

GATS Companion to Installing BOOST

Author: Garth Santor

Editors: Trinh Hân

Copyright Dates: 2009-20

Version: 1.0.0 (2020-09-09)

Overview

How to install BOOST for use within Visual Studio 2019.

boost version: 1.74.0

Visual Studio version: 16.7.4

Acquiring the files

Go to the website <https://www.boost.org/> and click the **download** link on the right side of the window.

Companions - GATS x Boost C++ Libraries x +

← → ↺ 🏠 🔒 <https://www.boost.org> ☆ b | ☆ 📁 👤 ...

boost
C++ LIBRARIES

"...one of the most highly regarded and expertly designed C++ library projects in the world."
— Herb Sutter and Andrei Alexandrescu, C++ Coding Standards

WELCOME TO BOOST.ORG!

Boost provides free peer-reviewed portable C++ source libraries.

We emphasize libraries that work well with the C++ Standard Library. Boost libraries are intended to be widely useful, and usable across a broad spectrum of applications. The [Boost license](#) encourages the use of Boost libraries for all users with minimal restrictions.

We aim to establish "existing practice" and provide reference implementations so that Boost libraries are suitable for eventual standardization. Beginning with the ten Boost Libraries included in the Library Technical Report (TR1) and continuing with every release of the ISO standard for C++ since 2011, the C++ Standards Committee has continued to rely on Boost as a valuable source for additions to the Standard C++ Library.

BOOST AS AN ORGANIZATION

The Boost organization and wider Boost community supports research and education into the best possible uses of C++ and libraries developed for it, particularly, but not exclusively those contained in the Boost Library.

The organization and community support mailing lists and chat rooms to educate about best practices and cutting edge techniques for the user of Boost Libraries and C++ in general.

Since 2006 an intimate week long annual conference related to Boost called C++ Now has been held in Aspen, Colorado each May. The conference is an educational opportunity focused on cutting-edge C++. Boost has been a participant in the annual Google Summer of Code since 2007, in which students develop their skills by working on Boost Library development.

GETTING STARTED

Boost works on almost any modern operating system, including UNIX and Windows variants. Follow the [Getting Started Guide](#) to download and install Boost. Popular Linux and Unix distributions such as [Fedora](#), [Debian](#), and [NetBSD](#) include pre-built Boost packages. Boost may also already be available on your organization's internal web server.

DOWNLOADS

CURRENT RELEASE

- Version 1.74.0
[Release Notes](#) | [Download](#) | [Documentation](#)
August 14th, 2020 05:02 GMT

[More Downloads... \(RSS\)](#)

NEWS

- Version 1.74.0
New Libraries: STLInterfaces. Updated Libraries: Asio, Atomic, Beast, Bimap, Config, Core, DLL, Endian, Filesystem, Flyweight, Format, Geometry, GIL, Heap, Integer, Iterator, LexicalCast, Log, Mp11, MultIndex, Nowide, Outcome, PolyCollection, Polygon, SmartPtr, Stacktrace, System, TypeTraits, Variant, Variant2, Wave, YAP.
August 14th, 2020 05:02 GMT
- Version 1.73.0
New Libraries: Nowide, StaticString. Updated Libraries: Align, Any, Asio, Assert, Atomic, Beast, Context, Conversion, DateTime, DLL, DynamicBitset, Flyweight, Geometry, GIL, Histogram, ICL, IO, LexicalCast, Log, Math, Mp11, MultIndex, Multiprecision, Outcome, PolyCollection, Stacktrace, Test, ThrowException, TTI, TypeIndex, Utility, Variant, Variant2, WinAPI.
April 28th, 2020 03:57 GMT

GET BOOST

ENHANCED BY Gc 🔍

Donate

VISA MASTERCARD AMERICAN EXPRESS DISCOVER

WELCOME

- [Getting Started](#)
- [Download](#)
- [Libraries](#)
- [Mailing Lists](#)
- [Reporting and Fixing Bugs](#)
- [Wiki](#)

[INTRODUCTION](#)

[COMMUNITY](#)

[DEVELOPMENT](#)

[DOCUMENTATION](#)

Pick the *prebuilt windows binaries* link.

<https://www.boost.org/users/download/>

Companions – GATS

Boost Downloads

+

←

→

↺

🏠

🔒

https://www.boost.org/users/download/

☆


b

⌵

🔍

👤

⋮



“...one of the most highly regarded and expertly designed C++ library projects in the world.”
— Herb Sutter and Andrei Alexandrescu, C++ Coding Standards

BOOST DOWNLOADS

[Current Release](#)
[Old Boost Releases](#)
[Git Repositories](#)

CURRENT RELEASE

VERSION 1.74.0
August 14th, 2020 05:02 GMT

New Libraries: STLInterfaces. Updated Libraries: Asio, Atomic, Beast, Bimap, Config, Core, DLL, Endian, Filesystem, Flyweight, Format, Geometry, GIL, Heap, Integer, Iterator, LexicalCast, Log, Mp11, MultiIndex, Nowide, Outcome, PolyCollection, Polygon, SmartPtr, Stacktrace, System, TypeTraits, Variant, Variant2, Wave, YAP.

[Release Notes](#) [Download](#) [Documentation](#)

DOWNLOADS

Platform	File	SHA256 Hash
unix	boost_1_74_0.tar.bz2	83bfc1507731a0906e387fc28b7ef5417d591429e51e788417fe9ff025e116b1
	boost_1_74_0.tar.gz	afff36d392885120bcac079148c177d1f6f7730ec3d47233aa51b0afa4db94a5
windows	boost_1_74_0.7z	903fd40074de8dc733d2f0b6b0642e43b714653eae79289b12161b9567fae97
	boost_1_74_0.zip	a0e7ce67c52d816708fdeccdd8c9725626ba61254c13c18770498cacd514710a


OTHER DOWNLOADS


As well as the official releases, members of the community provide other downloads for your convenience. Since these are produced separately, there might be a delay before they're available after a release and they're not be available for all releases.

- Prebuilt windows binaries.
- PDF documentation (only for BoostBook based documentation).





OLD BOOST RELEASES

Old versions of boost can be found on the [version history](#) page or from the [sourceforge download](#) page.



ENHANCED BY Gc 

Donate



WELCOME

INTRODUCTION

[Download](#)

[News](#)

[Version History](#)

[License](#)

[People](#)

[Bibliography](#)

[Who's Using Boost](#)

[FAQ](#)

[How Boost Started](#)

[Index](#)

COMMUNITY

DEVELOPMENT

DOCUMENTATION

You'll now be on SourceForge. Pick the latest 1.74.0 version link.

<https://sourceforge.net/projects/boost/files/boost-binaries/>

The screenshot shows the SourceForge website for Boost C++ Libraries. The page has a dark header with the SourceForge logo and navigation links. Below the header, there are several advertisements, including one for Nest Hello Video Doorbell and another for Adidas Women's Grand Court Sneakers. The main content area is titled "Boost C++ Libraries" and includes a description: "Free peer-reviewed portable C++ source libraries". It lists contributors: beman_dawes, danieljames, eric_niebler, mclow, and vladimir_prus. There are tabs for Summary, Files, Reviews, Support, Wiki, Mailing Lists, News, Discussion, and Code. The "Files" tab is active, showing a list of versions. A green button "Download Latest Version" is visible, along with a "Get Updates" button and a RSS feed icon. The version list table is as follows:

Name	Modified	Size	Downloads / Week
Parent folder			
1.74.0	2020-08-12		2,901
1.74.0_b1	2020-07-16		32
1.73.0	2020-04-26		177
1.73.0_b1	2020-04-14		4
1.72.0	2019-12-11		595
1.72.0_b1	2019-11-20		4

On the right side, there are more advertisements, including one for LEGO City Exclusive and another for OpenCV. A "Recommended Projects" section lists OpenCV. At the bottom right, there is a "Get latest updates about Open Source Projects, Conferences and News" section with a "Sign Up" button.

There are two binaries you'll need to install, a 32-bit version, and a 64-bit version. You'll also have to select the release for the current version of Visual Studio 2019. The currently updated version of Visual Studio 2019 is compiler is 16.7.3, but the binary format is 14.2, so that's the version we will use.

<https://sourceforge.net/projects/boost/files/boost-binaries/1.74.0/>

Download both:


- boost_1_74_0-msvc_14.2-32.exe
- boost_1_74_0-msvc_14.2-64.exe

Companions – GATS
Boost C++ Libraries - Browse /b
https://sourceforge.net/projects/boost/files/boost-binaries/1.74.0/

SOURCEFORGE
Help
Create
Join
Login
Open Source Software
Business Software
Resources
Search for software or solutions

BEST BUY
Let's get back to it for Back to School.
Learn More

Home / Browse / Development / Software Development / Boost C++ Libraries / Files



Boost C++ Libraries

Free peer-reviewed portable C++ source libraries
Brought to you by: beman_dawes, danieljames, eric_niebler, mclow, vladimir_prus


Summary
Files
Reviews
Support
Wiki
Mailing Lists
News
Discussion
Code

Download Latest Version
boost_1_66_0-7z (74.1 MB)
Get Updates

Home / boost-binaries / 1.74.0

Name	Modified	Size	Downloads / Week
Parent folder			
SHA256SUMS.asc	2020-08-12	2.4 kB	0
SHA256SUMS	2020-08-12	1.5 kB	1
boost_1_74_0-msvc-14.1-64.exe	2020-08-11	183.1 MB	226
boost_1_74_0-msvc-14.0-64.exe	2020-08-11	185.0 MB	1,089
boost_1_74_0-msvc-14.2-64.exe	2020-08-11	170.5 MB	220
boost_1_74_0-msvc-12.0-64.exe	2020-08-11	176.8 MB	7
boost_1_74_0-msvc-14.1-32.exe	2020-08-11	168.6 MB	253
boost_1_74_0-msvc-11.0-32.exe	2020-08-11	169.4 MB	0
boost_1_74_0-msvc-10.0-64.exe	2020-08-11	176.0 MB	17
boost_1_74_0-msvc-12.0-32.exe	2020-08-11	160.2 MB	7
boost_1_74_0-msvc-14.0-32.exe	2020-08-11	169.8 MB	16
boost_1_74_0-msvc-11.0-64.exe	2020-08-11	186.7 MB	0
boost_1_74_0-msvc-10.0-32.exe	2020-08-11	159.0 MB	13
boost_1_74_0-msvc-14.2-32.exe	2020-08-11	158.2 MB	375
boost_1_74_0-bin-msvc-all-32-64.7z	2020-08-11	1.2 GB	662
boost_1_74_0-64bitlog.txt	2020-08-11	29.0 kB	3

Explore The Haunted House



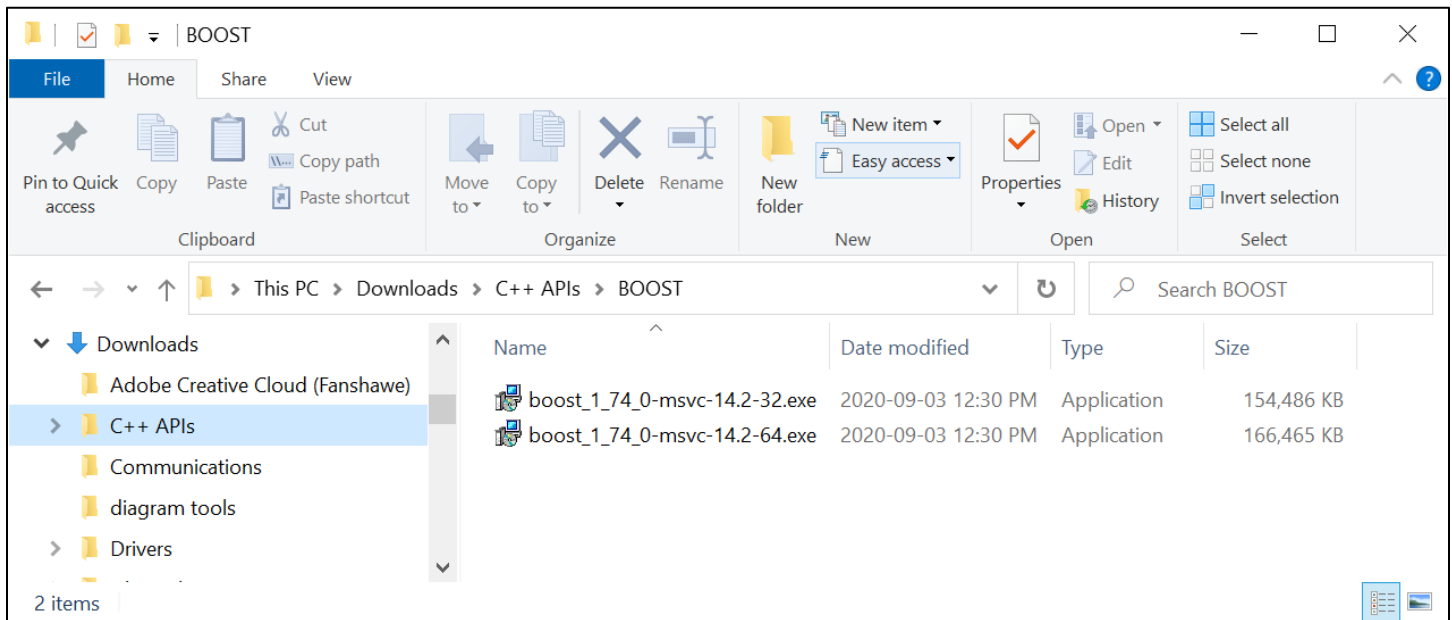
Discover the thrills and surprises packed into this spooky building project! Buy Now.

Get latest updates about Open Source Projects, Conferences and News.

Sign Up

No, Thank you

You should now have the two new files in your download folder...



Install the libraries

The two files you downloaded are self-extracting archives. You'll need about 4GiB of free memory (ideally on drive C). There are two reasons drive C is the best choice:

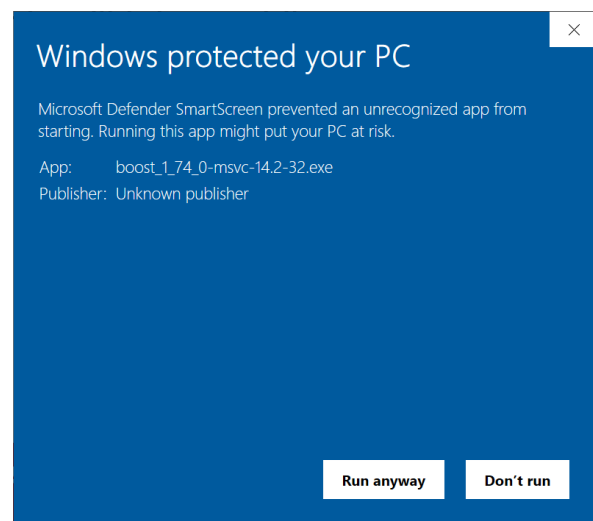
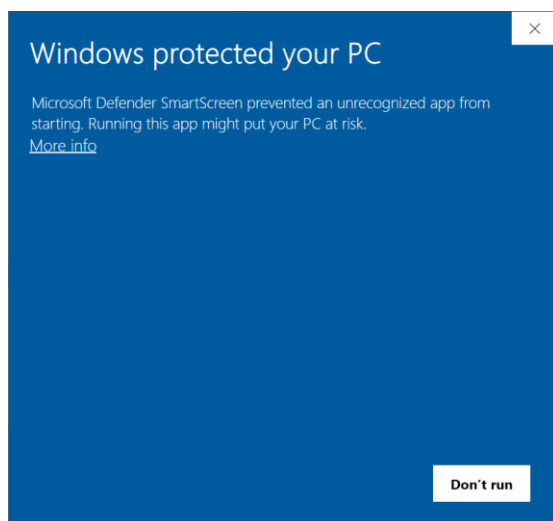
Compatibility: System configuration and portability will be better (hopefully seamless) if we all use the same location.

Performance: Many of us will have a solid-state drive (SSD) as their primary drive. I have a ½ TiB SSD for my primary drive (drive C:) and a 2 TiB HDD for my secondary drive (drive D:). Since boost has a huge number of small header files (14,184 in version 1.73.0) the lower latency and higher transfer rate of an SSD will provide a noticeable improvement when compiling.

You'll install each library (one after the other) as many of the files will be overwritten by the other. The order you install them doesn't matter. Double click either archive and follow the steps below.

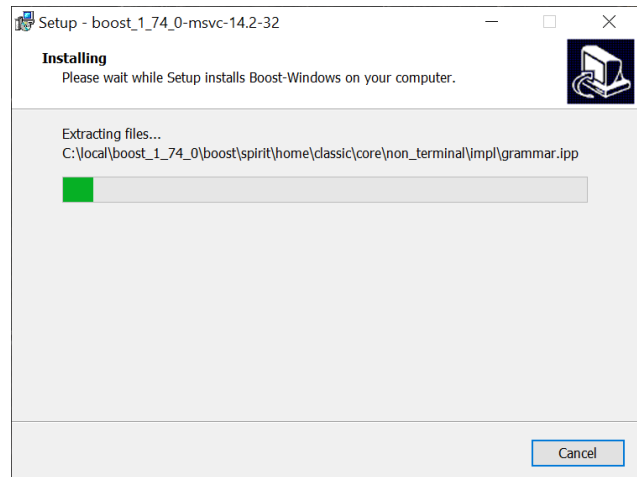
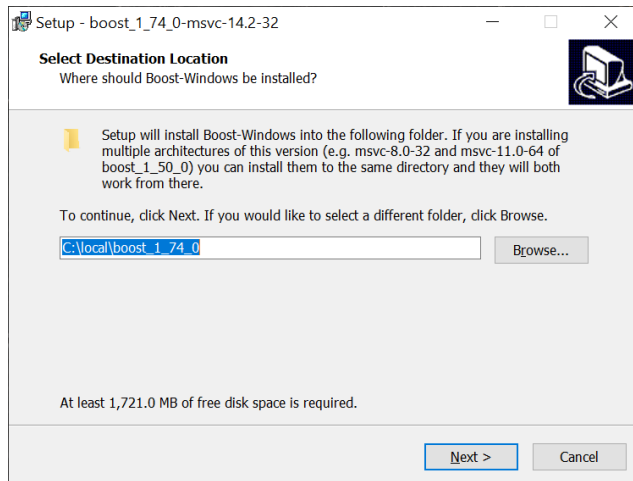
You'll probably see this as soon as you run the installer.

Click [More info](#), you should see:



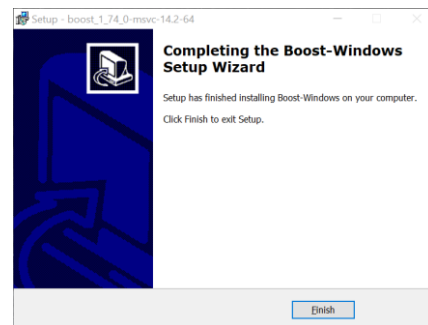
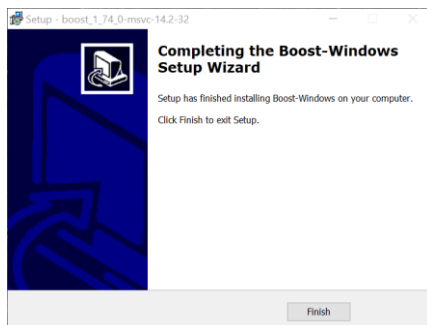
Click **Run anyway**.

It is best to use the default folder (C: drive), so just click **Next >**.



Installation could take several minutes, depending on the speed of your drive, RAM, and CPU. My desktop installs boost in about 1 minute, but I've seen some system take more than 20 minutes to install.

When complete, do the same for the 64-bit version.



Creating Property Pages for Visual C++

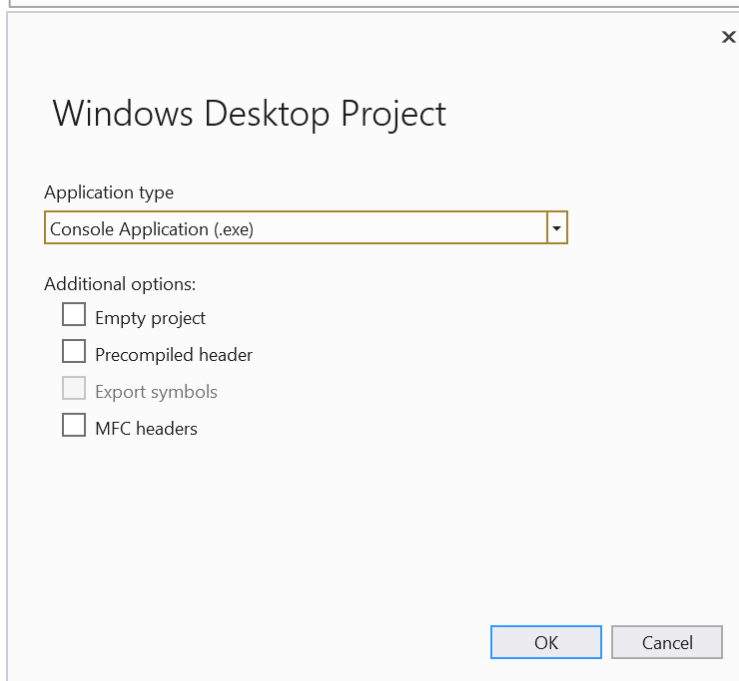
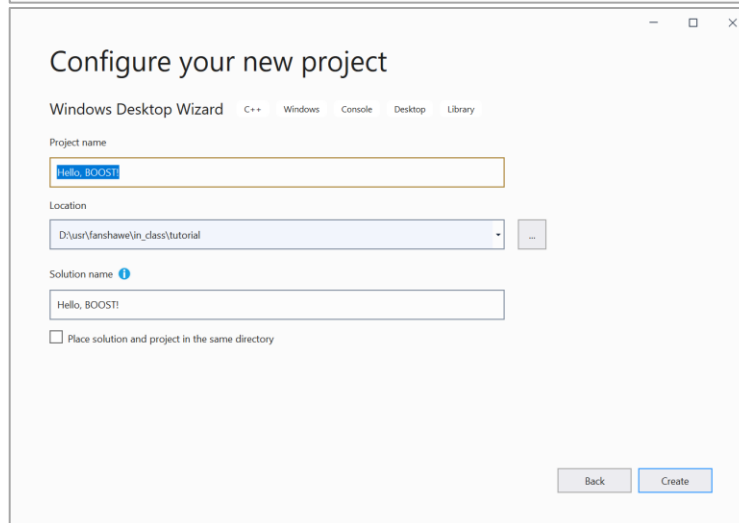
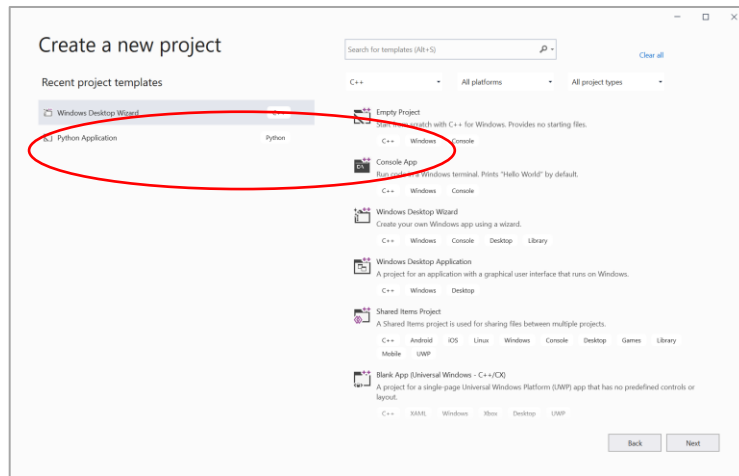
Visual Studio 2017 and earlier had global property pages, but they were removed in Visual Studio 2019 (they haven't said why...)

We'll need a new C++ project loaded into Visual Studio 2019. We'll create a *Hello, BOOST!* application to test our configuration.

Use the **Windows Desktop Wizard** to create a hello world desktop application.

Name the project...

Verify **Console Application (.exe)** as the application type, no additional options need to be checked.



Ignoring the comments your source code should appear as:

```
#include <iostream>

int main() {
    std::cout << "Hello World!\n";
}
```

Modify the code to use a *header-only* boost library.

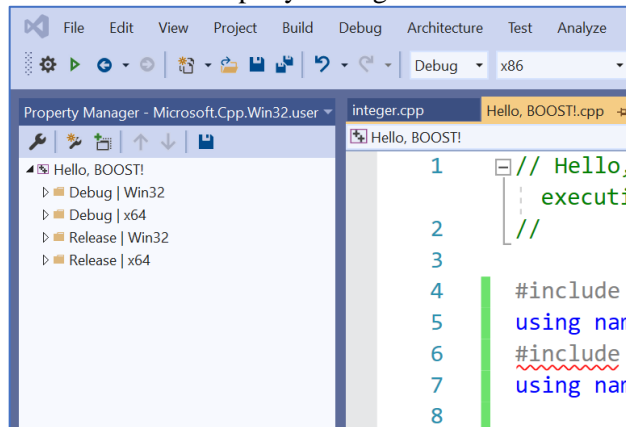
```
#include <iostream>
using namespace std;
#include <boost/multiprecision/cpp_int.hpp>
using namespace boost::multiprecision;

int main() {
    cout << "Hello World!\n";
    cpp_int big("1234567890123456789012345678901234567890");
}
```

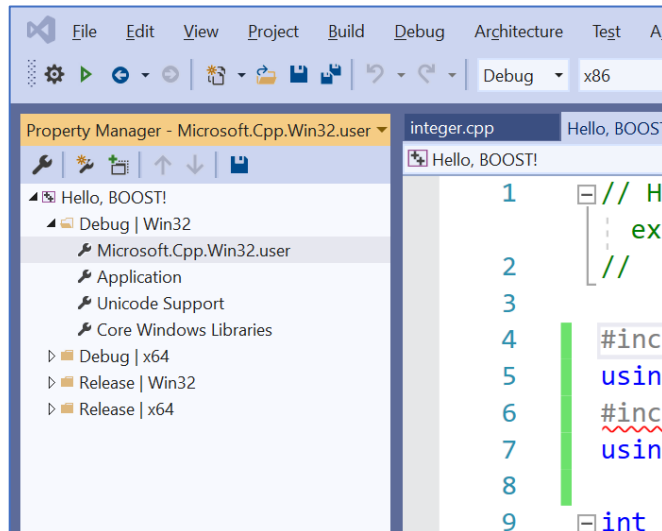
Add a new *reusable* project property sheet (32-bit)

Open the Property Manager

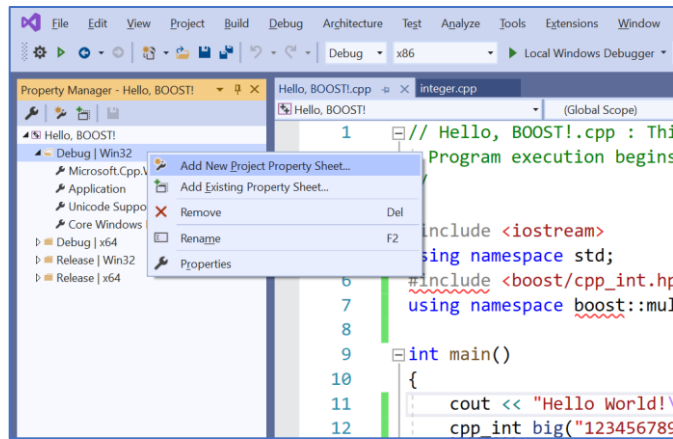
Menu: View → Property Manager



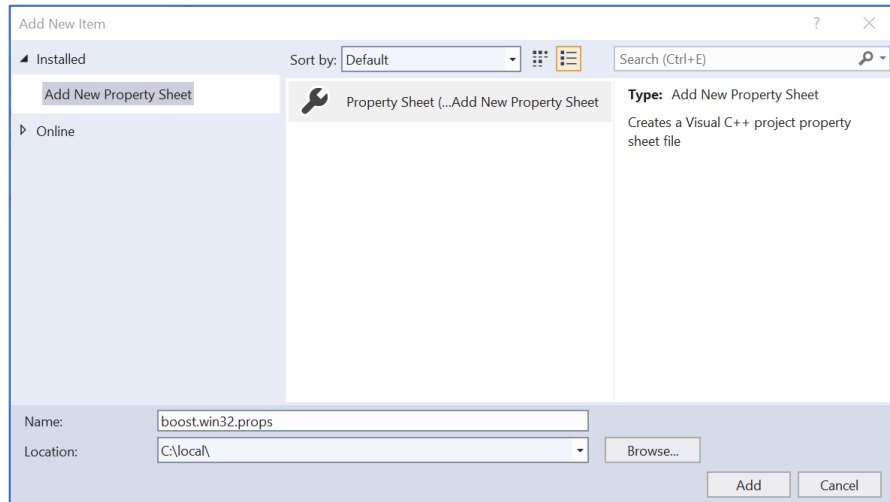
Expand the Debug | Win32 folder



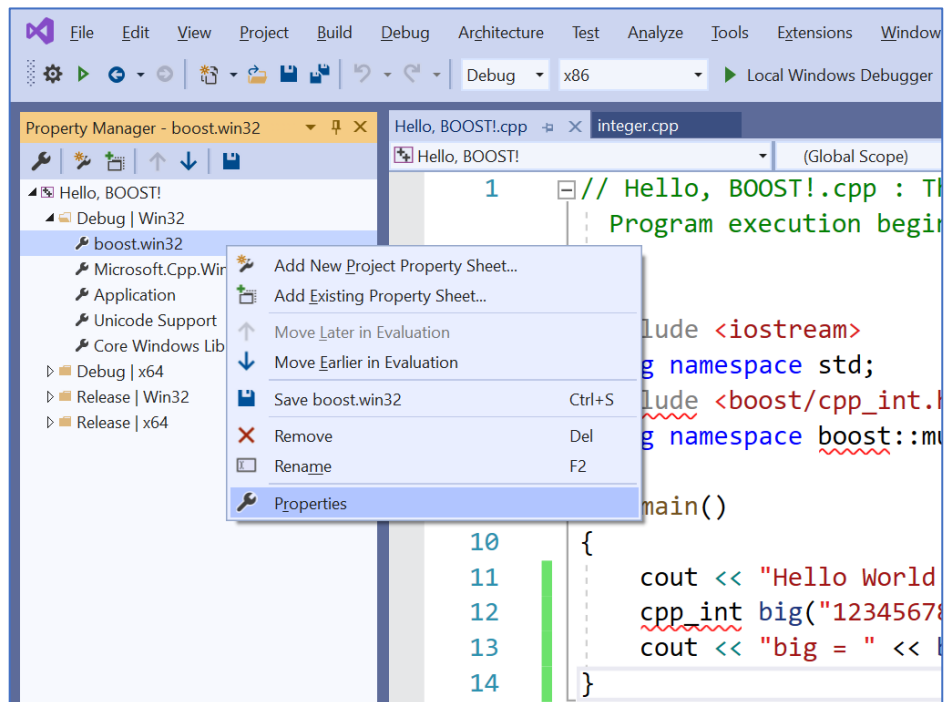
Add a new project property sheet



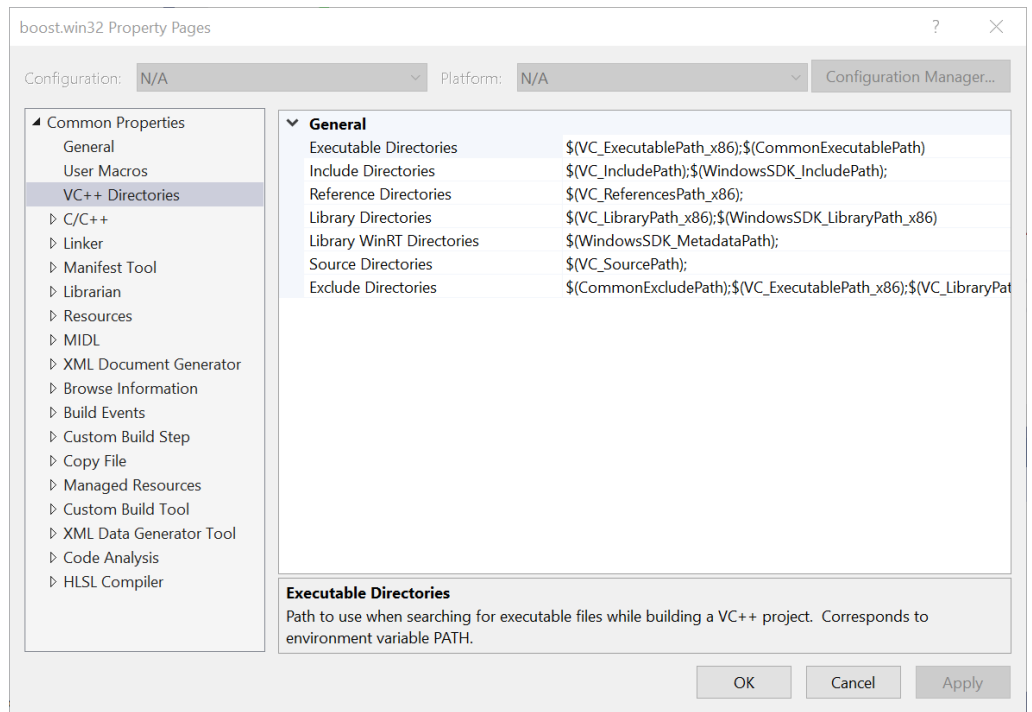
Name the file **boost.win32.props** and save the property sheet to the boost installation folder **C:\local**.



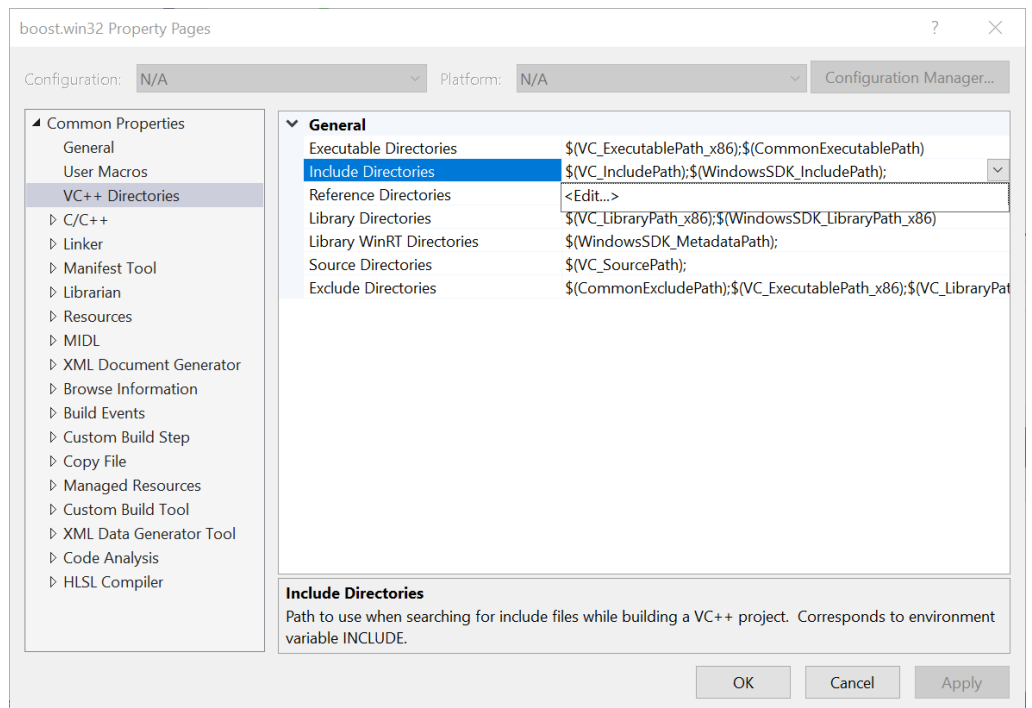
Open the property sheet.



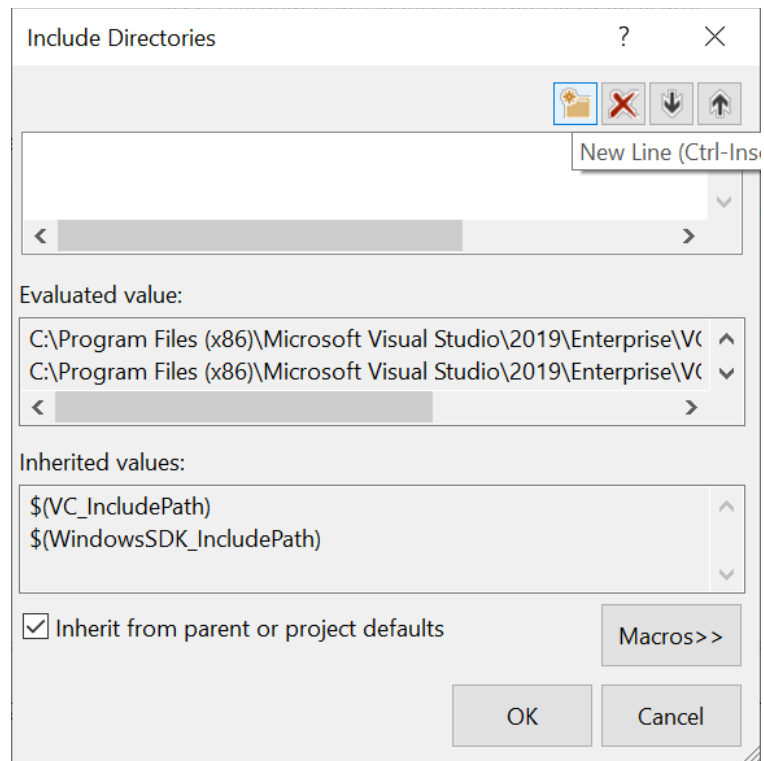
Select the **VC++ Directories** tab.



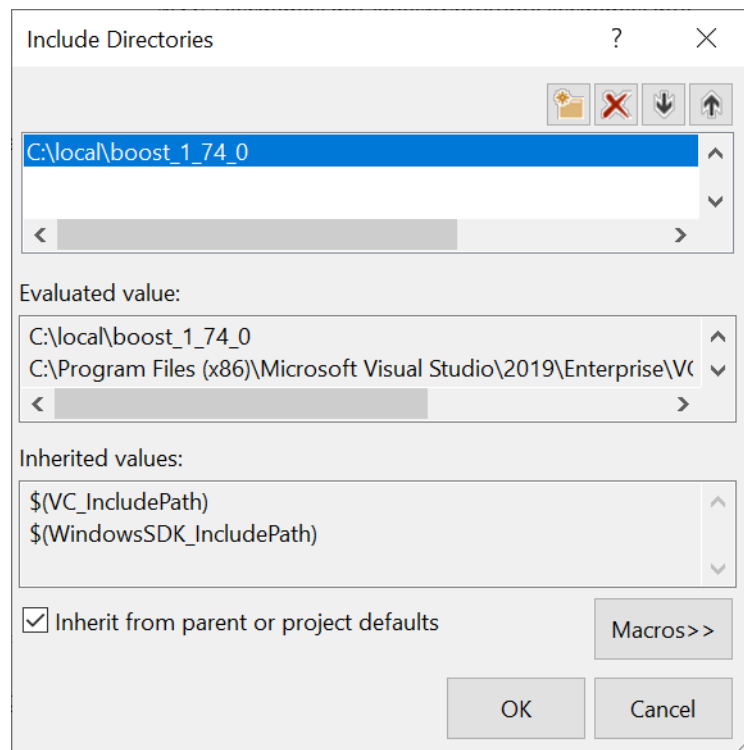
Edit the include paths.



Add a new line...

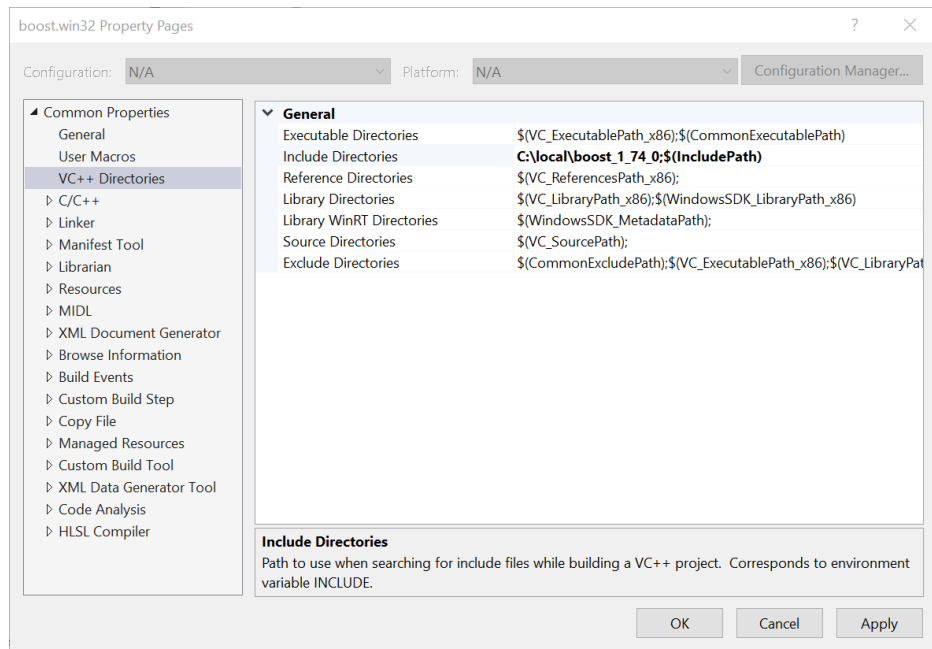


Add the boost main folder



You should see...

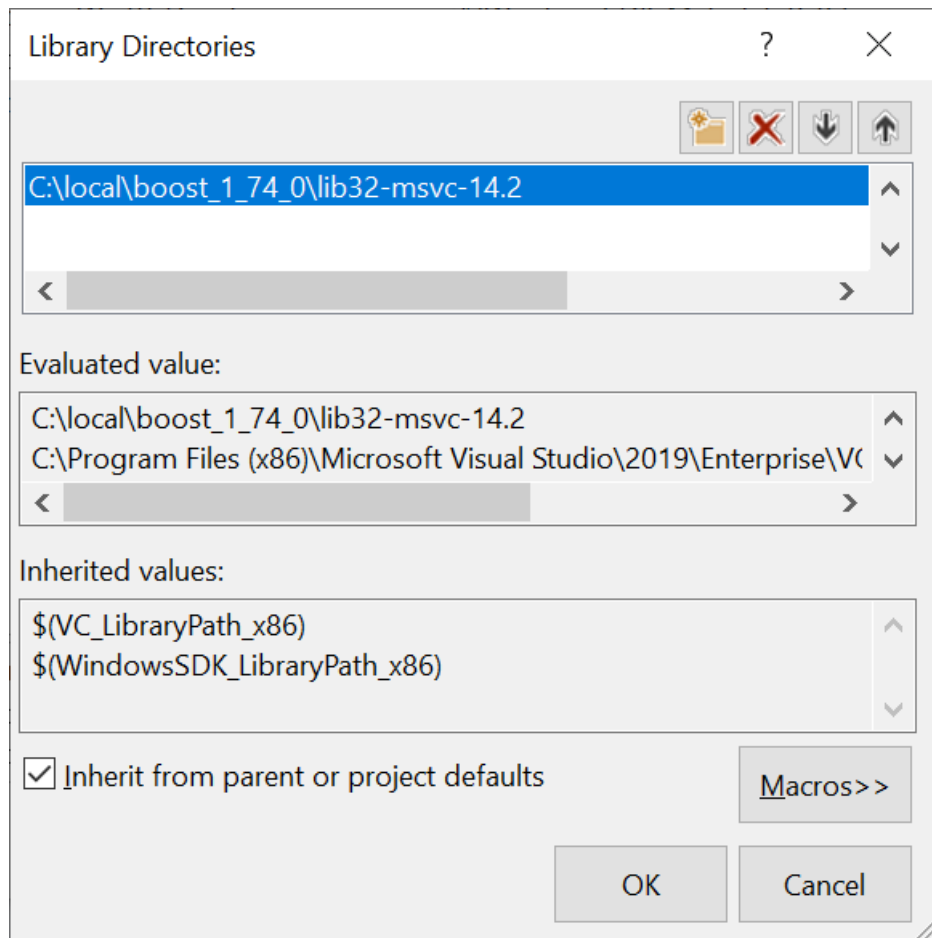
The \$(IncludePath) must be there as it represents the original path from the default configuration.



Add to the library path.

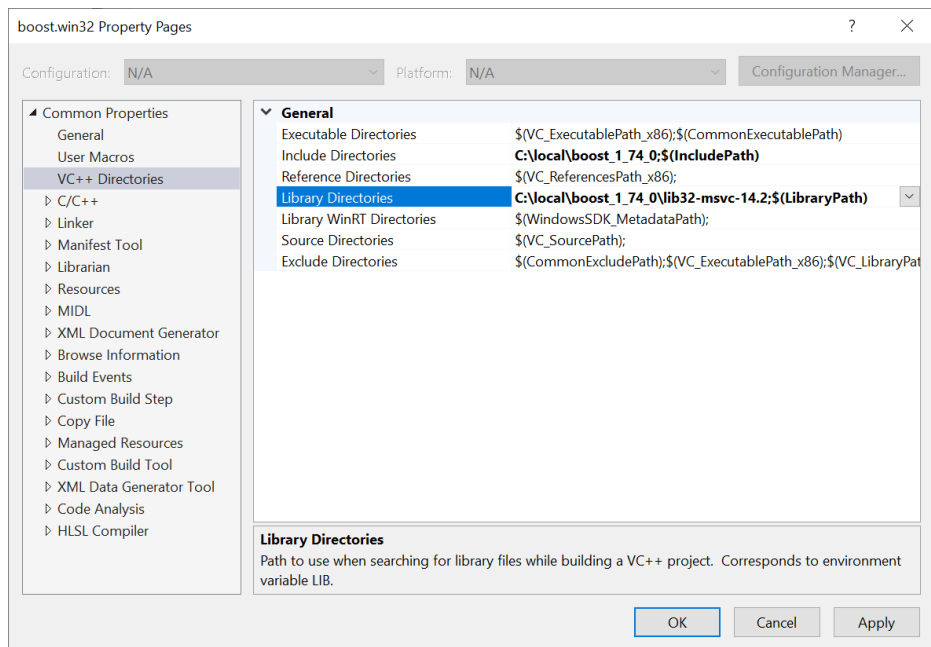
Note that we must specify which of the two libraries we want.

In this case we want the Win32 library, so we pick the **lib32-msvc-14.2** folder



You should see...

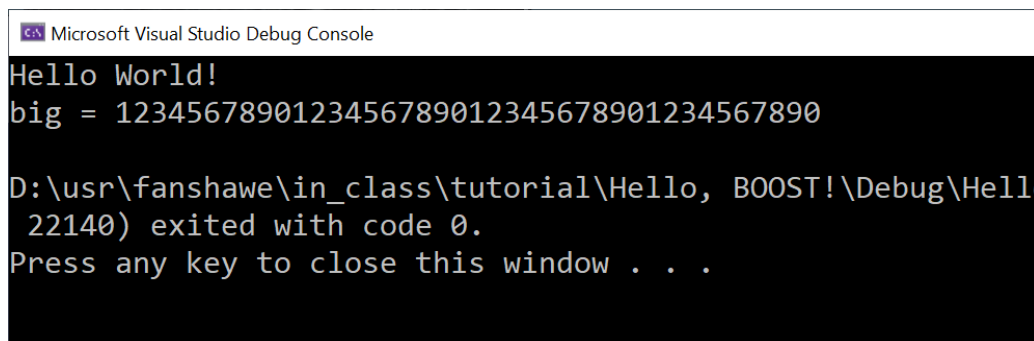
Click OK



Test the configuration...

Run the program.

The errors should disappear and the project should build and execute.

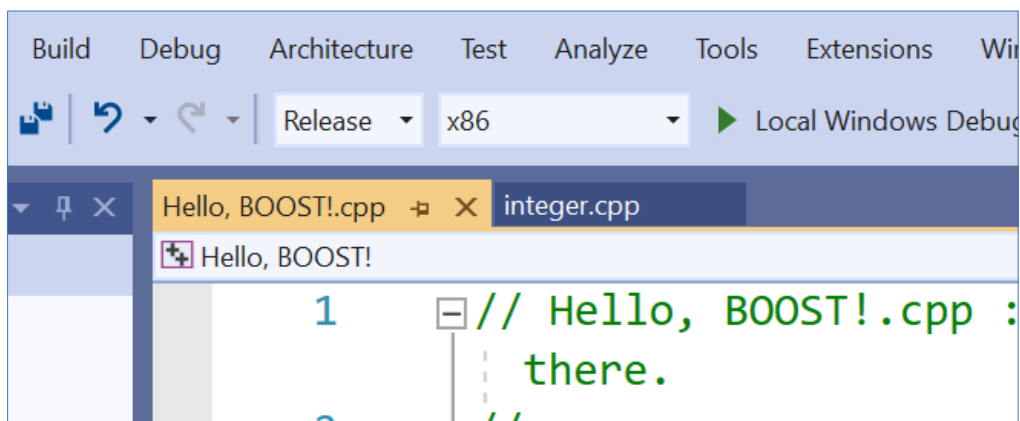


Adding the property sheet to release mode

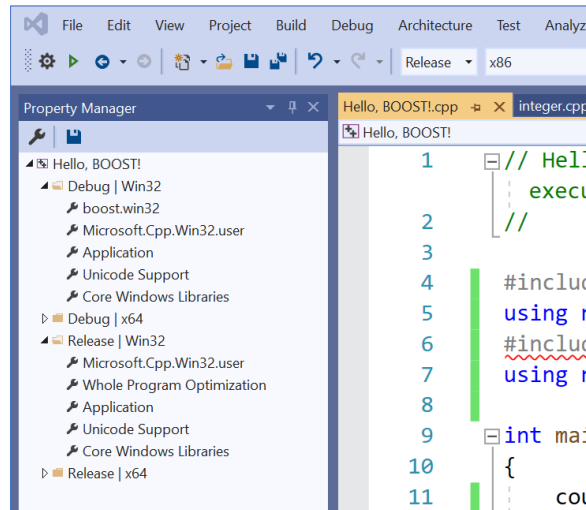
The release build uses the same header files as debug mode, and those header files instruct the linker to use the correct debug/release libraries. So, we can reuse the property page from the debug build in the release build.

Switch to release mode.

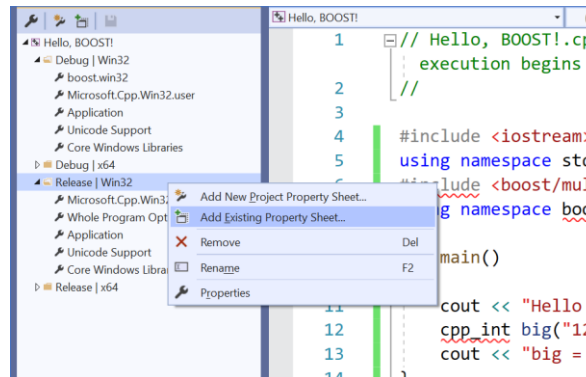
You'll notice that the include directive, the using statement, and the object definition are generating errors again.



Expand the Release | Win32 tab in the Property Manager

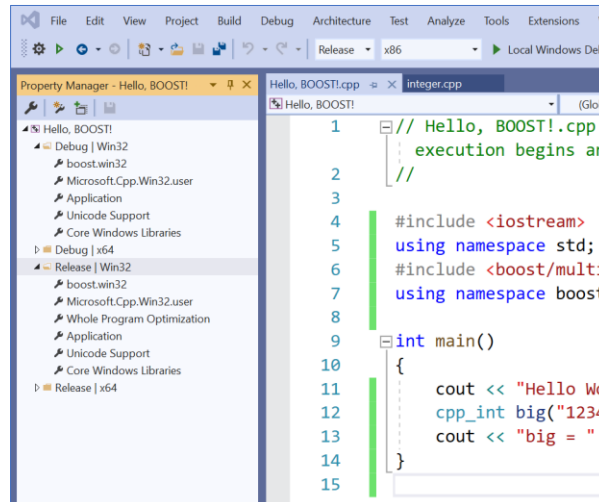


Add the existing property sheet.

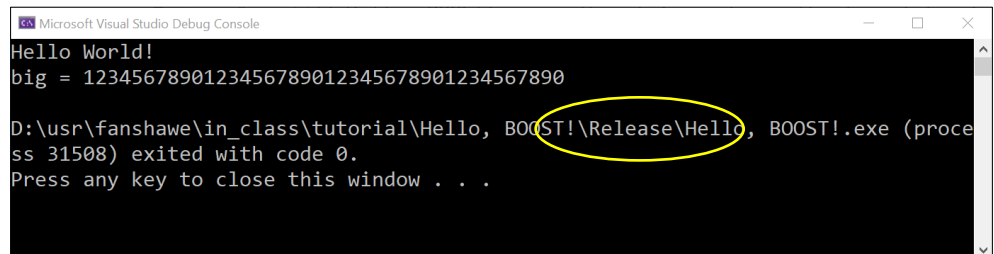


You should see...

And note that the errors have cleared and



Verify the release mode builds



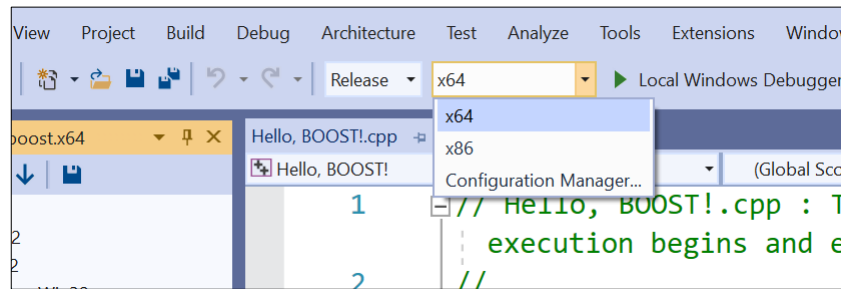
Configure for 64-bit builds

The 64-bit (x64) configurations use the same header files, but different libraries. Libraries specifically built with the 64-bit compiler.

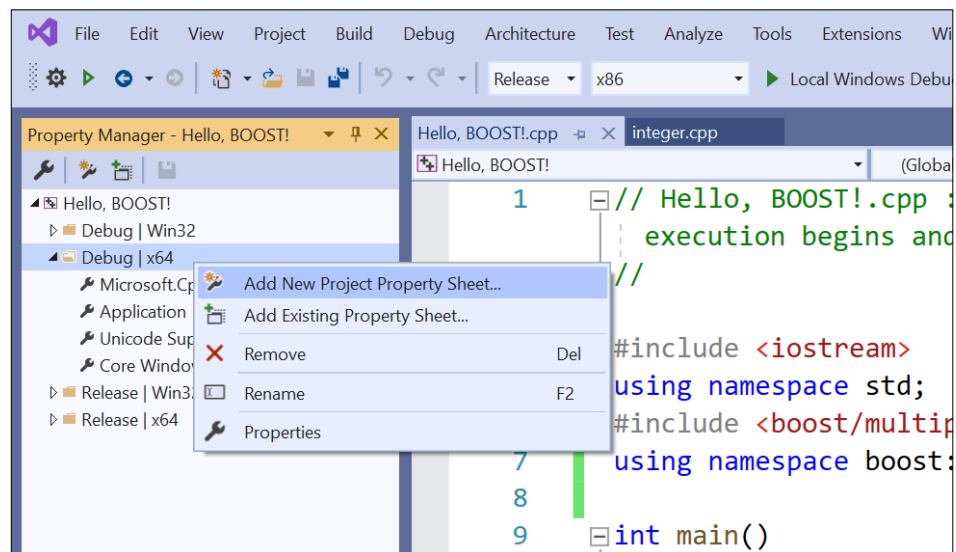
The process will be similar to the x86 installation:

- Create a property sheet in the boost folder
- Set the paths: same include, 64-bit binaries
- Add to both configurations.

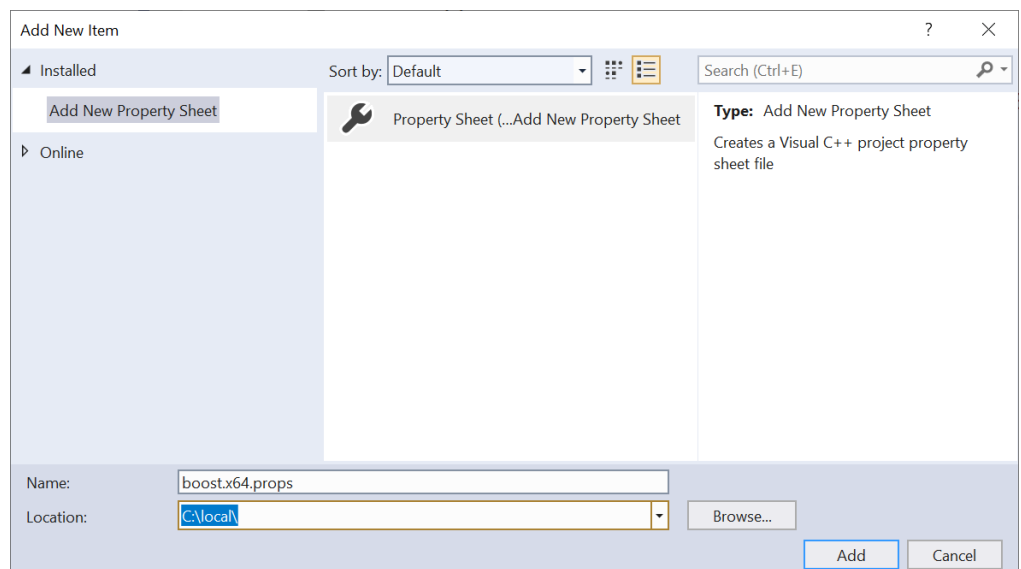
Switch to the 64-bit (x64) build.



Add a new project property sheet to Debug | x64

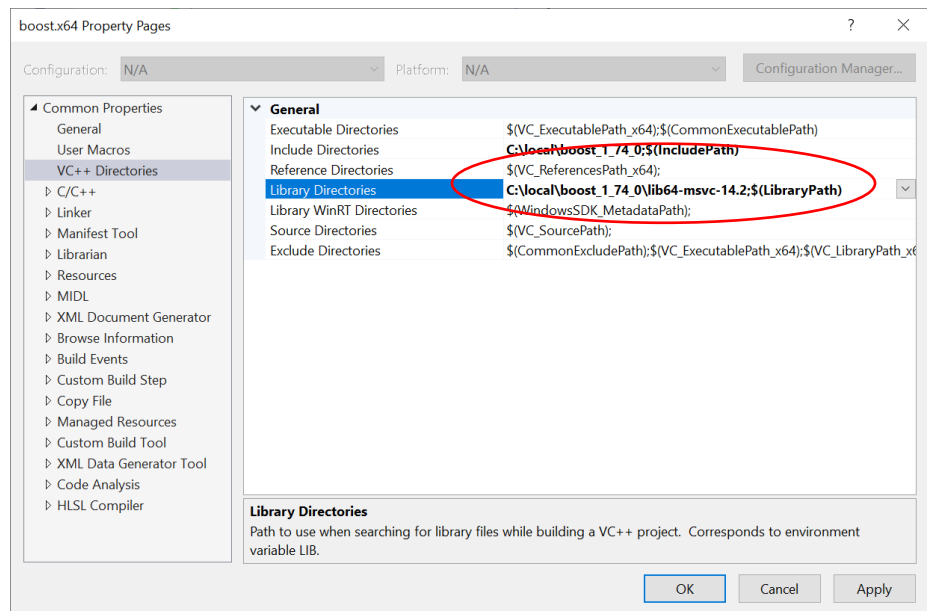


Name the file boost.x64.props and save it in the C:\local folder.



Configure the paths.

Note that the library path points to the 64-bit folder.



Add the file you just created to the release configuration as well.

Save and test both configurations

New projects

For new projects you'll only need to add the existing property sheets to the appropriate project type in the project manager.