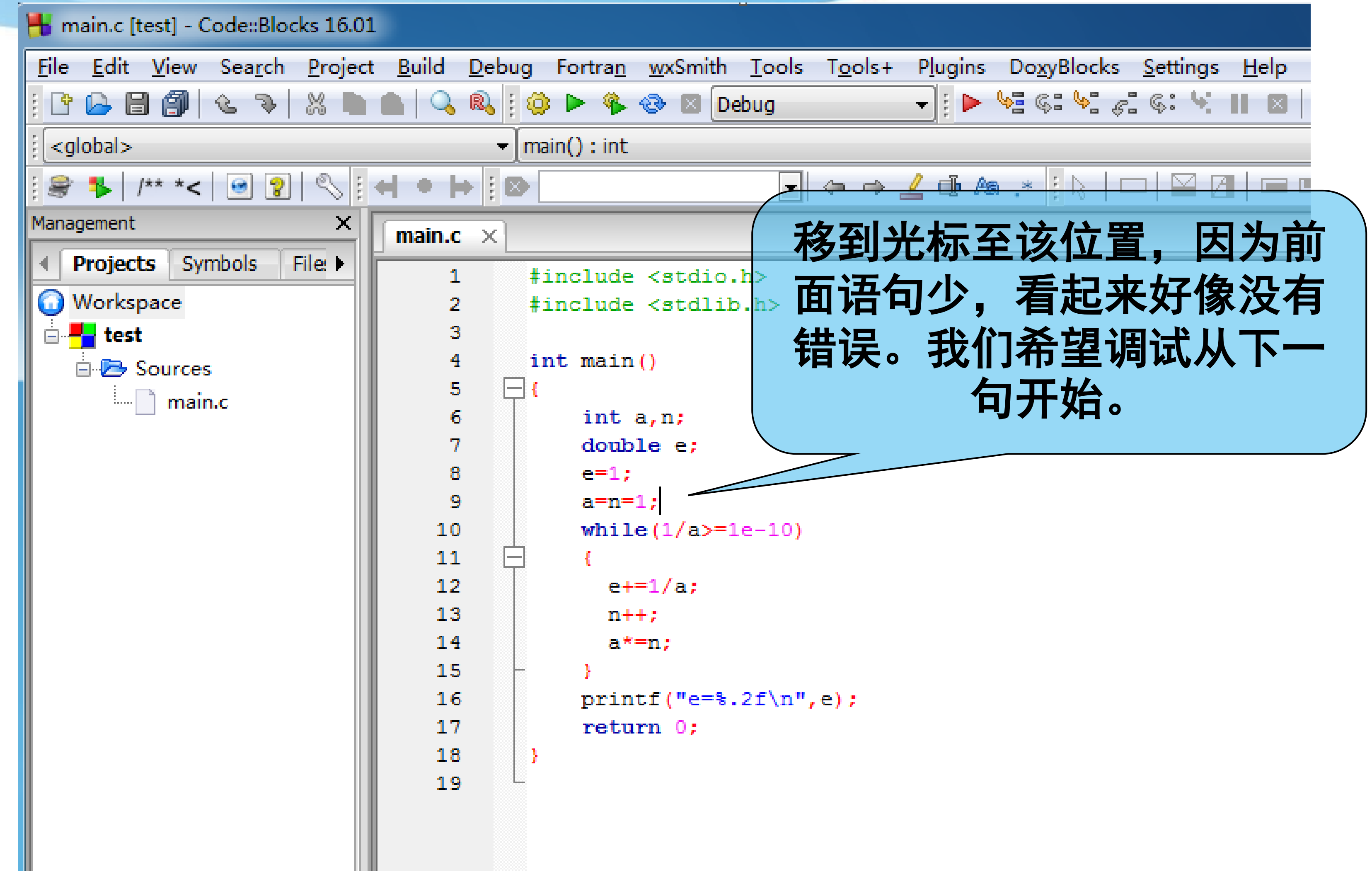
```
#include <stdio.h>

int main()
{
    int a,n;
    double e;
    e=1;
    a=n=1;
    while(1/a >= 1e-10)
    {
        e+=1/a;
        n++;
        a*=n;
    }
    printf("e=%.15lf",e);
    return 0;
}
```

按F9，程序运行结果如下：



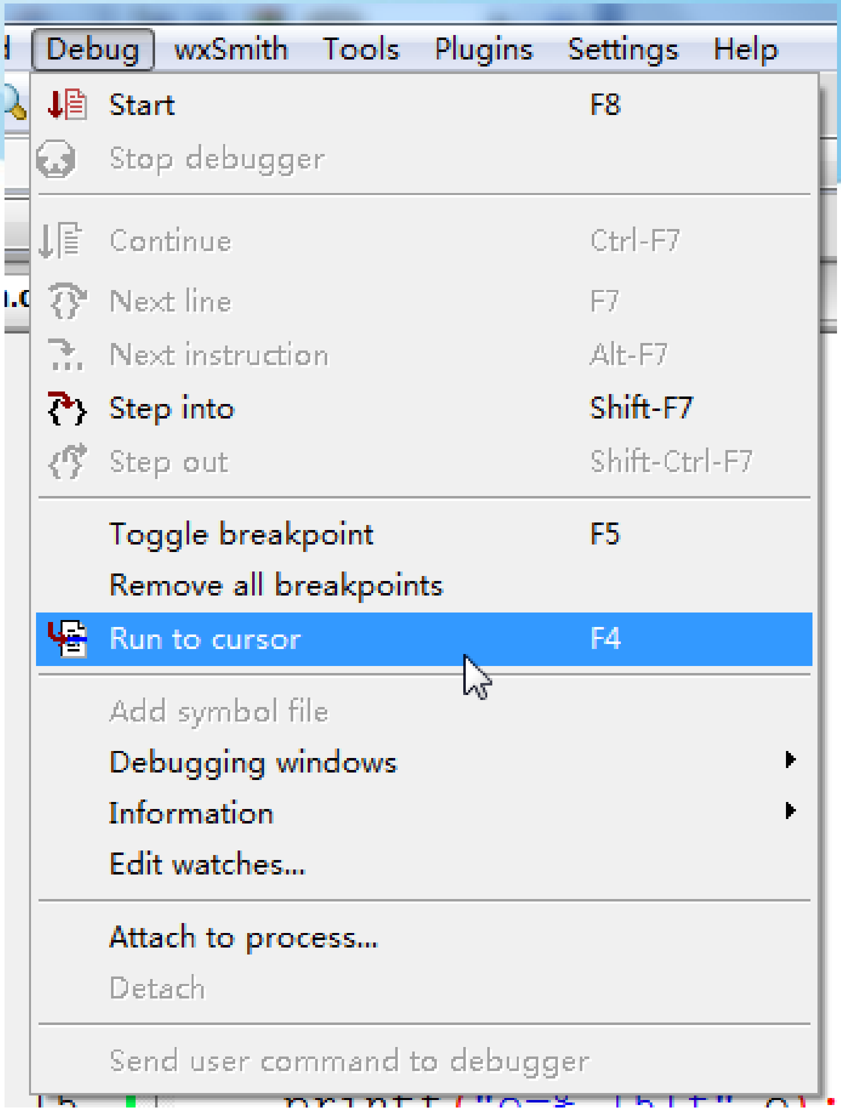
```
e=2.000000000000000
Process returned 0 (0x0)   execution time : 0.133 s
Press any key to continue.
```

显然结果是错误的，下面我们进行调试排错。

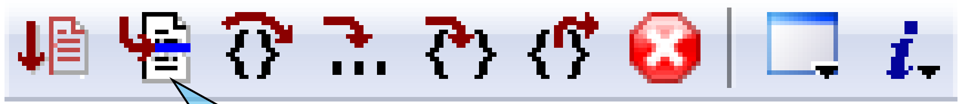

main.c [test] - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main() : int
Management
Projects Symbols Files
Workspace
test
Sources
main.c

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

int main()
{
    int a,n;
    double e;
    e=1;
    a=n=1;
    while(1/a>=1e-10)
    {
      e+=1/a;
      n++;
      a*=n;
    }
    printf("e=%.2f\n",e);
    return 0;
}
```

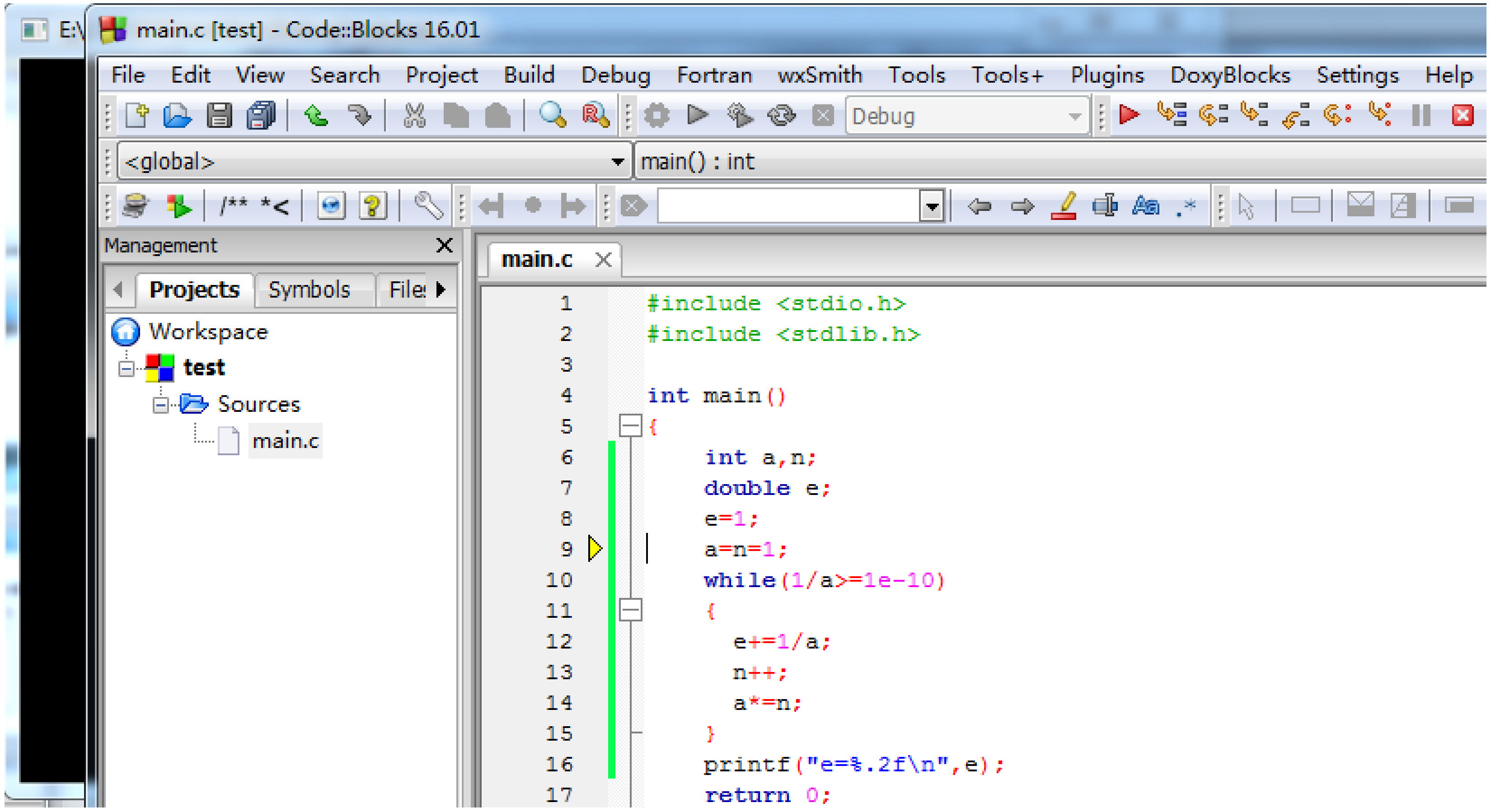
移到光标至该位置，因为前面语句少，看起来好像没有错误。我们希望调试从下一句开始。



进入Debug(调试)菜单，点击Run to cursor(运行至光标)，以后也可按F4。



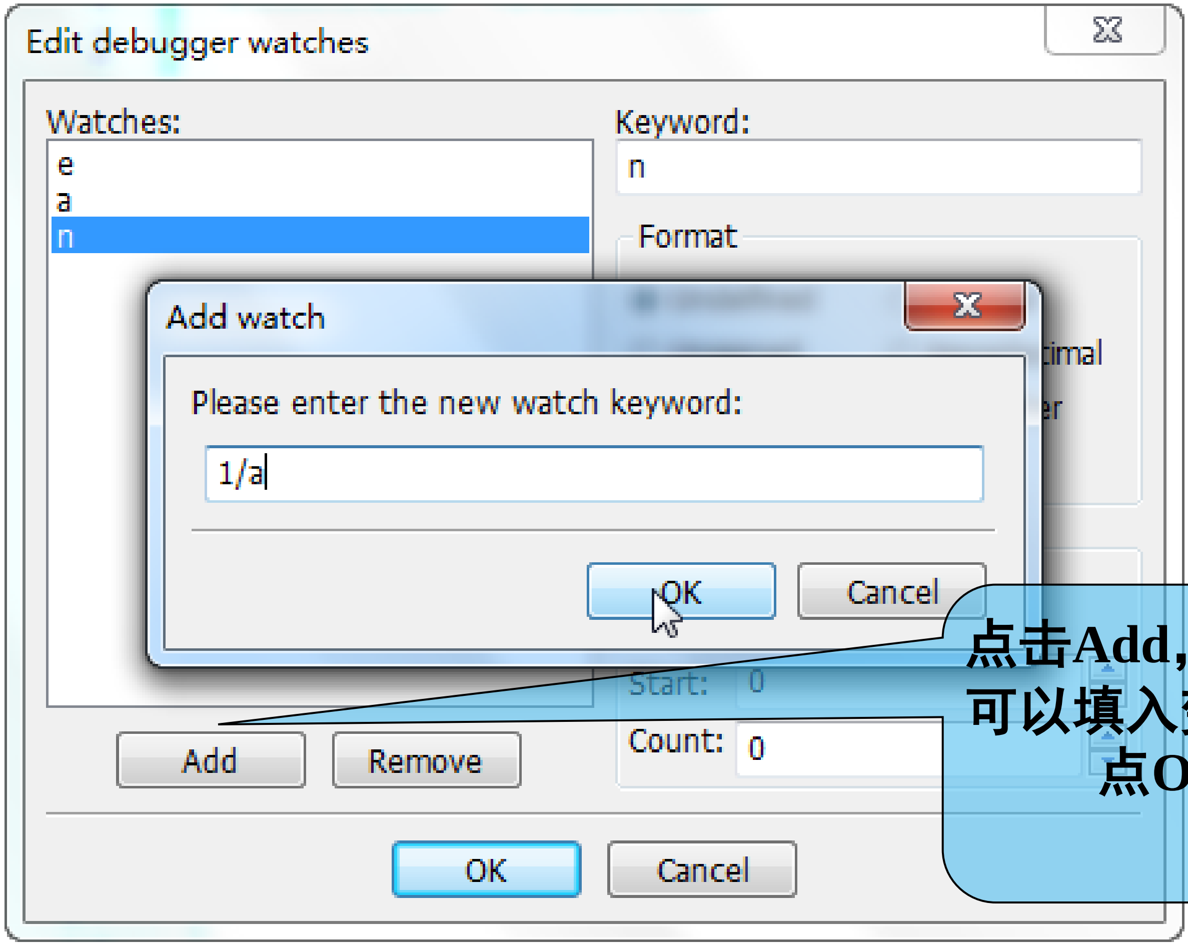
也可以找到此工具栏(调试工具栏)，点击该图标，运行至光标处。


main.c [test] - Code::Blocks 16.01
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
main() : int
Management
Projects Symbols Files
Workspace
test
Sources
main.c
main.c

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

int main()
{
    int a,n;
    double e;
    e=1;
    a=n=1;
    while(1/a>=1e-10)
    {
      e+=1/a;
      n++;
      a*=n;
    }
    printf("e=%.2f\n",e);
    return 0;
```

屏幕变成此图，后面的黑色窗口说明程序正在运行，全黑说明还没有任何输出。前面编辑窗口中的第9行前的小三角形说明已经运行到行。

为了看清程序运行过程中变量的变化，可以添加变量的观察。从Debug菜单进去，点击Edit watchs…



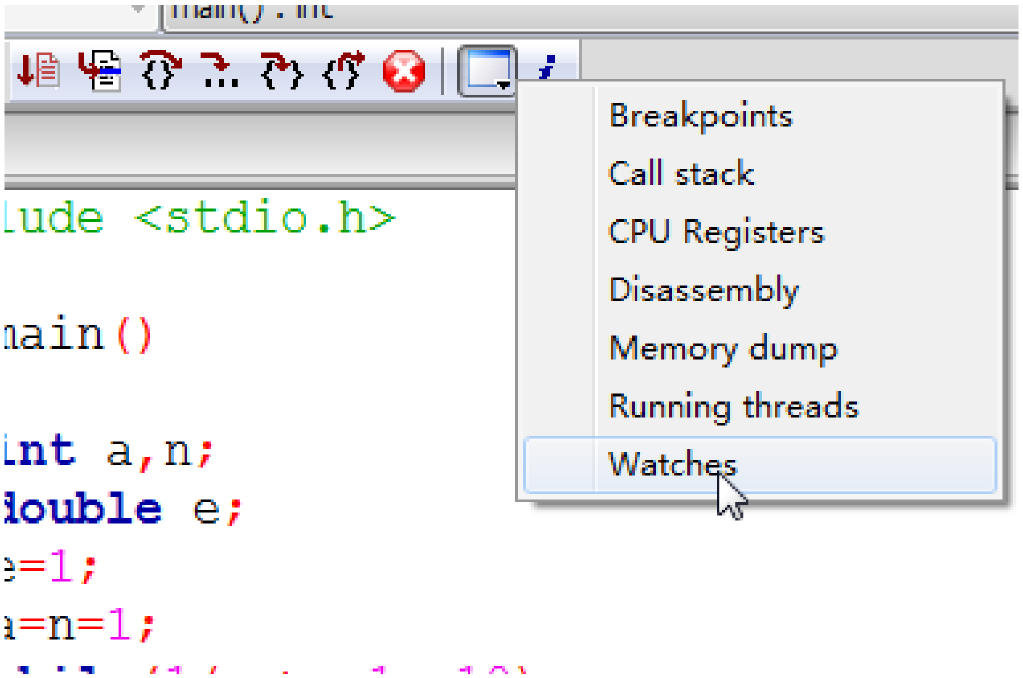
Please enter the new watch keyword:

1/a

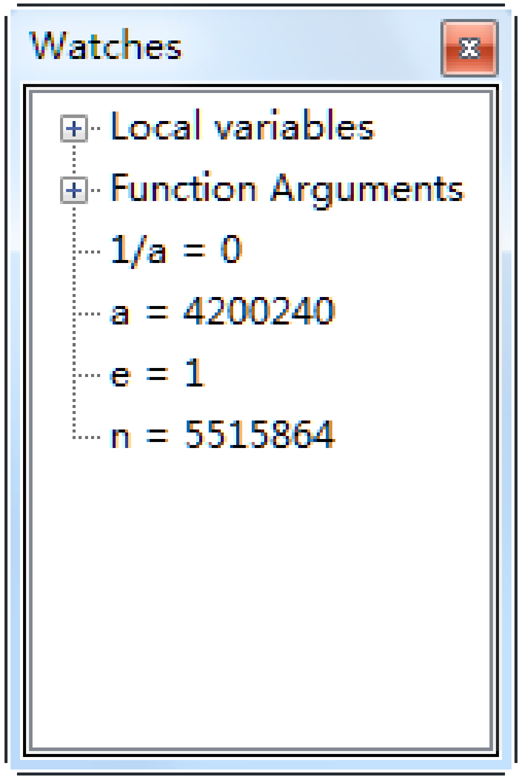
点击Add，弹出上面的窗口，可以填入变量或变量表达式，点OK就可加入。

添加完成后，点击OK退出。

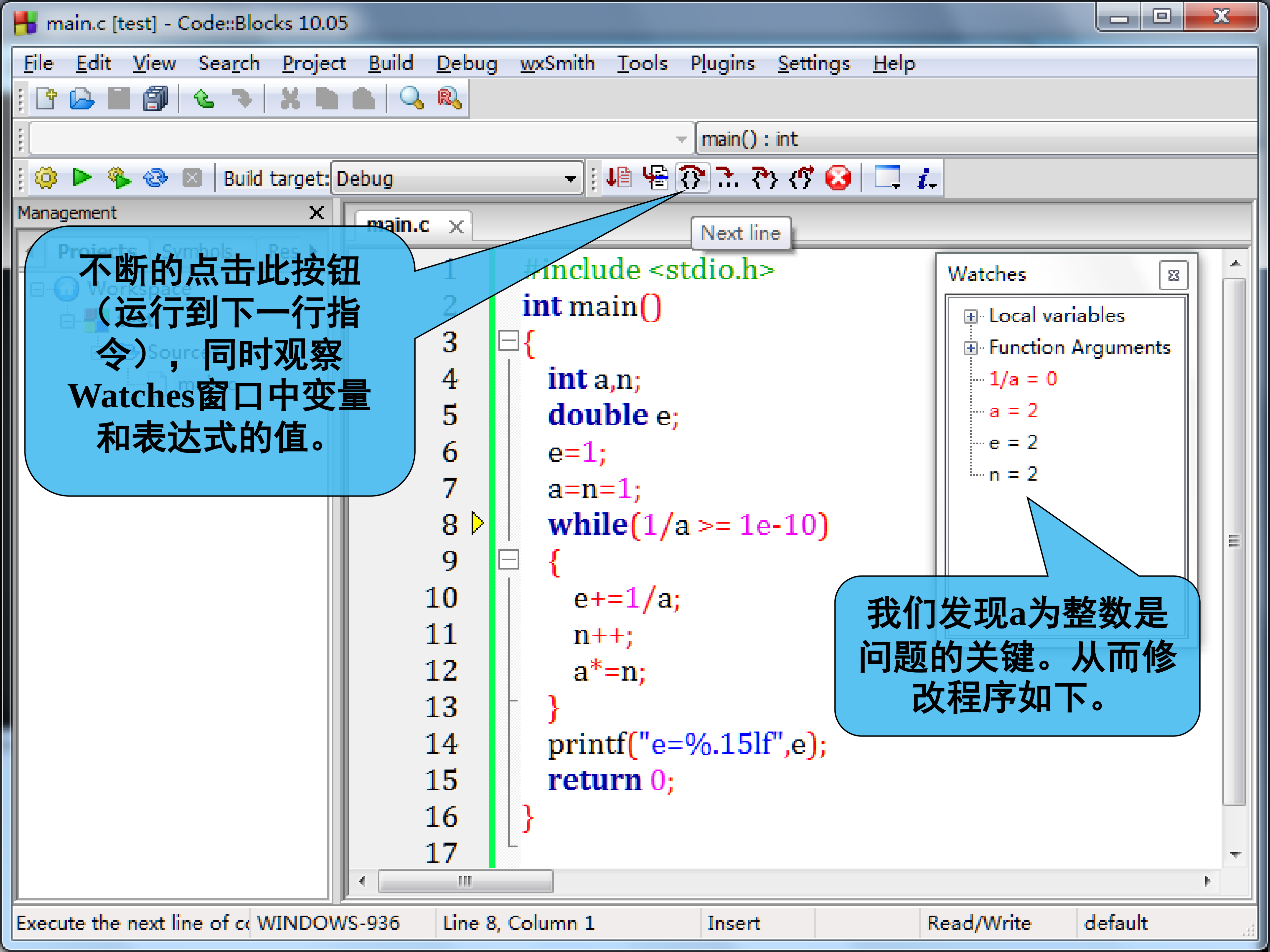
为了能看见变量的值，按如图所示调出Watches窗口。



```
lude <stdio.h>

nain()

int a,n;
double e;
e=1;
a=n=1;
```



```
Local variables
Function Arguments
1/a = 0
a = 4200240
e = 1
n = 5515864
```


main.c [test] - Code::Blocks 10.05
File Edit View Search Project Build Debug wxSmith Tools Plugins Settings Help
main() : int
Build target: Debug
Management
main.c
Next line
Watches
Local variables
Function Arguments

```c
#include <stdio.h>
int main()
{
    int a,n;
    double e;
    e=1;
    a=n=1;
    while(1/a >= 1e-10)
    {
        e+=1/a;
        n++;
        a*=n;
    }
    printf("e=%.15lf",e);
    return 0;
}
```

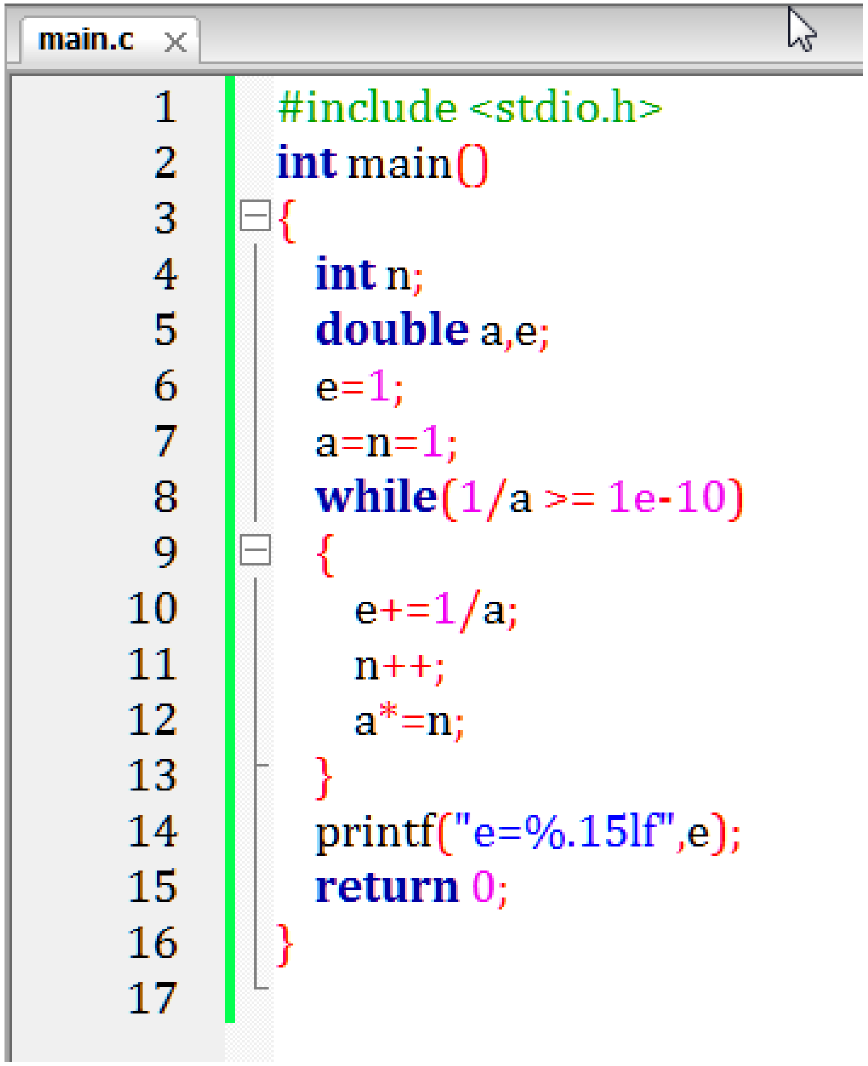
```
1/a = 0
a = 2
e = 2
n = 2
```

```
Execute the next line of cc WINDOWS-936   Line 8, Column 1   Insert   Read/Write   default
```

不断的点击此按钮（运行到下一行指令），同时观察Watches窗口中变量和表达式的值。

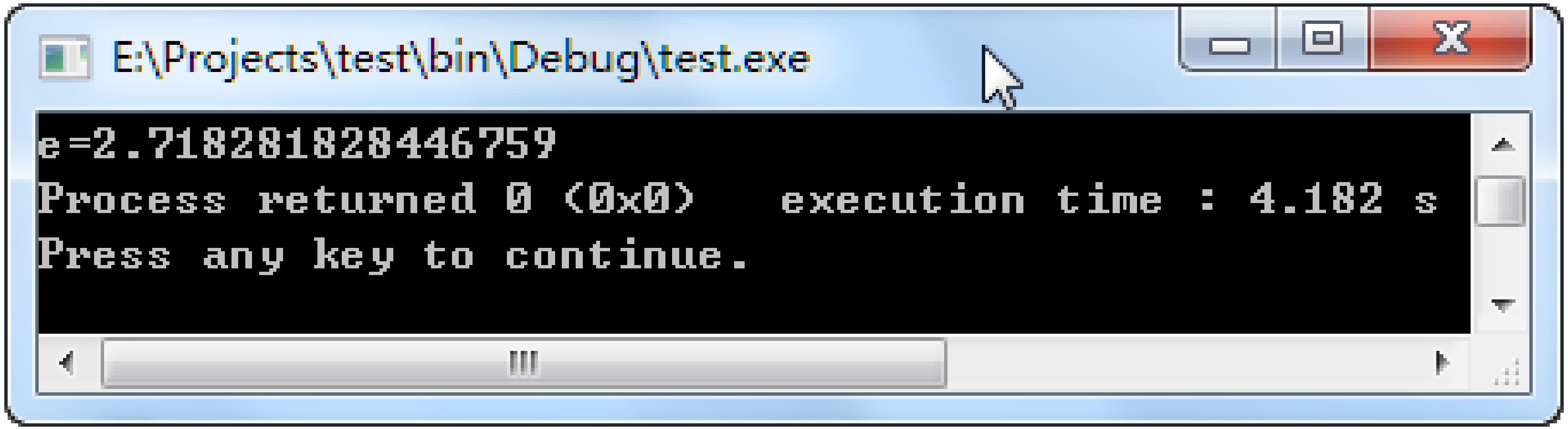
我们发现a为整数是问题的关键。从而修改程序如下。

```
#include <stdio.h>
int main()
{
    int n;
    double a,e;
    e=1;
    a=n=1;
    while(1/a >= 1e-10)
    {
        e+=1/a;
        n++;
        a*=n;
    }
    printf("e=%.15lf",e);
    return 0;
}
```



```
#include <stdio.h>
int main()
{
   int n;
   double a,e;
   e=1;
   a=n=1;
   while(1/a >= 1e-10)
   {
      e+=1/a;
      n++;
      a*=n;
   }
   printf("e=%.15lf",e);
   return 0;
}
```

运行结果为：



```
e=2.718281828446759
Process returned 0 (0x0)   execution time : 4.182 s
Press any key to continue.
```

正确了！

常见的反馈信息

- **Compile Error**：编译出错，源代码中有语法错误，比如使用某些函数需要的头文件没有包含。
- **Run Time Error**：程序运行时发生错误，多为数组访问越界。
- **Time Limit Exceeded**：超时错误，程序运行时间超过运行时间，比如陷入死循环，算法不够高效等等。
- **Wrong Answer**：答案错误，若通过了样例，可能是因为没有更多的可能情况，导致某些数据通不过。
- **Restricted Function**：使用某些受限的函数，比如重定向、文件操作函数等。
- **Presentation Error**：输出格式错误，可能程序输出中多（或少）输出了空格，回车符等。
- **Accepted**：恭喜，通过