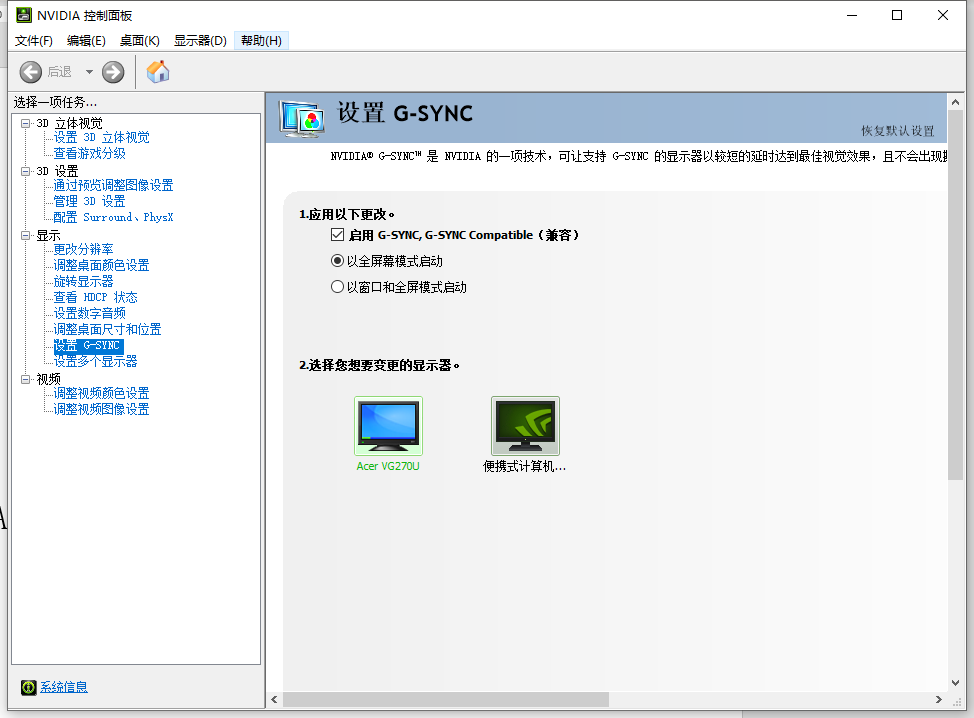
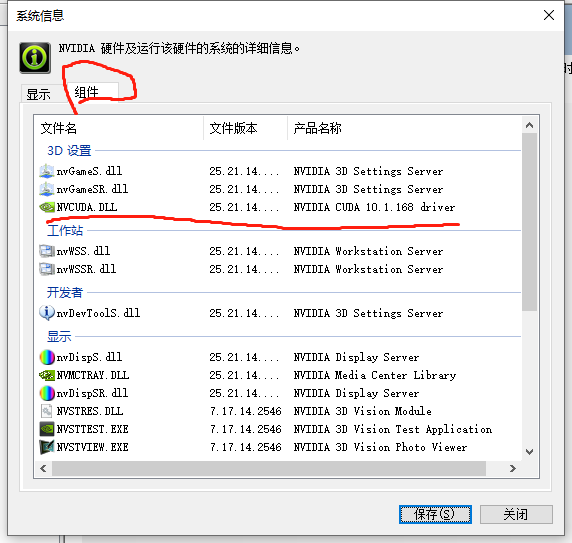
# 下述所有安装默认c盘为系统盘

# 步骤0.环境监测

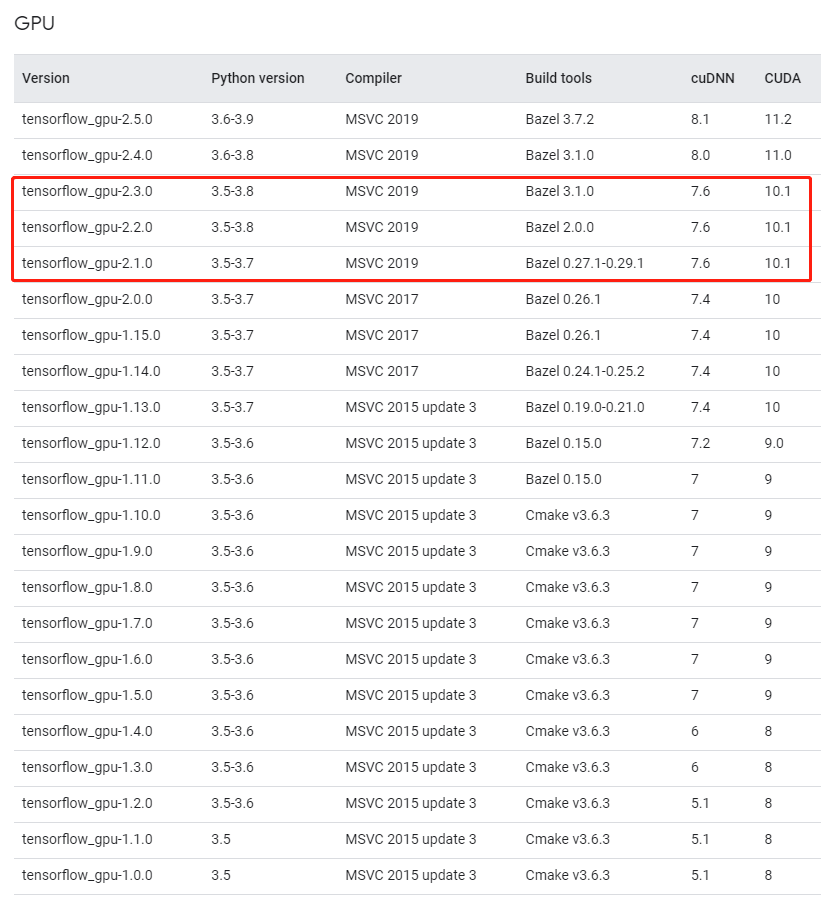
0.1查看GPU可兼容的CUDA版本：控制面板--硬件和声音--NVIDA设置—帮助—系统信息—组件；下图支持最高CUDA 10.1





0.2查看tensorflow、cuda、cudnn、python版本兼容性：<https://www.tensorflow.org/install/source_windows>

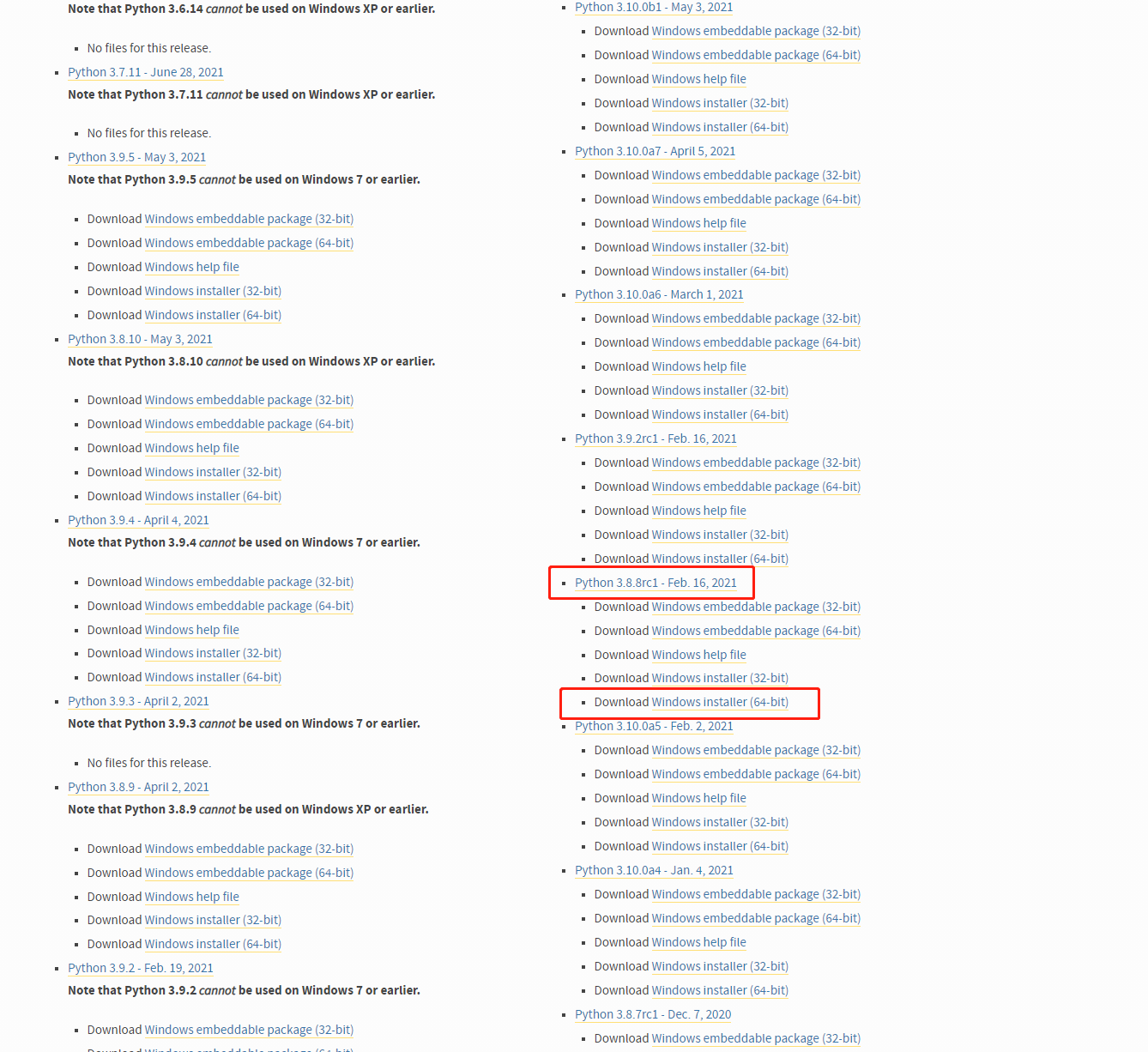
CUDA 10.1需要python3.5-3.8，cudnn7.6、tensorflow2.1-2.3.



# 步骤一. 安装python和pip

## 1.1 安装python（版本必须满足步骤0的要求）：

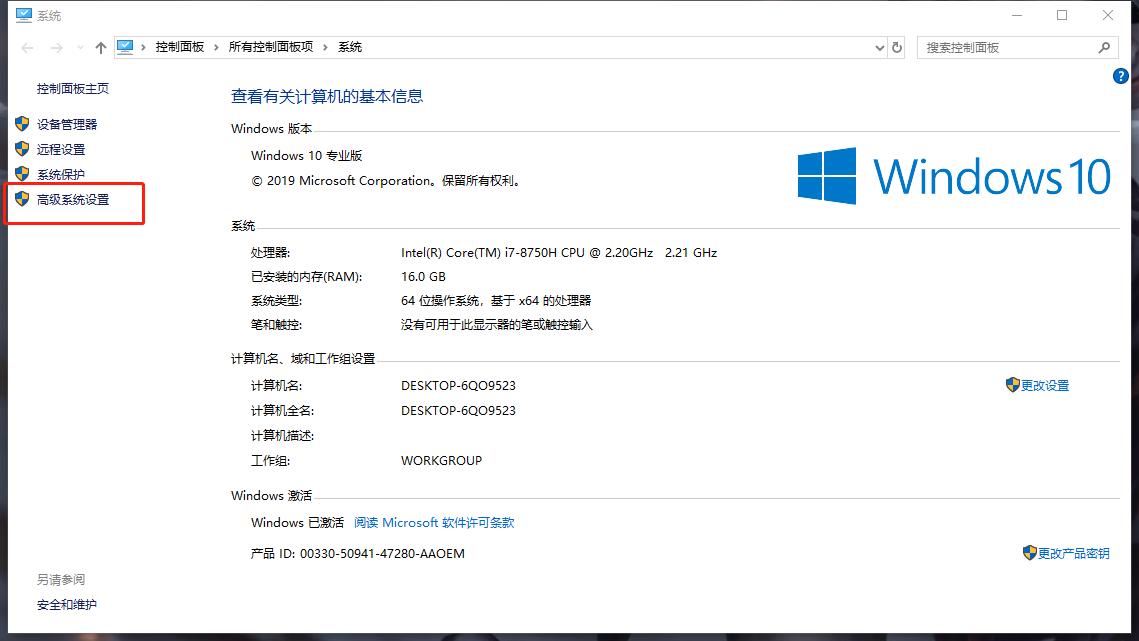
a 下载python，网址：https://www.python.org/downloads/windows/

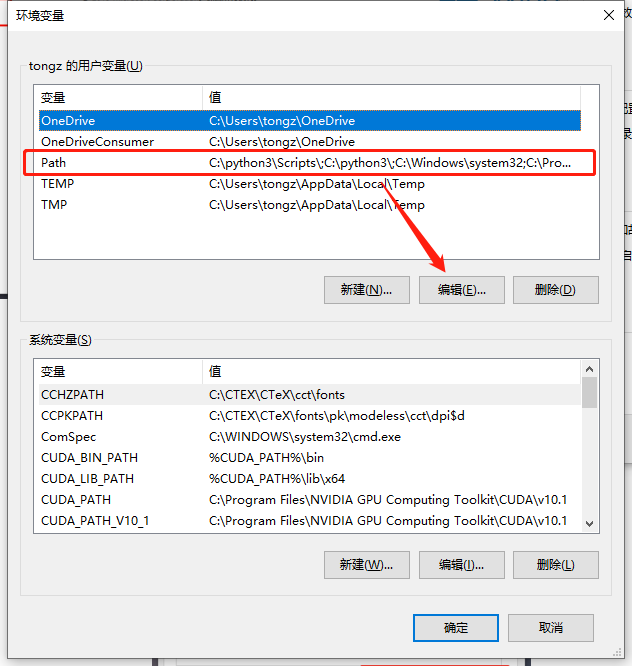


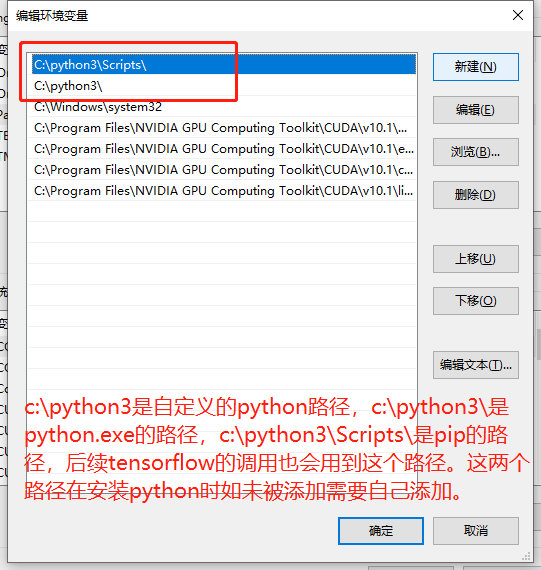
b. python自带pip，不用额外安装，或者anaconda安装；

c. 如果还有其他版本的python请自行删除，不然后续python和pip调用会出现矛盾；

d. 安装时需把python和pip的路径添加到环境变量PATH上，方法如下：



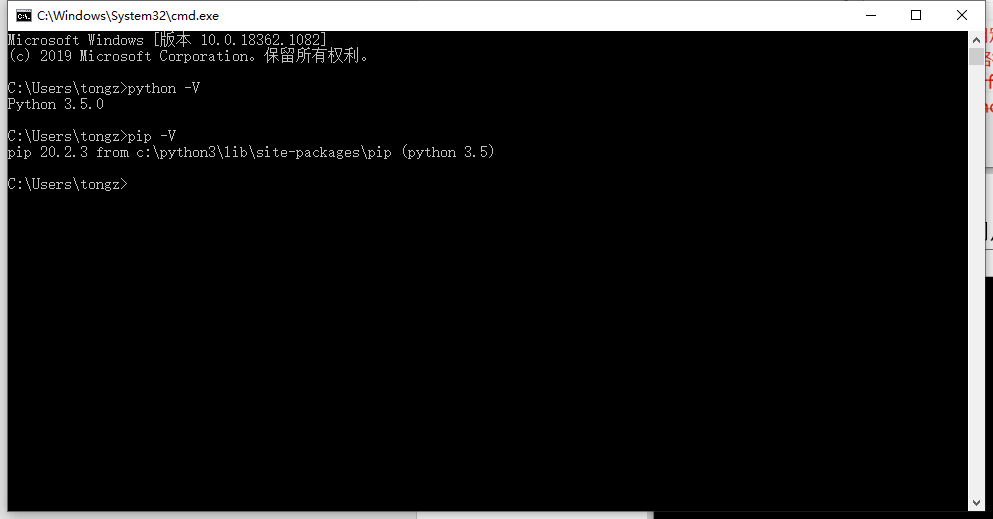


## 1.2 验证python和pip是否安装成功

a. win+R后在对话框内输入\cmd，调用cmd.exe



b. 输入python -V和pip -V查看版本，如果显示版本如下说明安装成功

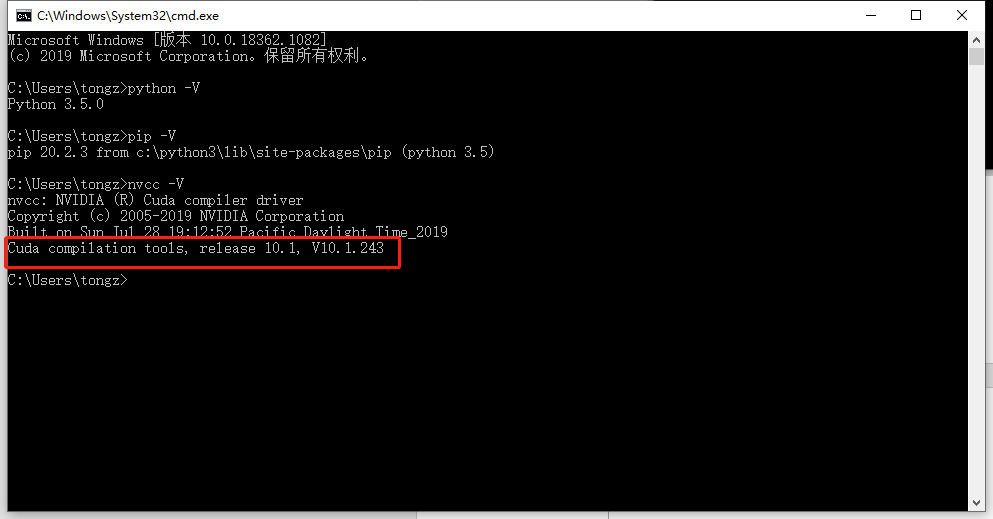


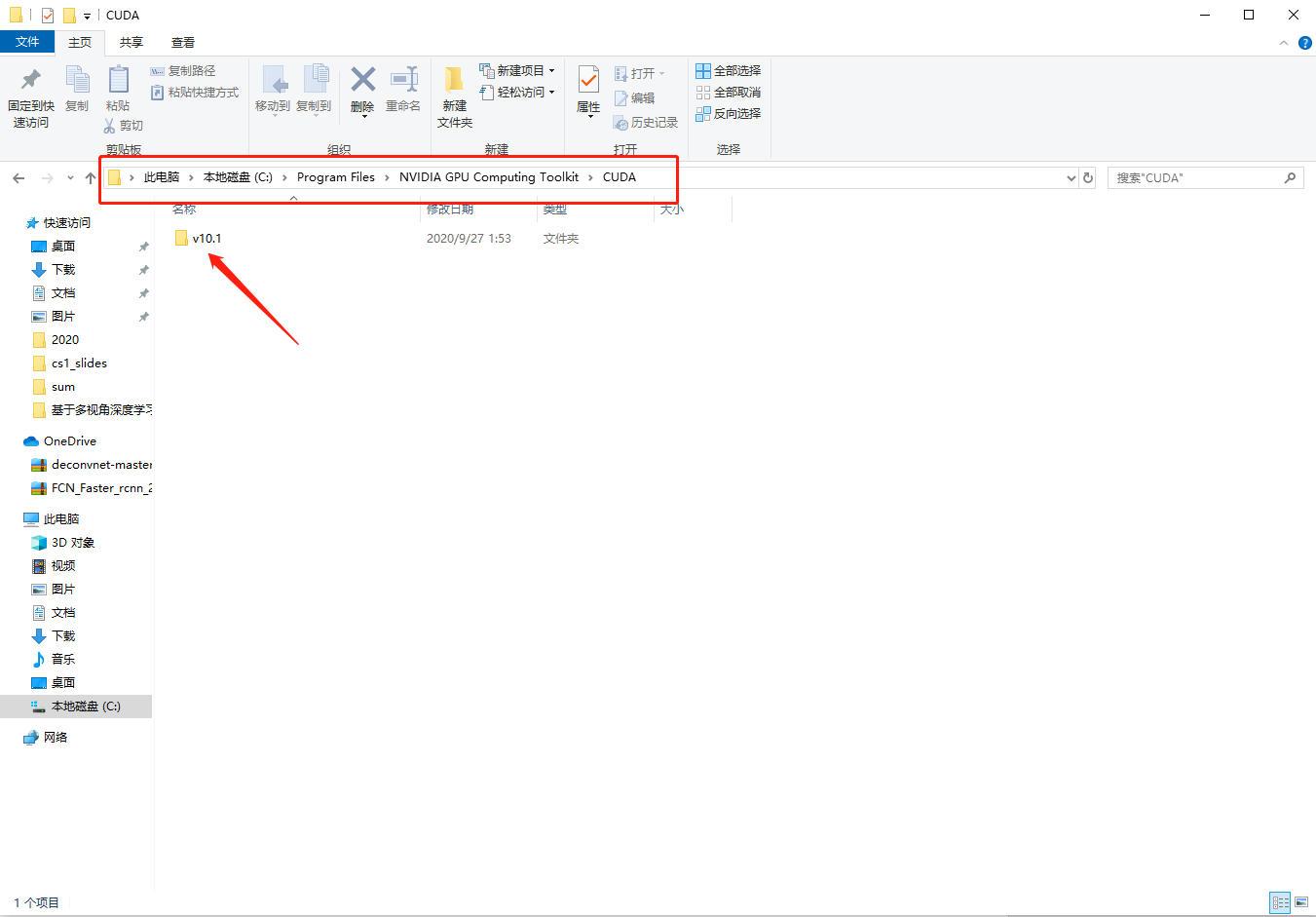
c. 如果pip版本低于19.0，采用python -m pip install --upgrade pip在cmd里面自动更新。

# 步骤二. 安装CUDA

## 2.1 检查现有的CUDA版本

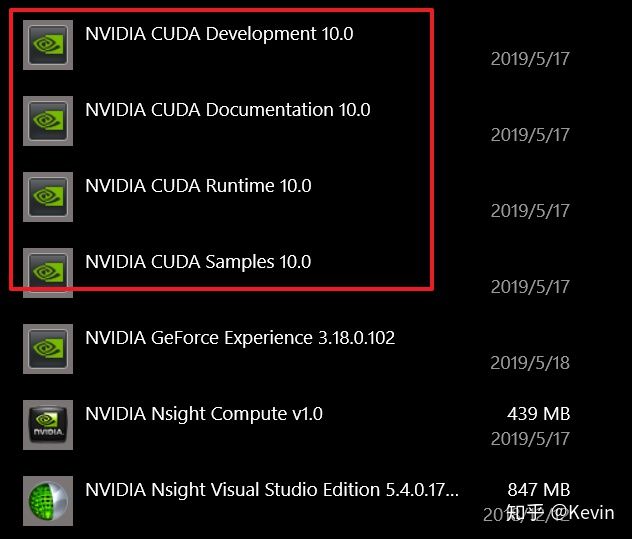
在cmd中直接输入nvcc -V，或者进入到C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\查看下一级文件夹显示的数字，如v10.1；下两图说明现有的CUDA版本是v10.1，无需重新安装；如果非v10.1，需要进行后续操作，tensorflow 2.0.0仅兼容CUDA v10.1和cudnn v7.6.0。



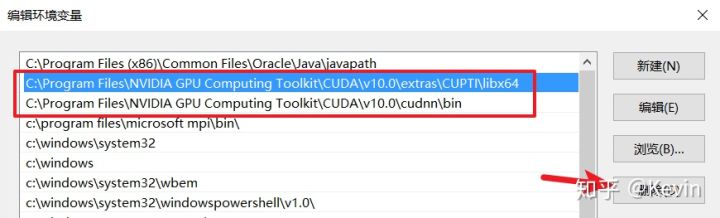


## 2.2 卸载当前版本的CUDA

a. 鼠标右击win键，选择第一项应用和功能将，带NVDIA并且具有版本号的软件都卸载掉（比如要卸载10.0，那就将下图的软件都卸载），其他不用做更改。



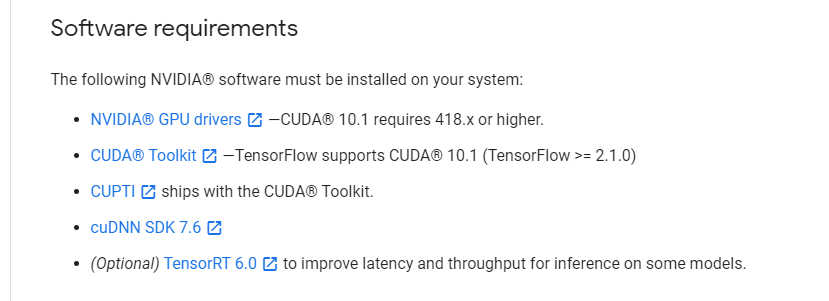
b. 进入到环境变量的path变量中，删除旧版本的两个变量。



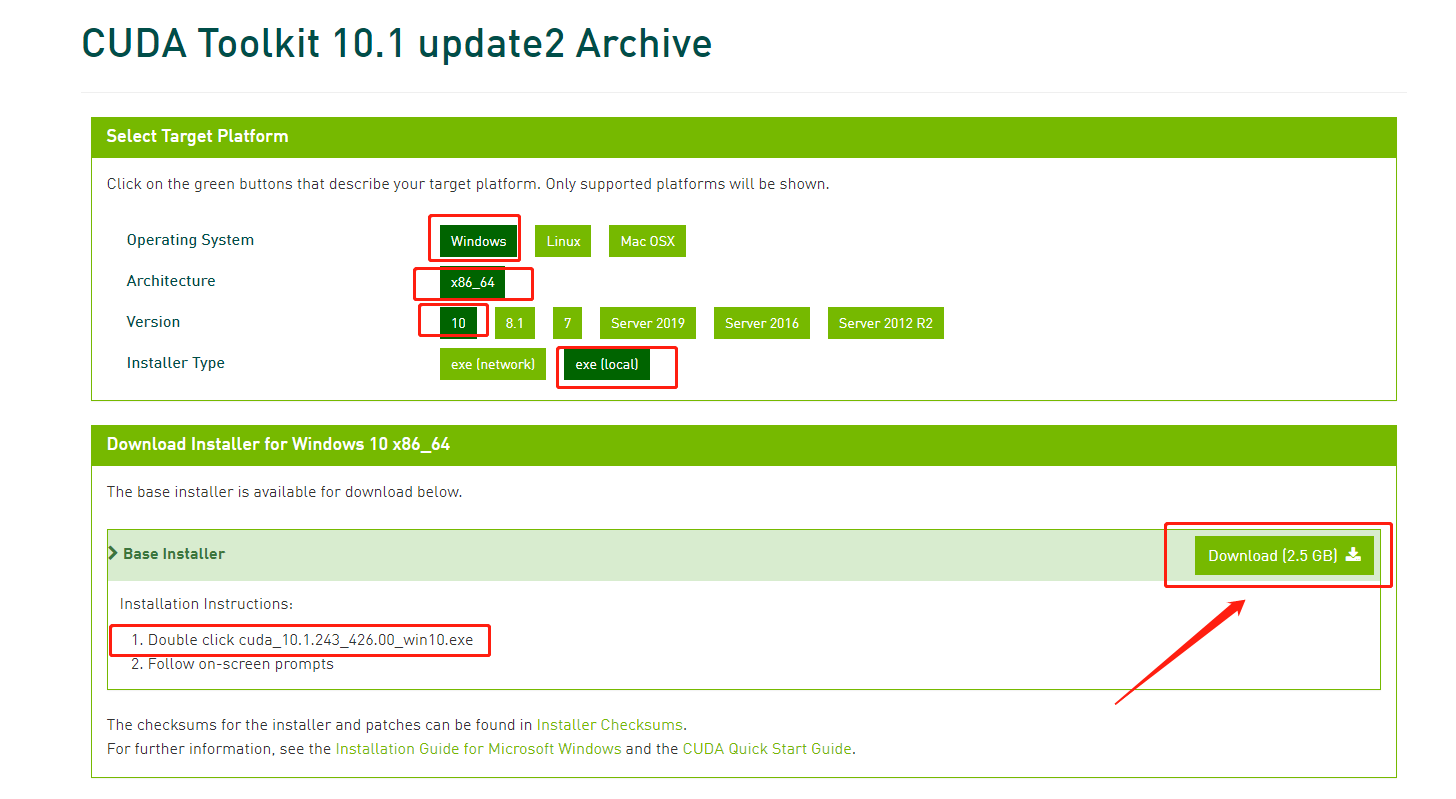
c. 删除C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA 下的v10.0或者其他版本文件夹，cudnn就在其中。

## 2.3 下载对应版本的CUDA

a. 明确需要的CUDA版本。tf官网（https://tensorflow.google.cn/install/gpu）给出了不同版本tensorflow的软件要求，tensorflow 2.0.0要求如下



b. 从NVIDA官网（https://developer.nvidia.com/cuda-10.1-download-archive-update2?target\_os=Windows&target\_arch=x86\_64&target\_version=10&target\_type=exelocal）下载对应版本CUDA v10.1，



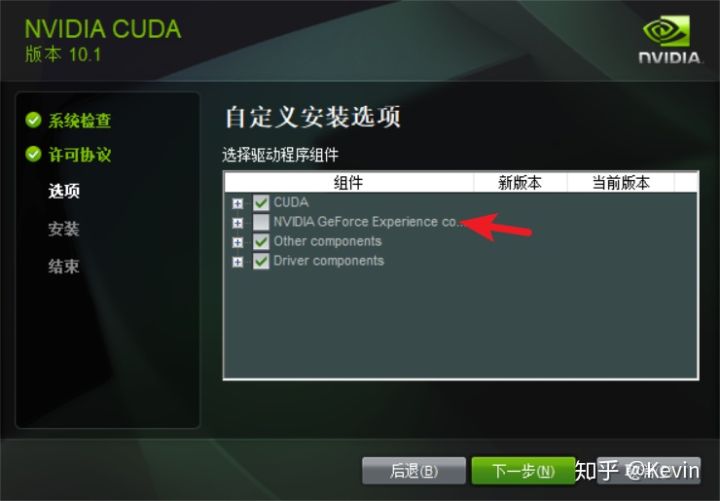
## 2.4. CUDA的安装(必须注意以下每个注意点，不然tensorflow可能无法编译)

a0. cuda\_10.1.243\_426.00\_win10.exe需要一个额外的文件夹来释放文件，建议直接在桌面弄一个给它释放，安装完成后这个文件夹会被自动删除。千万不要把释放文件夹放在C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA 下，这会导致后续安装后新安装的CUDA被自动删除。

a. 选择自定义安装



b. 去掉NVDIA GeForce Experience，因为这东西没用。对于已经搭载了GTX 1050及以上显卡的电脑，此项必须点去，不然会存在两个显卡驱动，导致下次电脑无法正常开机显示。



c. 必须去掉在CUDA中的visual studio integration这一项



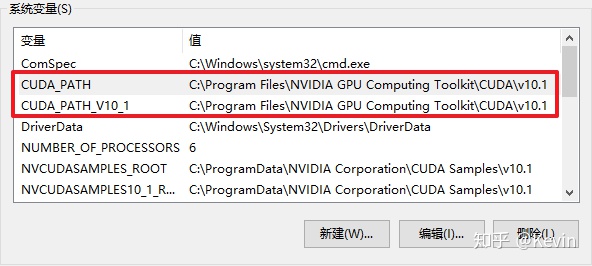
d. 请确认其他要安装的组件，都满足“新版本”（待安装版本）比“旧版本”（已有版本）要高级的，否请请去掉勾选。为了方便安装，可以全部不选这些选项。



e. 接下来下一步完成安装就好了，路径用它的默认路径，千万别修改。

## 2.5 检查CUDA

a. 在系统变量中，确认以下两项：



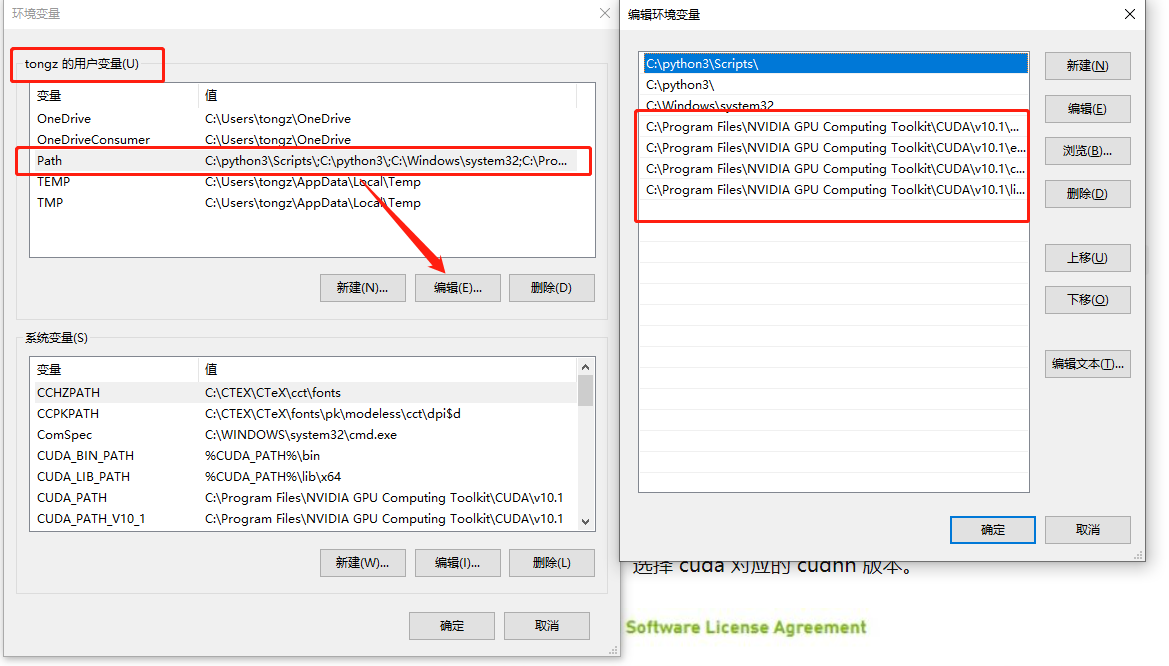
b. 在用户变量的PATH下添加以下四项：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\bin

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\extras\CUPTI\lib64

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\cudnn\bin

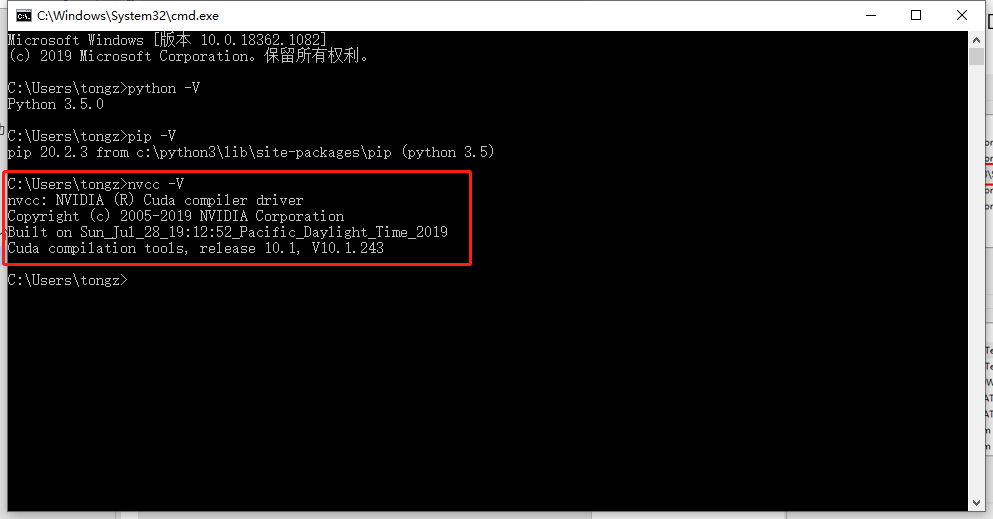
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\libnvvp



c. 打开路径 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\bin 检查nvcc.exe是否存在。

d. 打开路径 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\extras\CUPTI\lib64 检查cupti64\_101.dll是否存在

e.使用cmd查看新的CUDA的版本，如下则CUDA安装成功



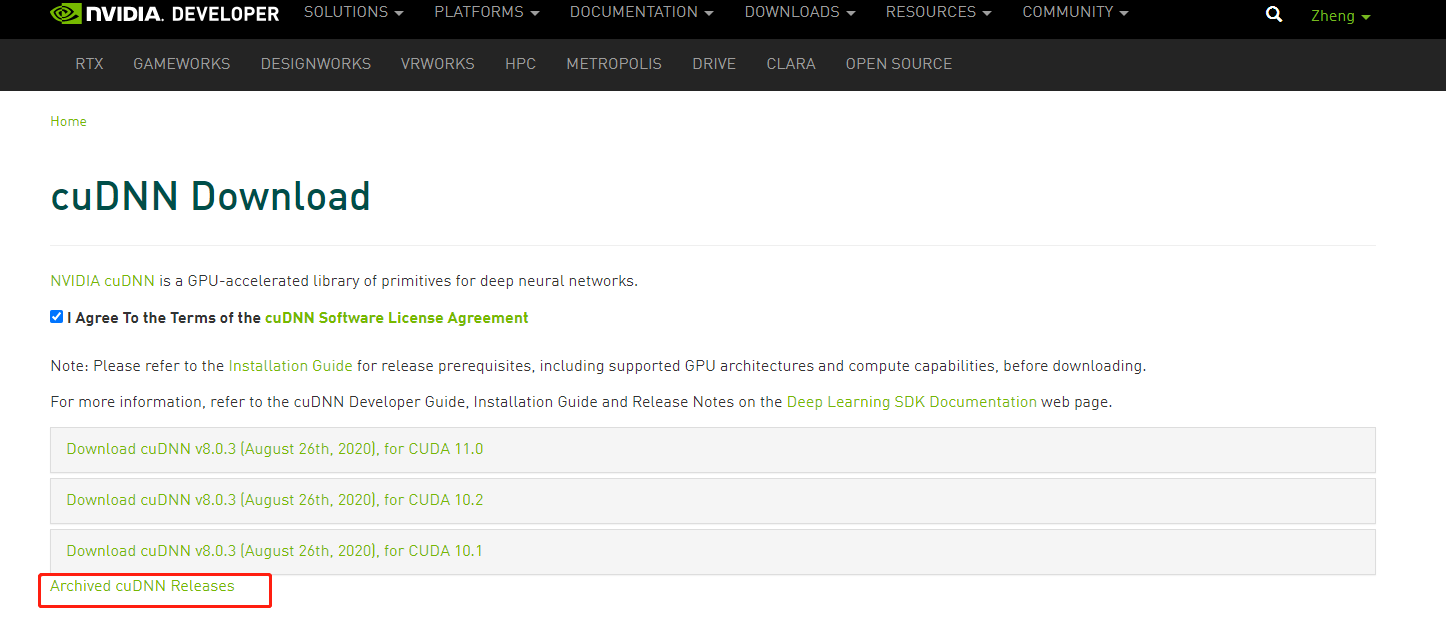
# 步骤三. 安装cudnn

## 3.1 注册NVIDA账号

<https://developer.nvidia.com/rdp/form/cudnn-download-survey>，需要自己注册账号

## 3.2 下载对应版本的cudnn

因为最新的cudnn和tensorflow 2.0.0不兼容，选择下方红框



然后找到我们需要的版本再下载，一定要注意cudnn版本号和cudnn适用的CUDA



## 3.2 安装cudnn

将下载到的压缩包解压，解压完成后文件夹下的整个文件夹复制到cuda的安装路径C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1下。注意解压后的文件夹名为cuda，请将其改成cudnn。



## 3.3 安装cudnn

（1）在 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\cudnn\bin 下确认是否存在cudnn64\_7.dll动态库。必须是\_7，如果是其他数字说明cudnn下载错了，重新去找。

（2）将cudnn文件下bin、include、lib文件夹中的文件分别放入CUDA v10.1文件夹中对应的文件夹位置：

-cudnn的bin中包含cudnn64\_7.dll放入C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\bin中

-cudnn的include中包含cudnn.h放入C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\include

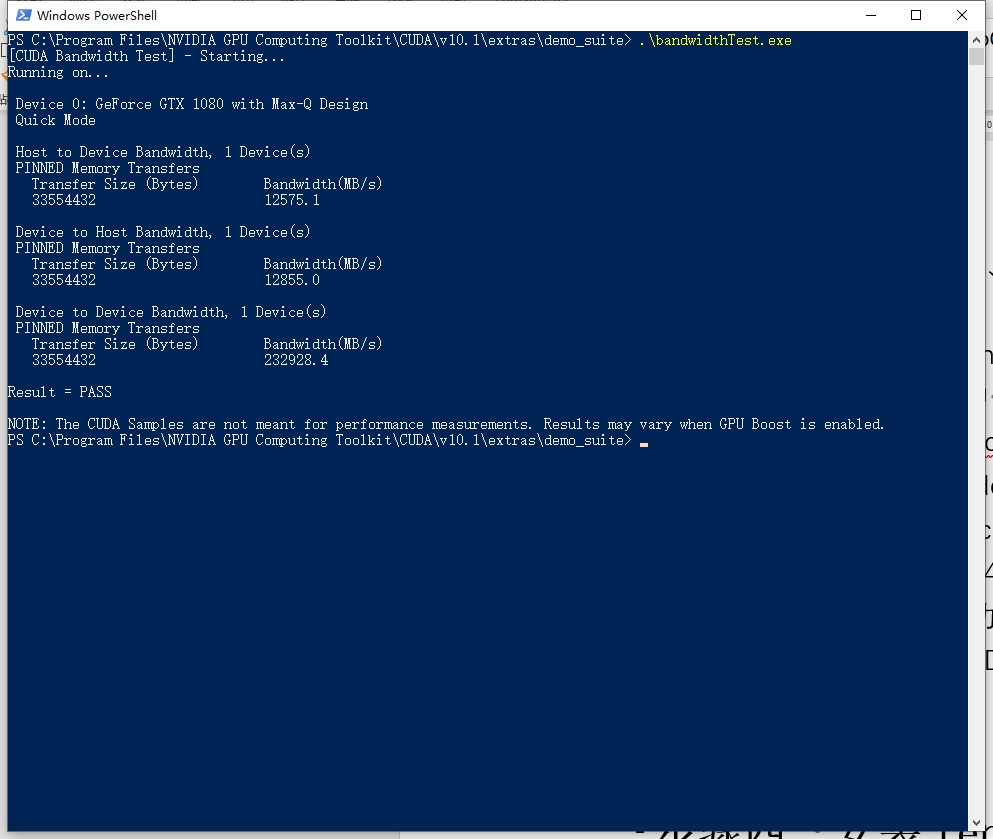
-cudnn的lib\x64中包含cudnn.lib放入C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\lib\x64

（3）测试cudnn是否成功安装

打开C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.1\extras\demo\_suite

使用powershell

输入.\bandwidthTest.exe，结果如下则安装成功



# 步骤四. 安装tensorflow

可以参照官方攻略进行安装，推荐采用下方步骤安装

## 4.1 再次确认系统版本

我们是python 3.8.8，pip 20.2.3，CUDA 10.1，cudnn7.6，win10，且存在GTX 950以上显卡，不存在问题

## 4.2 安装Microsoft Visual C++

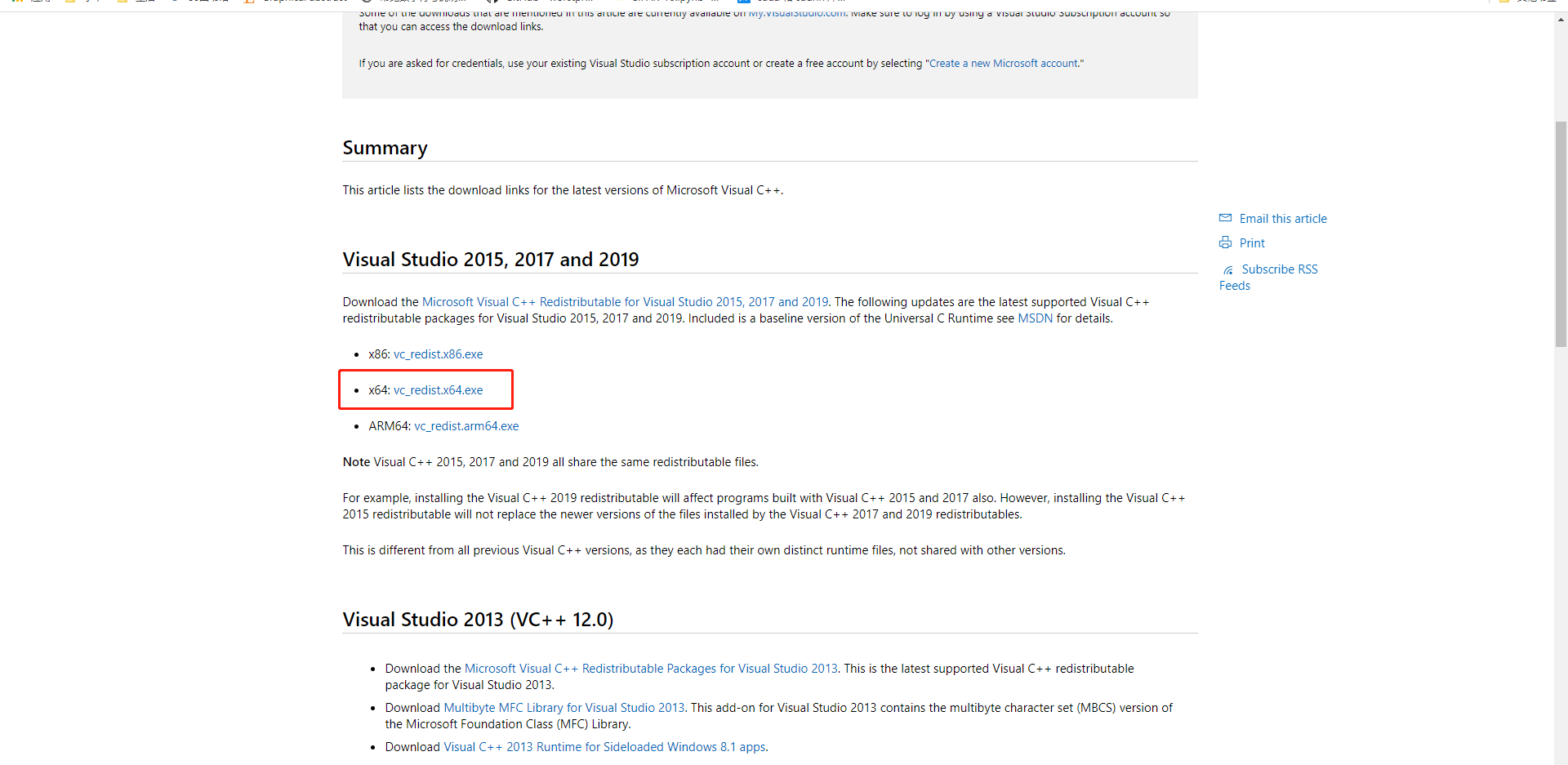
安装适用于 Visual Studio 2015、2017 和 2019 的 Microsoft Visual C++ 可再发行软件包。从 TensorFlow 2.1.0 版开始，此软件包需要 msvcp140\_1.dll 文件（旧版可再发行软件包可能不提供此文件）。该可再发行软件包随附在 Visual Studio 2019 中，但可以单独安装：

a. 转到 [Microsoft Visual C++ 下载](https://support.microsoft.com/en-us/help/2977003/the-latest-supported-visual-c-downloads/)(https://support.microsoft.com/en-us/help/2977003/the-latest-supported-visual-c-downloads)页面。

b. 在页面中向下滚动到“Visual Studio 2015、2017 和 2019”部分。

c. 为您的平台下载并安装适用于 Visual Studio 2015、2017 和 2019 的 Microsoft Visual C++ 可再发行软件包。

d．下载完直接安装就好。

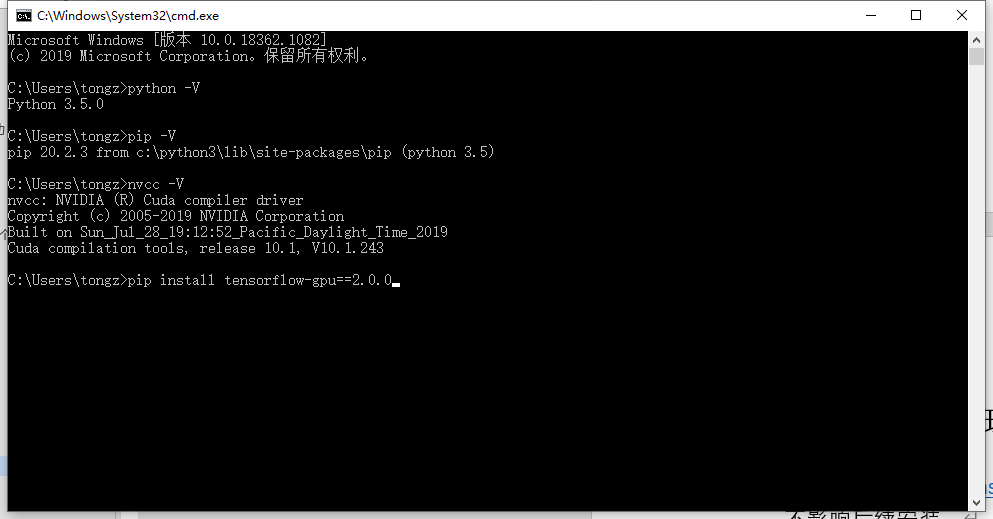


## 4.3 创建虚拟环境（可以不进行）

参考[https://www.tensorflow.org/install/pip?lang=python3#windows](https://www.tensorflow.org/install/pip?lang=python3" \l "windows)。我没有进行这一步，不影响后续安装。

## 4.4 安装tensorflow

在cmd中输入pip install tensorflow-gpu==2.3.0，回车，pip会自动安装和编译，下载和安装需要一些时间，最后会提示successfully install xx；xx里面包括很多东西，其中包括tensorflow 2.3.0。

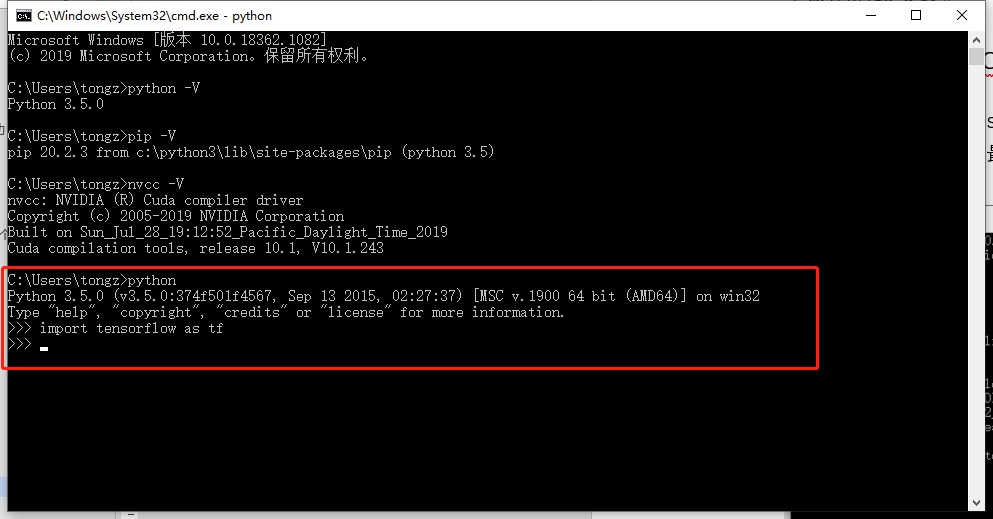


## 4.4 检测tensorflow

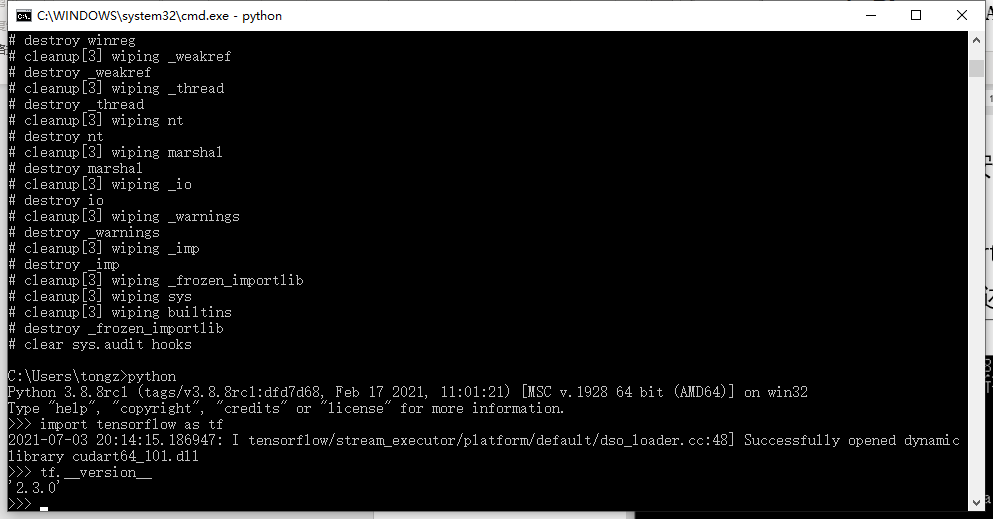
（1）检测tensorflow是否安装完成、安装的版本

在cmd中进入python

在python界面中输入import tensorflow as tf，如没有提示错误，说明tensorflow安装完成。如果提示错误，检测上述所有步骤哪里出了问题。

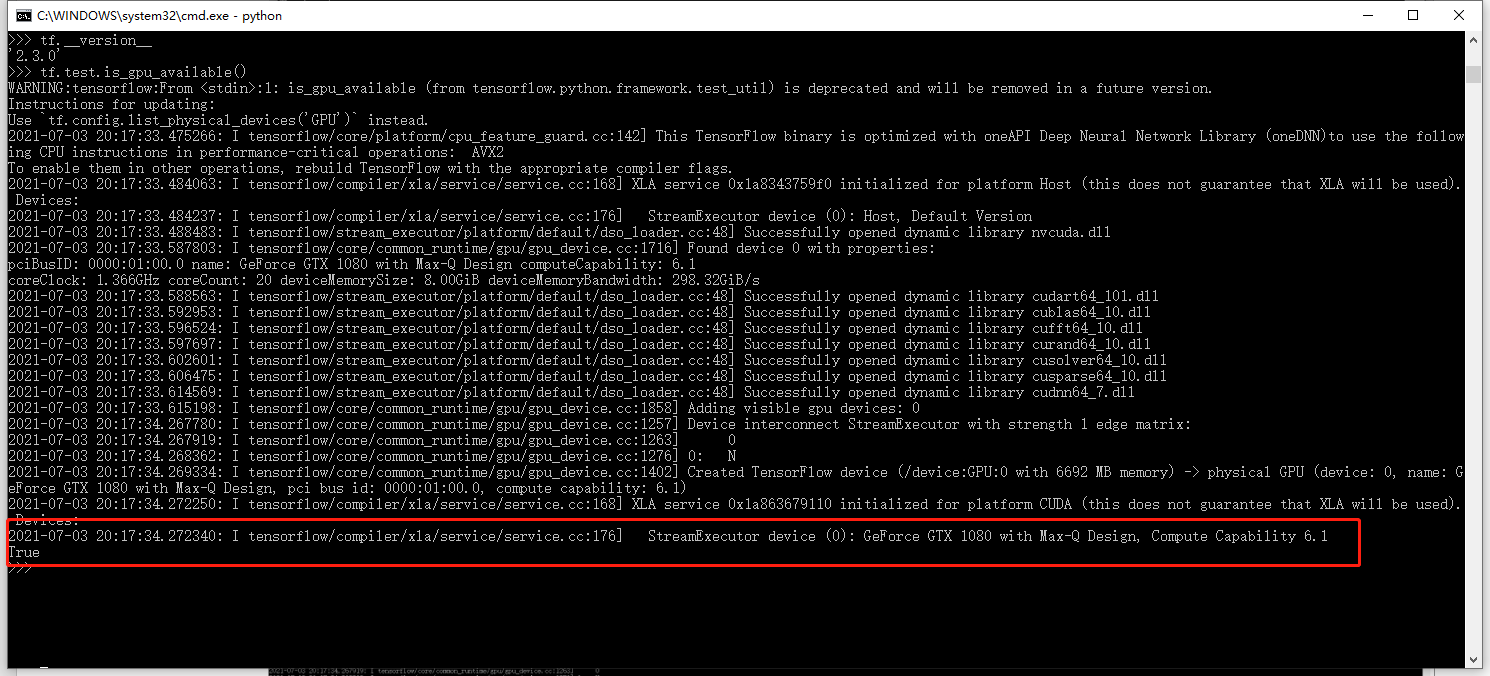


（2）在python界面中输入tf.\_\_version\_\_查看版本



（3）查看是否成功配置GPU

在python界面中输入tf.test.is\_gpu\_available()，如图显示可用GPU



# 步骤五. python 3.5.0和jupyter notebook不兼容的问题

使用pip install jupyter安装jupyter notebook后仍然无法使用，此时是因为tornado版本的问题，进行更新就好。pip install tornado==5.1.1。之后就可以用cmd调用了。jupyter notebook也可以调用tensorflow.

