

# Boost 安装 (Windows x86)

## 一、安装过程

1. 下载 boost 安装包, 如 (boost\_1\_83\_0.7z), 解压到本地目录, 如 D:/boost-1.83.0
2. 运行 D:/boost-1.83.0/bootstrap.bat, 会在同目录下生成 b2.exe
3. .\b2.exe toolset=msvc-14.3 install -prefix=D:/boost-1.83.0 link=static

上面命令生成的库为静态库, 包括 release、debug 版本, 64 位、32 位版本, 支持多线程, 使用的编译器为 msvc14.3 (对应 vs2022), 足以满足需要了。安装位置为 e:/dev/boost。若不指定 -prefix=D:/boost-1.83.0, 则安装地址默认为 c:/boost。

一般以静态库方式使用 boost。

## 二、安装测试

### 1. CMakeList 测试环境构建

```
cmake_minimum_required( VERSION 3.5.1)
project(BOOST)

set(CMAKE_PREFIX_PATH D:/boost-1.83.0/boost-1.83.0)

find_package(Boost COMPONENTS thread REQUIRED)
INCLUDE_DIRECTORIES( ${Boost_INCLUDE_DIR} )
LINK_DIRECTORIES( ${Boost_LIBRARY_DIRS} )
set( Boost_USE_STATIC_LIBS OFF )
set( Boost_USE_MULTITHREADED ON )
set( Boost_USE_STATIC_RUNTIME OFF )
set( BOOST_ALL_DYN_LINK ON )

add_executable( ${PROJECT_NAME} HelloThreader.cpp )
target_link_libraries( ${PROJECT_NAME} ${Boost_LIBRARIES})
```

## 2. HelloThreader.cpp 测试线程模块是否正确引入

```
#include <iostream>
#include <boost/thread/thread.hpp>

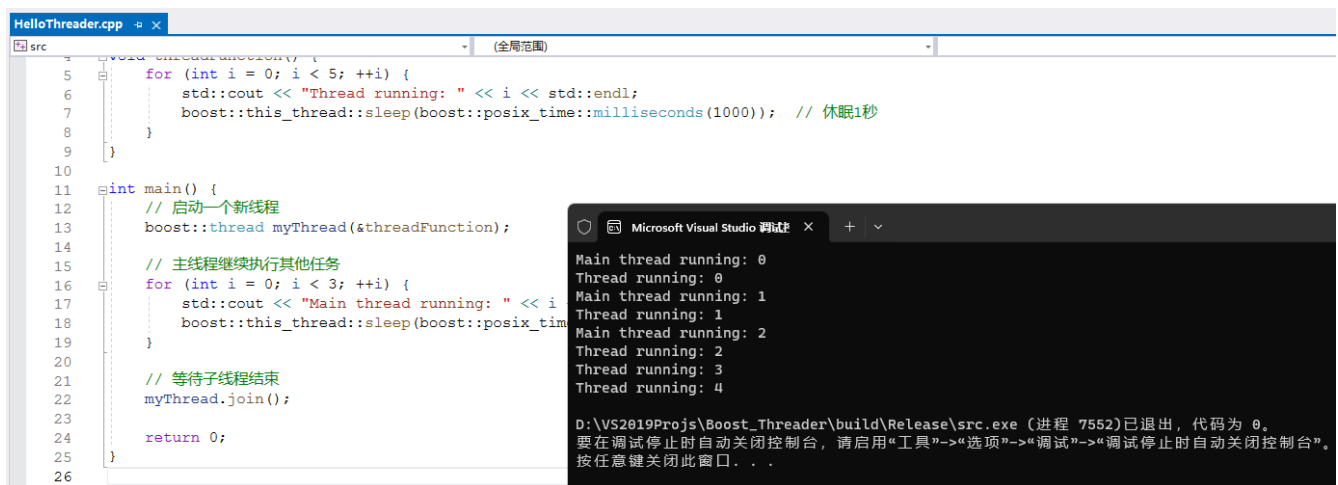
void threadFunction() {
    for (int i = 0; i < 5; ++i) {
        std::cout << "Thread running: " << i << std::endl;
        boost::this_thread::sleep(boost::posix_time::milliseconds(1000));
        // 休眠1秒
    }
}

int main() {
    // 启动一个新线程
    boost::thread myThread(&threadFunction);

    // 主线程继续执行其他任务
    for (int i = 0; i < 3; ++i) {
        std::cout << "Main thread running: " << i << std::endl;
        boost::this_thread::sleep(boost::posix_time::milliseconds(500));
        // 休眠0.5秒
    }

    // 等待子线程结束
    myThread.join();

    return 0;
}
```



The screenshot shows the Visual Studio IDE with the file `HelloThreader.cpp` open. The code is as follows:

```
1  #include <iostream>
2  #include <boost/thread/thread.hpp>
3
4  void threadFunction() {
5      for (int i = 0; i < 5; ++i) {
6          std::cout << "Thread running: " << i << std::endl;
7          boost::this_thread::sleep(boost::posix_time::milliseconds(1000)); // 休眠1秒
8      }
9  }
10
11 int main() {
12     // 启动一个新线程
13     boost::thread myThread(&threadFunction);
14
15     // 主线程继续执行其他任务
16     for (int i = 0; i < 3; ++i) {
17         std::cout << "Main thread running: " << i << std::endl;
18         boost::this_thread::sleep(boost::posix_time::milliseconds(500)); // 休眠0.5秒
19     }
20
21     // 等待子线程结束
22     myThread.join();
23
24     return 0;
25 }
26
```

The output window on the right shows the following output:

```
Microsoft Visual Studio 调试
Main thread running: 0
Thread running: 0
Main thread running: 1
Thread running: 1
Main thread running: 2
Thread running: 2
Thread running: 3
Thread running: 4
D:\VS2019Projs\Boost_Threder\build\Release\src.exe (进程 7552)已退出, 代码为 0。
要在调试停止时自动关闭控制台, 请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口...
```

(a) 正常编译运行测试结果