



Microsoft C++ 生成工具 Visual Studio Code

Microsoft C++ 生成工具

MinGW + GCC

# 安装 Microsoft C++ 生成工具

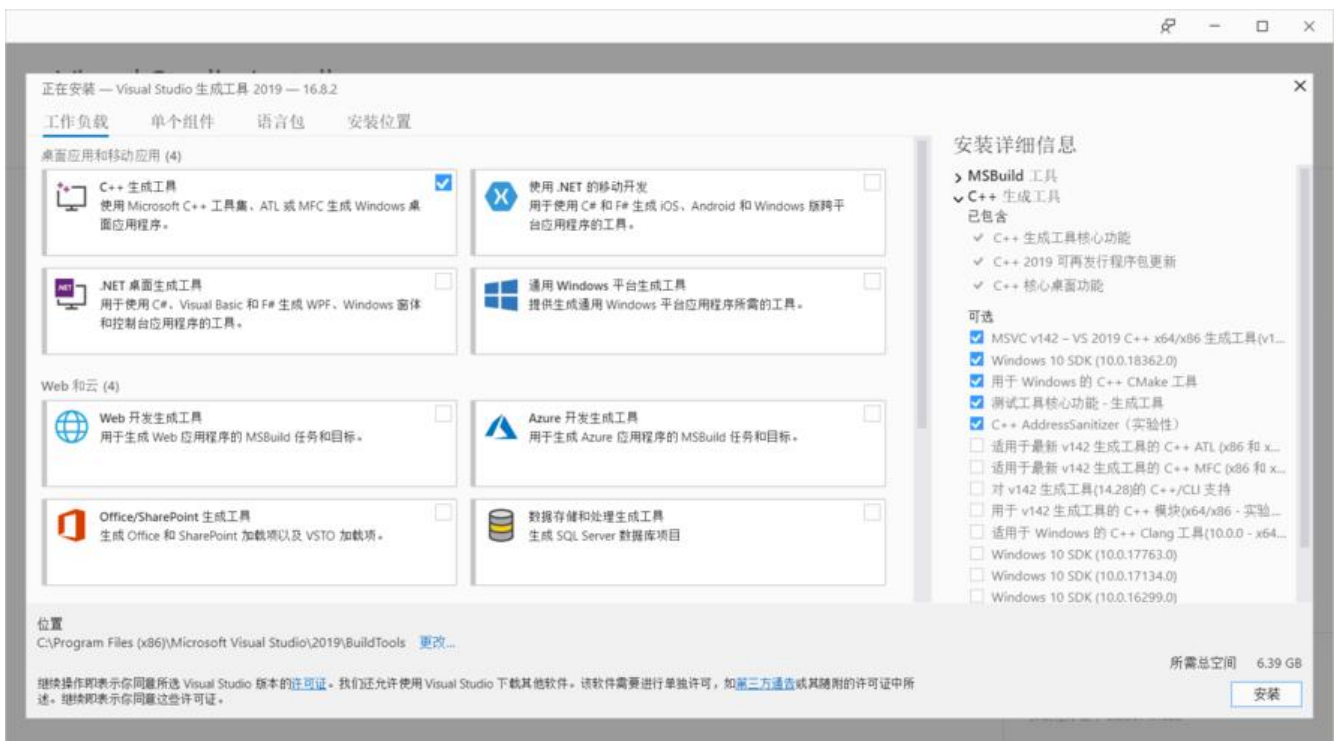
Microsoft C++ 生成工具

生成工具

Microsoft C++ 生成工具

Microsoft C++

Microsoft C++ 生成工具





# 安装rustup

# 安装rustup

```
C:\Users>wangzk\Downloads\rustup-init.exe

Rust Visual C++ prerequisites

Rust requires the Microsoft C++ build tools for Visual Studio 2013 or
later, but they don't seem to be installed.

The easiest way to acquire the build tools is by installing Microsoft
Visual C++ Build Tools 2019 which provides just the Visual C++ build
tools:

  https://visualstudio.microsoft.com/visual-cpp-build-tools/

Please ensure the Windows 10 SDK and the English language pack components
are included when installing the Visual C++ Build Tools.

Alternately, you can install Visual Studio 2019, Visual Studio 2017,
Visual Studio 2015, or Visual Studio 2013 and during install select
the "C++ tools":

  https://visualstudio.microsoft.com/downloads/

Install the C++ build tools before proceeding.

If you will be targeting the GNU ABI or otherwise know what you are
doing then it is fine to continue installation without the build
tools, but otherwise, install the C++ build tools before proceeding.

Continue? (Y/n) _
```

### Microsoft C++ 生成工具

```
C:\Users>lingyundu\Downloads\rustup-init.exe

The Cargo home directory located at:

  C:\Users\lingyundu\.cargo

This can be modified with the CARGO_HOME environment variable.

The cargo, rustc, rustup and other commands will be added to
Cargo's bin directory, located at:

  C:\Users\lingyundu\.cargo\bin

This path will then be added to your PATH environment variable by
modifying the HKEY_CURRENT_USER/Environment/PATH registry key.

You can uninstall at any time with rustup self uninstall and
these changes will be reverted.

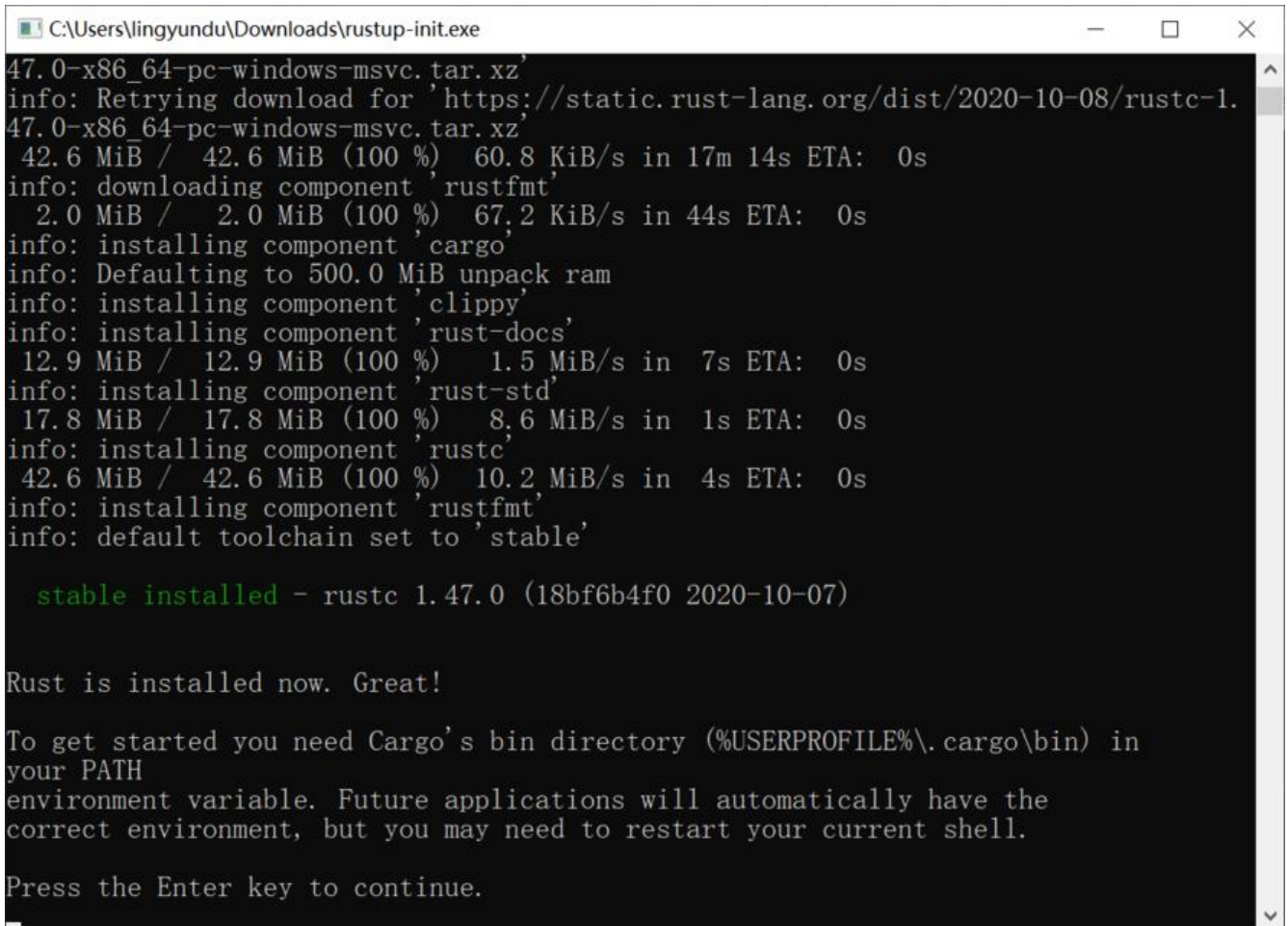
Current installation options:

  default host triple: x86_64-pc-windows-msvc
  default toolchain: stable (default)
  profile: default
  modify PATH variable: yes

1) Proceed with installation (default)
2) Customize installation
3) Cancel installation
>
```

## MinGW + GCC

### 安装成功



```
C:\Users\lingyundu\Downloads\rustup-init.exe
47.0-x86_64-pc-windows-msvc.tar.xz
info: Retrying download for 'https://static.rust-lang.org/dist/2020-10-08/rustc-1.
47.0-x86_64-pc-windows-msvc.tar.xz'
 42.6 MiB /  42.6 MiB (100 %)  60.8 KiB/s in 17m 14s ETA:  0s
info: downloading component 'rustfmt'
   2.0 MiB /   2.0 MiB (100 %)  67.2 KiB/s in 44s ETA:  0s
info: installing component 'cargo'
info: Defaulting to 500.0 MiB unpack ram
info: installing component 'clippy'
info: installing component 'rust-docs'
 12.9 MiB /  12.9 MiB (100 %)   1.5 MiB/s in  7s ETA:  0s
info: installing component 'rust-std'
 17.8 MiB /  17.8 MiB (100 %)   8.6 MiB/s in  1s ETA:  0s
info: installing component 'rustc'
 42.6 MiB /  42.6 MiB (100 %)  10.2 MiB/s in  4s ETA:  0s
info: installing component 'rustfmt'
info: default toolchain set to 'stable'

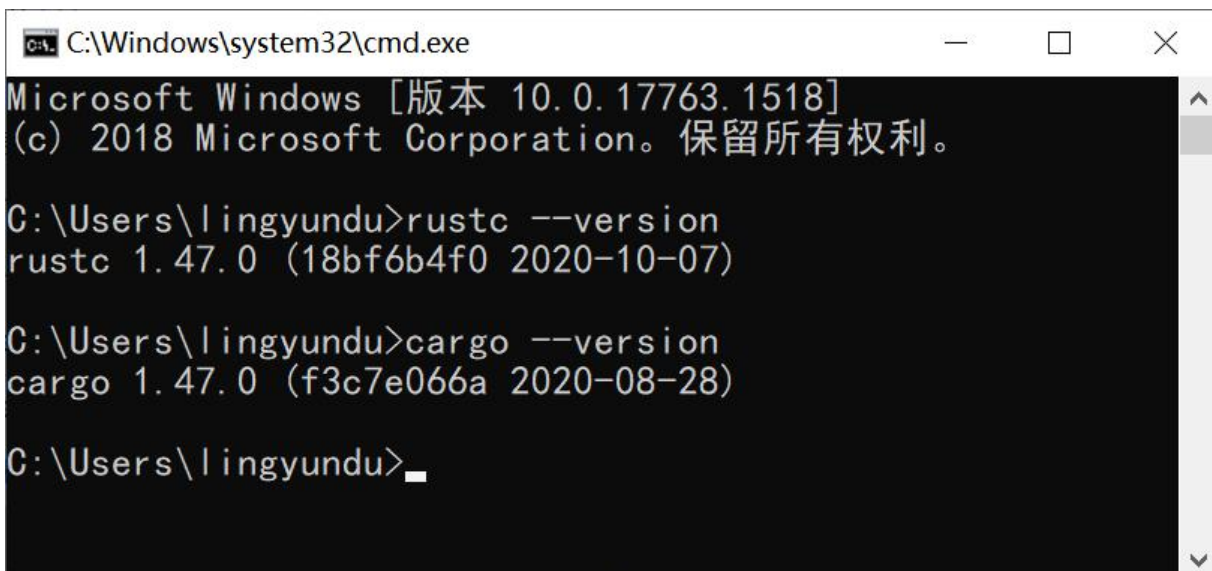
  stable installed - rustc 1.47.0 (18bf6b4f0 2020-10-07)

Rust is installed now. Great!

To get started you need Cargo's bin directory (%USERPROFILE%\cargo\bin) in
your PATH
environment variable. Future applications will automatically have the
correct environment, but you may need to restart your current shell.

Press the Enter key to continue.
```

### 确认



```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.17763.1518]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\lingyundu>rustc --version
rustc 1.47.0 (18bf6b4f0 2020-10-07)

C:\Users\lingyundu>cargo --version
cargo 1.47.0 (f3c7e066a 2020-08-28)

C:\Users\lingyundu>_
```

## 安装 VSCode 及相关插件

**配置国内 Crate 源**

**第一个 Rust 程序**



00

00

**解决方法**

**相关资料**