

全部开发者教程

深度剖析C语言知识体系

第1章 导学

1-1 引论

第2章 C语言必备基础

2-1 环境搭建

2-2 第一行C语言代码

2-3 程序调试方法

2-4 C语言的代码风格

第3章 C语言快速入门

【知识】背景知识

3-1 数据类型之int类型

3-2 数据类型之char类型

3-3 数据类型之浮点类型

3-4 变量

3-5 常量

3-6 运算符-1

3-7 运算符-2

3-8 条件分支语句

3-9 循环语句

第4章 函数与程序结构

4-1 函数基础

环境搭建

编程课的环境搭建是非常重要的一个环节。在开始课程制作之前，我专门做了一些调研，国内使用 Windows 环境的开发者占比仍然比较高，再加上开发者使用 Windows 环境做开发也通常面临相对更复杂的问题，因此课程当中为了照顾绝大多数的开发者，我专门切换成了 Windows 环境来为大家做讲解（学习过我之前的课程的同学们应该知道我平常的主要工作环境实际上是 macOS），涉及到不同平台差异的地方，我也会基于 Ubuntu 和 macOS 给出一些提示。

我们的环境搭建分为两部分：

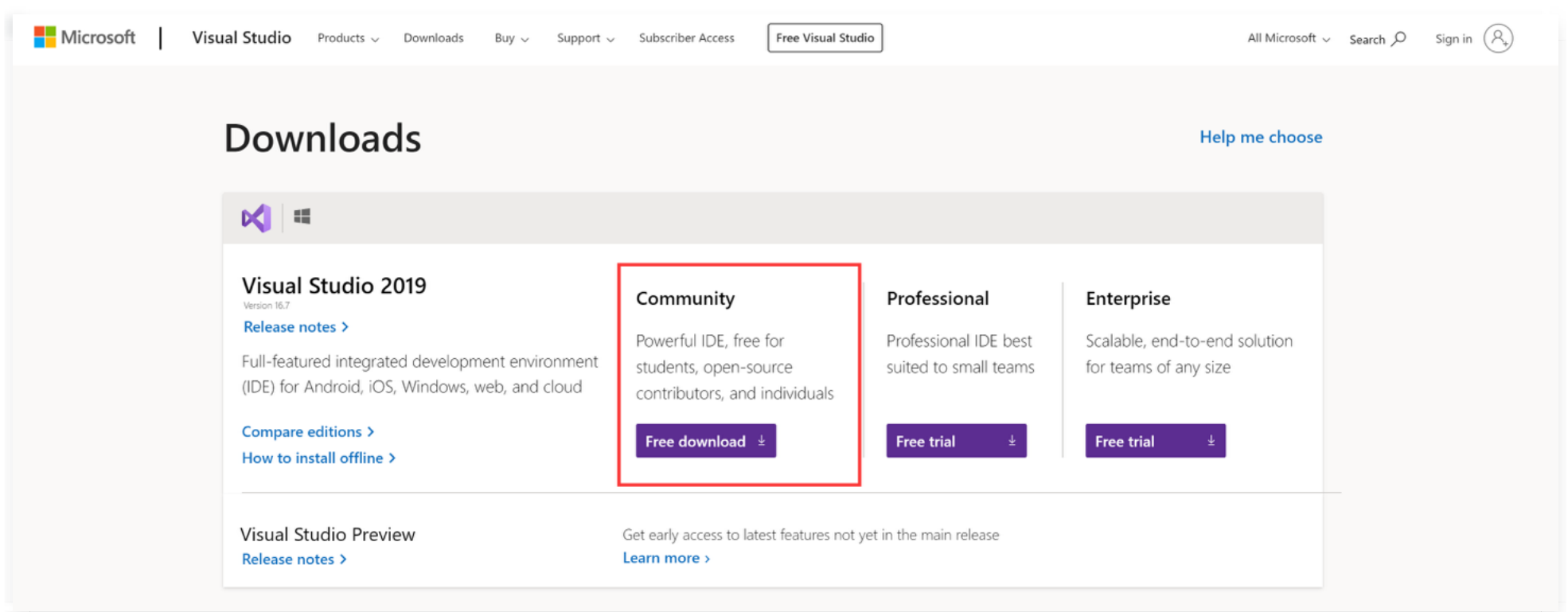
1. 安装编译器，其中包括在 Windows 上我们需要安装 Visual Studio，并选择安装其中的 MSVC 编译工具。
2. 安装 IDE。IDE 的选择有很多，例如 Windows 上的 Visual Studio 就是一款很好用的 IDE，macOS 上则有 Xcode。我们的课程选择 CLion 来为大家讲解，主要有以下原因：

- CLion 是一款跨平台的 IDE，因此不论大家使用什么操作系统，基本上都会获取一致的学习体验。
- CLion 功能强大，对各种编译器的支持也更开放。
- CLion 由 JetBrains 开发，与 IntelliJ IDEA、PyCharm 等 IDE 有着相同的使用体验，方便大家后续在学习其他语言时快速完成迁移。
- CLion 虽然收费，但可以免费使用 30 天，使用高校邮箱注册可以免费直接使用。

接下来我们看下如何完成 Windows 的环境搭建。

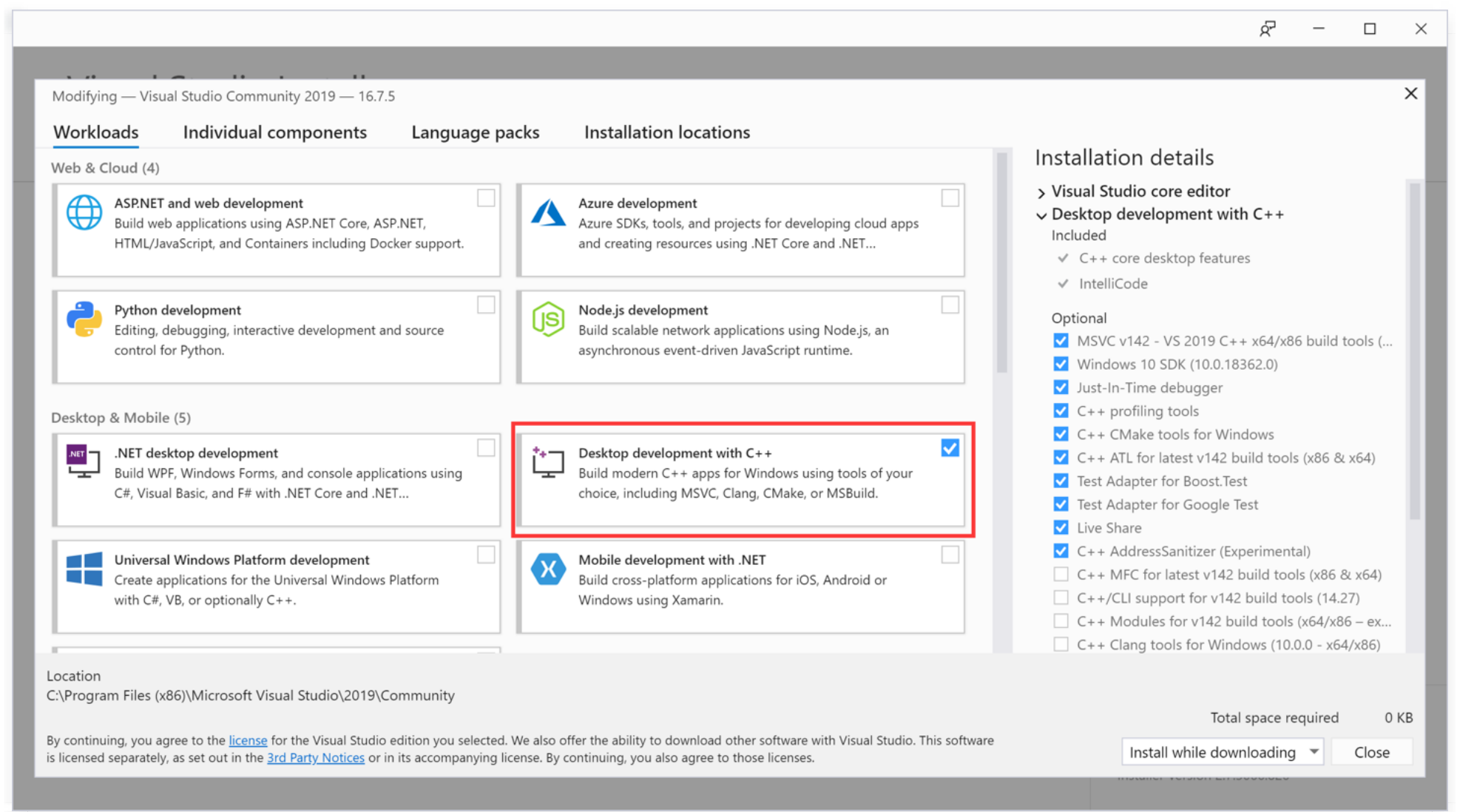
安装 Visual Studio 2019

我们需要到微软的网站下载 [Visual Studio](#)，我们的课程使用的是课程设计时的最新版本 Visual Studio 2019：



点击下载 **社区版本（Community）** 即可，这个版本是免费的。

运行这个安装程序，界面上会显示有非常多的选项，最方便的办法就是直接在 Workloads 下面选择 Desktop development with C++：

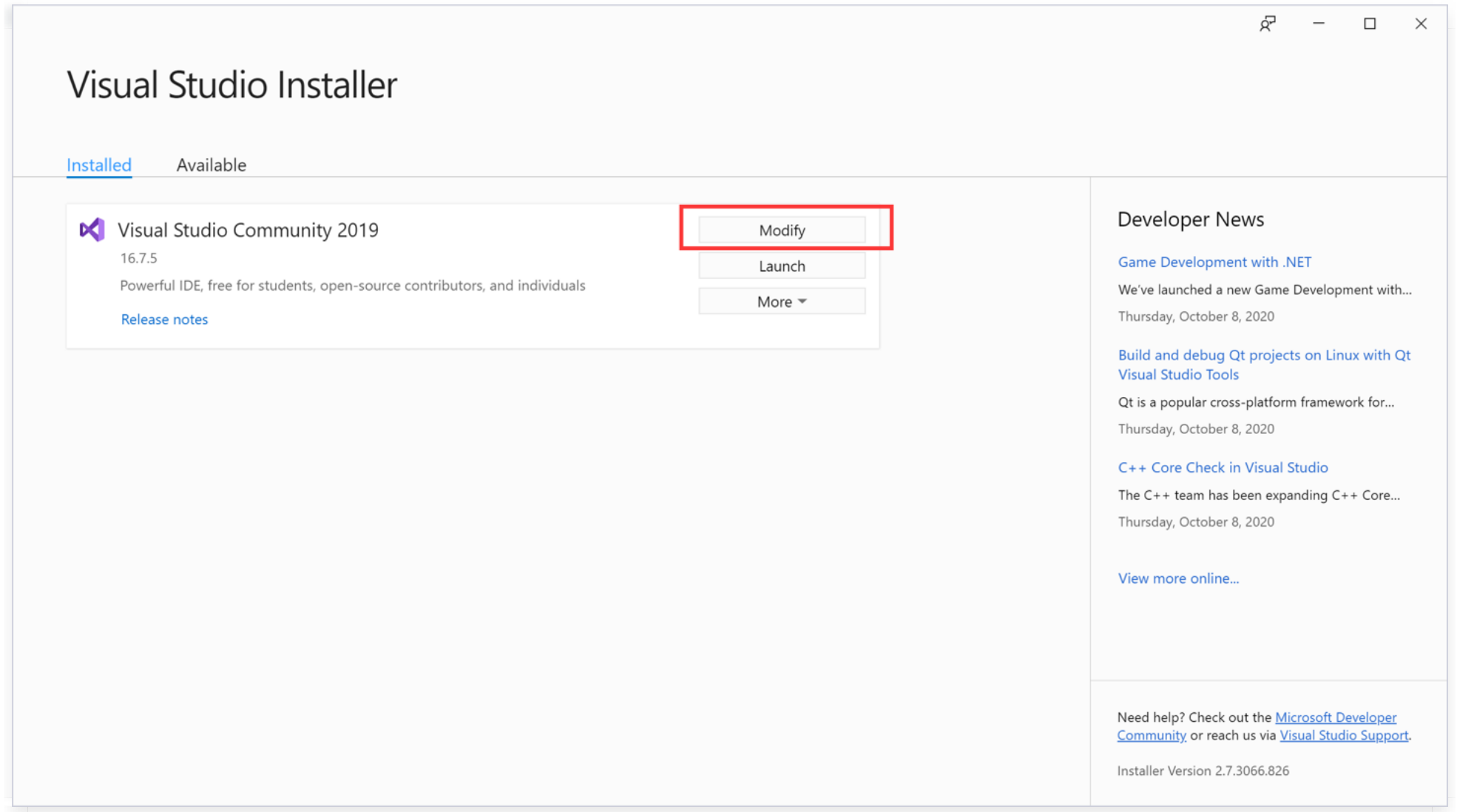


选择之后，右侧会显示这个 Workload 会包含的一些组件，点击安装即可。

安装的时间有些长，站起来活动活动，休息一下~ 安装完成之后，会提示重启，重启电脑即可。

微软官网也给出了 Visual Studio 2019 的完整安装方法，参见[安装 Visual Studio](#)。

如果你的电脑上已经安装了 Visual Studio 2019，那么直接打开 Visual Studio Installers，点击 modify，然后勾选前面的组件，完成安装即可。



实际上，Visual Studio 本身就是一个很好的集成开发环境，有兴趣的话，同学们也可以尝试使用它运行课程中的案例。

说明：课程当中所有使用 MSVC 成功运行的案例都可以在 Visual Studio 2019 当中成功运行。Visual Studio 现在也已经支持 cmake，并访问 WSL 当中的编译器，大家有兴趣也可以试试看。

安装 MinGW

使用 Windows 的开发环境，我们绝大多数情况下都会使用 MSVC 来演示我们的程序，但我们再次强调，MSVC 实际上是个 C++ 编译器，对于 C 语言的语法支持也主要涉及兼容 C++ 的部分。

如果我们想要测试 C90、C99、C11 的差异，很多时候 MSVC 就无能为力了。不过你要相信世界是美好的，微软关上了一扇门，社区就能给我们打开一扇窗，我们可以安装 MinGW 或者 Cygwin 来做到这一点，这当中我们更推荐 MinGW，而且我们在课程的最后还会用到它介绍 GUI 程序的开发。

安装 MinGW 我们推荐通过 Msys2 来安装，我们可以在 [Msys2 的官网](#) 看到它的安装方法。

Msys2 本质上提供是一套类似于 archlinux 的包管理工具，我们可以在 Msys2 的控制台中通过 pacman 来安装相应的工具包，这其中就包括 MinGW 的编译工具链。

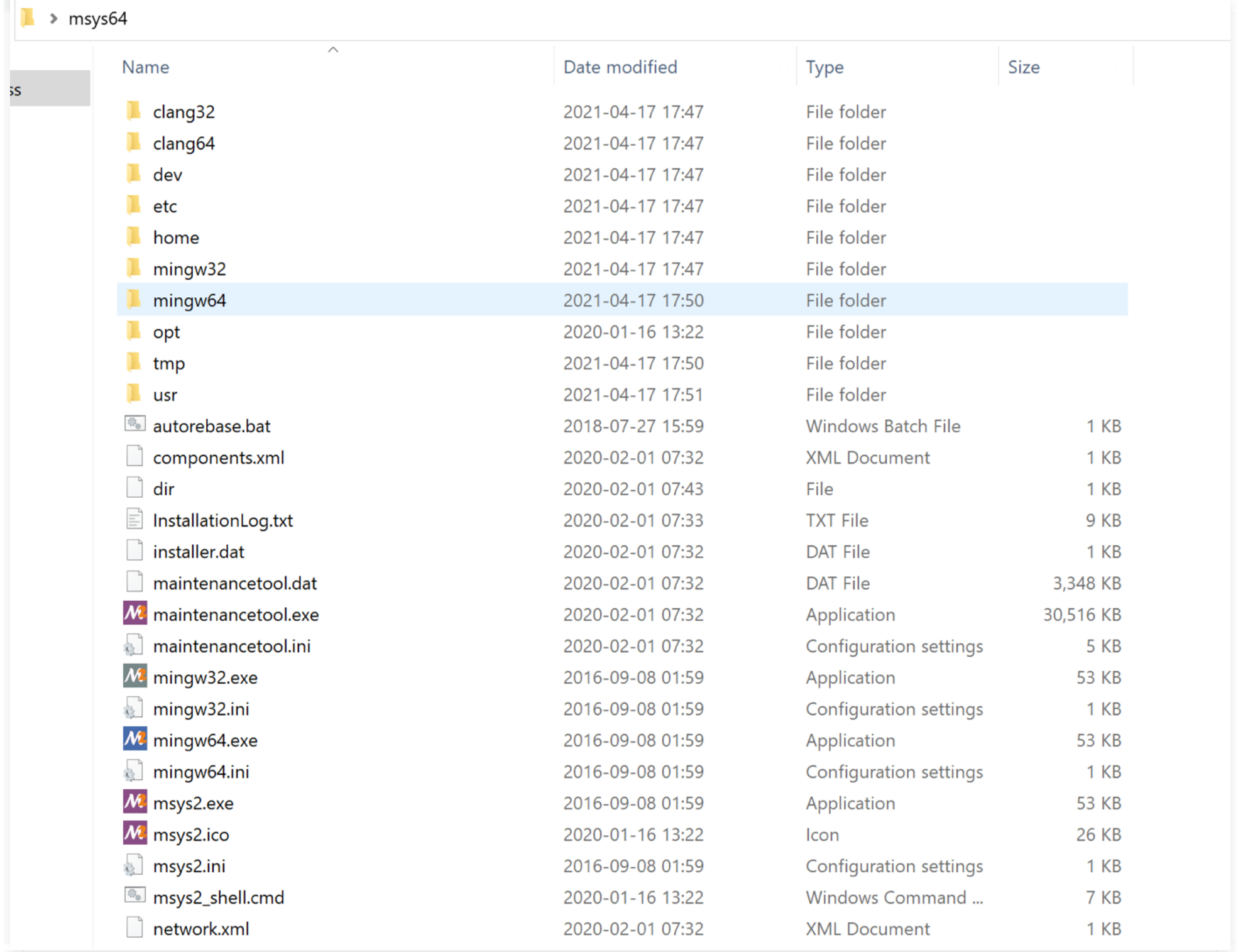
MingGW 的环境配置比较麻烦，因此我把本地的安装包打包好，大家只需要直接下载解压缩到 C 盘根目录即可，下载地址：

链接：<https://share.weiyun.com/Ph3XCz7g> 密码：b64kcv

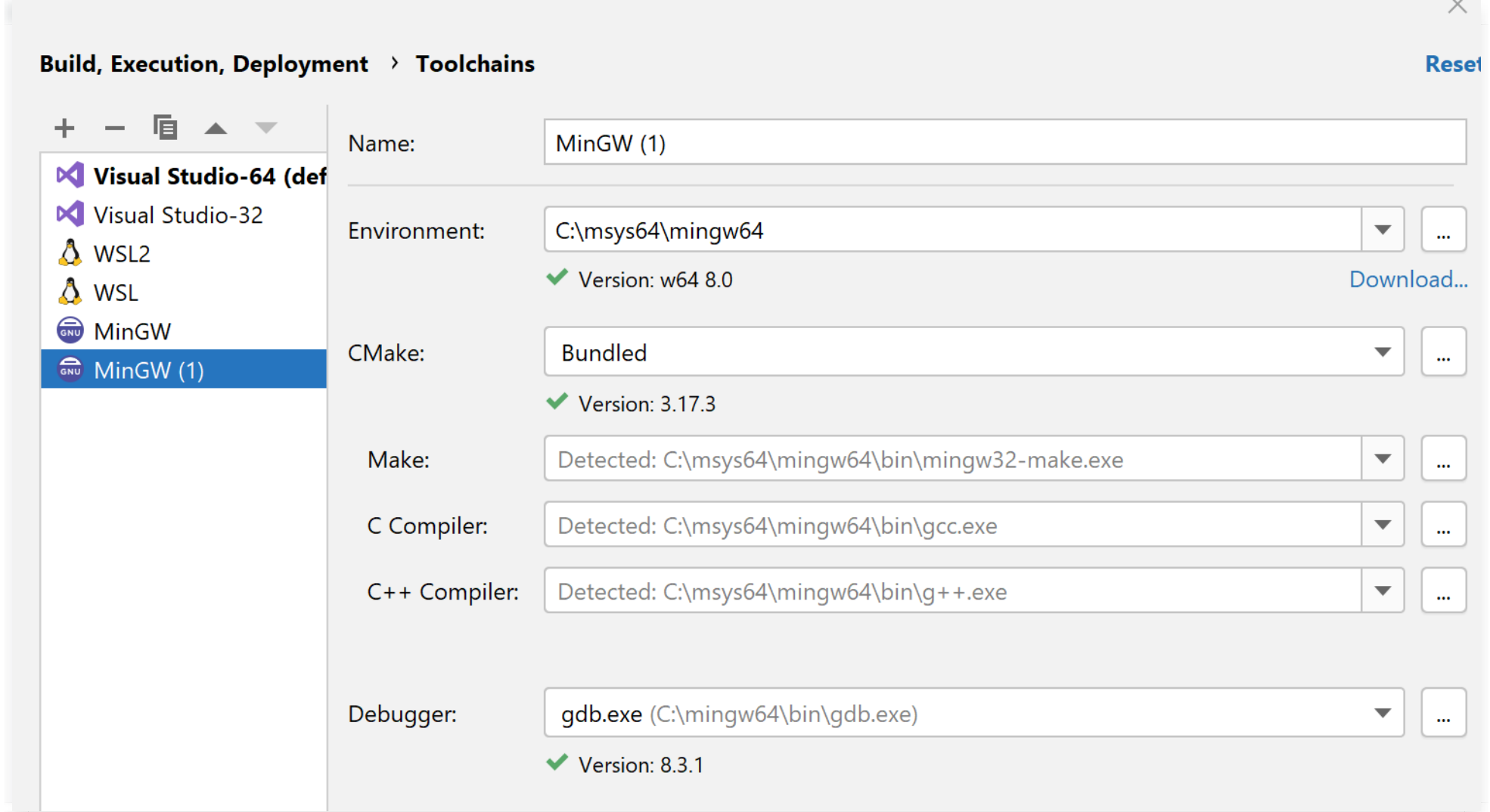
下载里面的 **msys64.rar** 即可。

下载完成以后，请按照下面的提示完成 CLion 的配置：

压缩包内就是 msys2 安装好的全部内容，里面包含了 mingw64 编译器，以及一些常用的工具。这个包下载之后解压即可，推荐解压到 C:\msys64 目录下，解压后效果如下：



在 CLion 当中配置时，注意路径的选取：



安装 WSL

WSL 并不是课程所必须的（如果你装了当然更好），如果大家想要在 Windows 上方便地使用 Linux 环境，除了通过 Virtual Box 这样的虚拟机软件安装虚拟机以外，也可以通过 WSL 来安装子系统。WSL 本身就是 Windows 子系统的意见，我们可以在 Windows 10 当中安装一个 Linux 子系统，这个子系统能够与 Windows 无缝衔接，而且性能非常好，不像使用 VirtualBox 这样的软件安装虚拟机那样耗费系统资源，因此 我们也非常建议大家安装 WSL，安装方法见微软官方的指引<https://docs.microsoft.com/zh-cn/windows/wsl/install-win10>，这样我们就可以在 Windows 上同时使用 Windows 的 MSVC 编译器以及 Linux 的 GCC 编译器来测试我们的 C 程序，并体验二者的不同。

安装并配置 CLion

CLion 的安装方法很简单，在官网下载 [CLion 安装包](#) 直接安装提示安装就行了。CLion 官方也提供了安装指引，大家也可以参考 [Install CLion](#)

Windows 上 CLion 配置工具链的方法可以参考 CLion 的官方指引 [Tutorial: Configure CLion on Windows](#)，文档中除了提供了 MSVC 的配置方法，也提供了 MinGW、Cygwin 的配置方法。

CLion 也可以直接配置使用 WSL 的编译器，也就是说如果我们在 Windows 上装了一个 Ubuntu 的子系统，我们就可以直接使用它的编译器，配置方法见 [WSL](#)。

索引目录

- [环境搭建](#)
- [安装 Visual Studio 2019](#)
- [安装 MinGW](#)
- [安装 WSL](#)
- [安装并配置 CLion](#)