并行计算实验一报告
——OpenMP 及 CUDA 实验环境的搭建

PB20111701 叶升宇 **PB20111689 蓝俊玮**

说明

以下为 PB20111701 叶升宇的实验环境。

OpenmMP 环境搭建

下载 CLion

本学期并行计算实验使用的 IDE 为 CLion，下载链接如下 *CLion*。

下载 MinGW64

使用 MinGW64 作为工具集，下载链接如下 *MinGW64*。

配置 LLVM + clang

前往 *LLVM* 官网下载，c 编译器使用 LLVM clang，c++ 编译器使用 LLVM clang++，构建工具使用 Ninja(而非通常的 Make)。整体工具链配置完毕后如下：

配置 CMake

编写 CMakeLists.txt 如下：

- 使用 c++ 14 标准；
- 用 `-fopenmp` 选项开启 OpenMP 支持

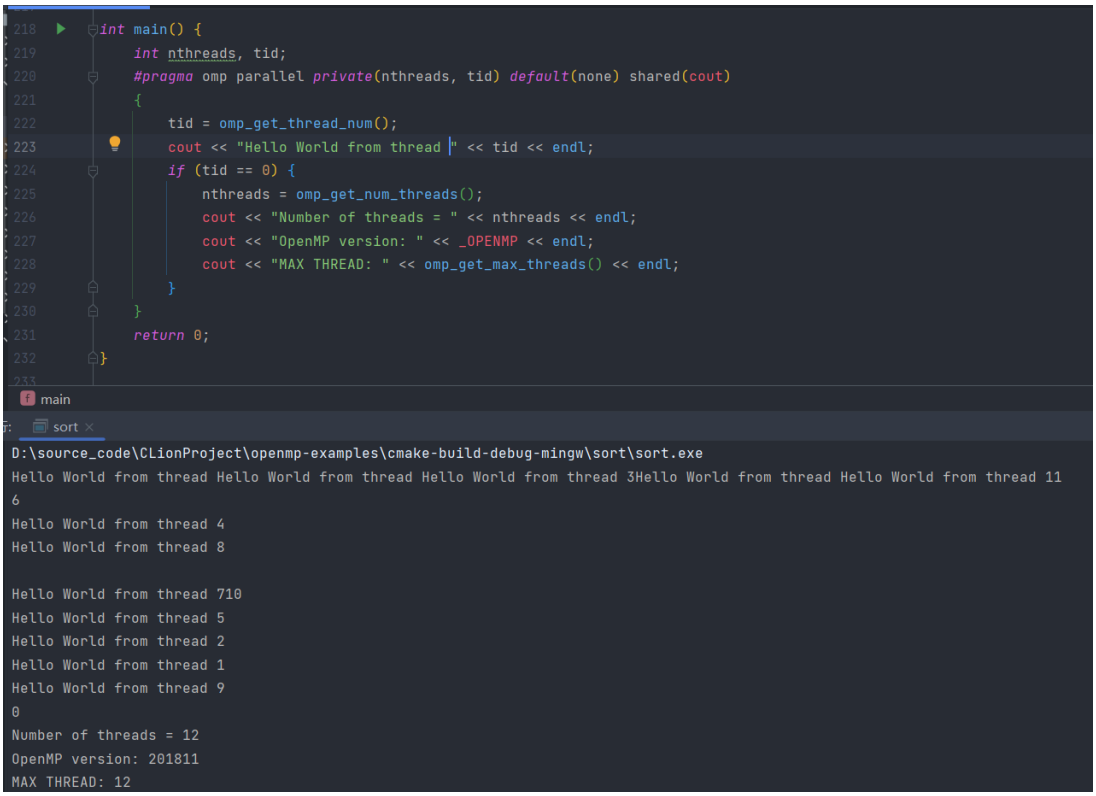
```
cmake_minimum_required(VERSION 3.10)
project(sort)
set(CMAKE_CXX_STANDARD 14)
find_package(OpenMP REQUIRED)
set(SOURCE_FILES main.cpp)
```

```
add_executable(sort ${SOURCE_FILES})
set(CMAKE_C_COMPILER "clang")
set(CMAKE_CXX_COMPILER "clang++")
set(CMAKE_C_FLAGS "${CMAKE_C_FLAGS} -fopenmp")
set(CMAKE_CXX_FLAGS "${CMAKE_CXX_FLAGS} -fopenmp")
```

Cmake 相关配置如下：

验证配置成功

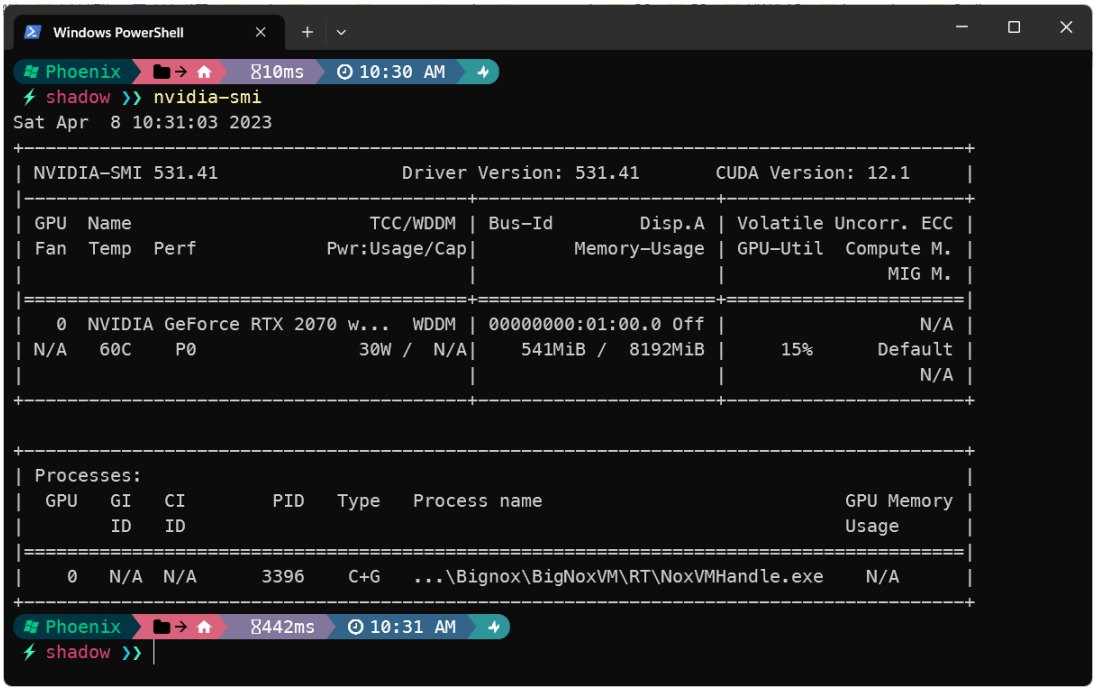
打印简单的 OpenMP 相关参数， 来验证配置成功：



CUDA 环境搭建

NVIDIA 驱动的安装

依然使用 CLion 作为 IDE，配 OpenMP 时已经完成，不再赘述 。 下面检查驱动是否安装：在 Windows Terminal 中输入 nvidia-smi：



说明

这里我是已经配好了 CUDA 环境，所以驱动和 CUDA 版本都有显示 。

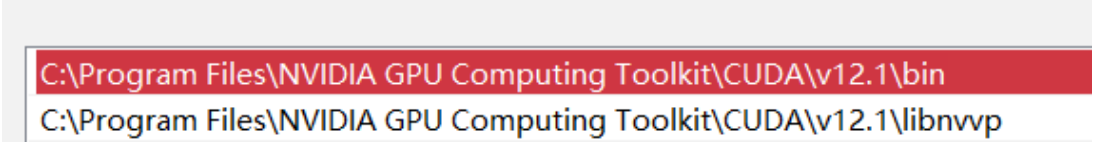
没有安装驱动的情况下，需要到 英伟达官网 选取对应显卡的版本下载安装。

CUDA 安装

进入 *CUDA Toolkit* 官网 进行下载并安装，我选择的是 12.1 版本，同时在系统变量中加入 CUDA 路径：

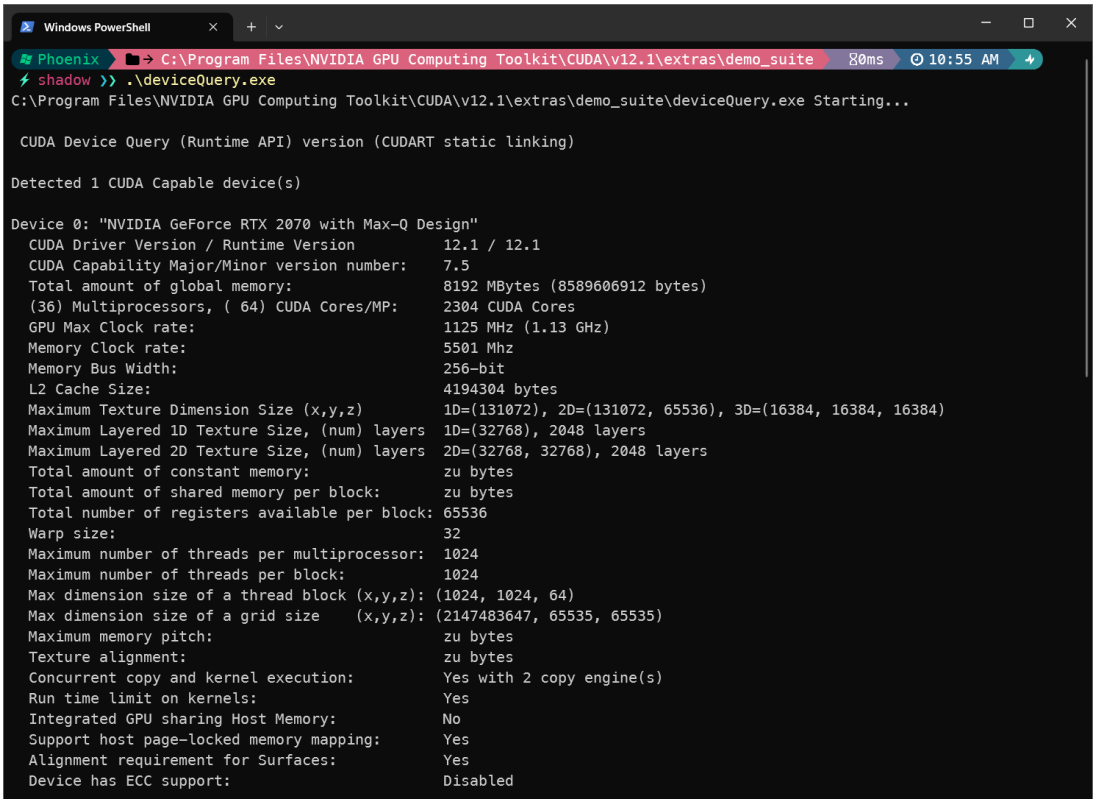
CUDA_PATH	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v12.1
CUDA_PATH_V12_1	C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v12.1

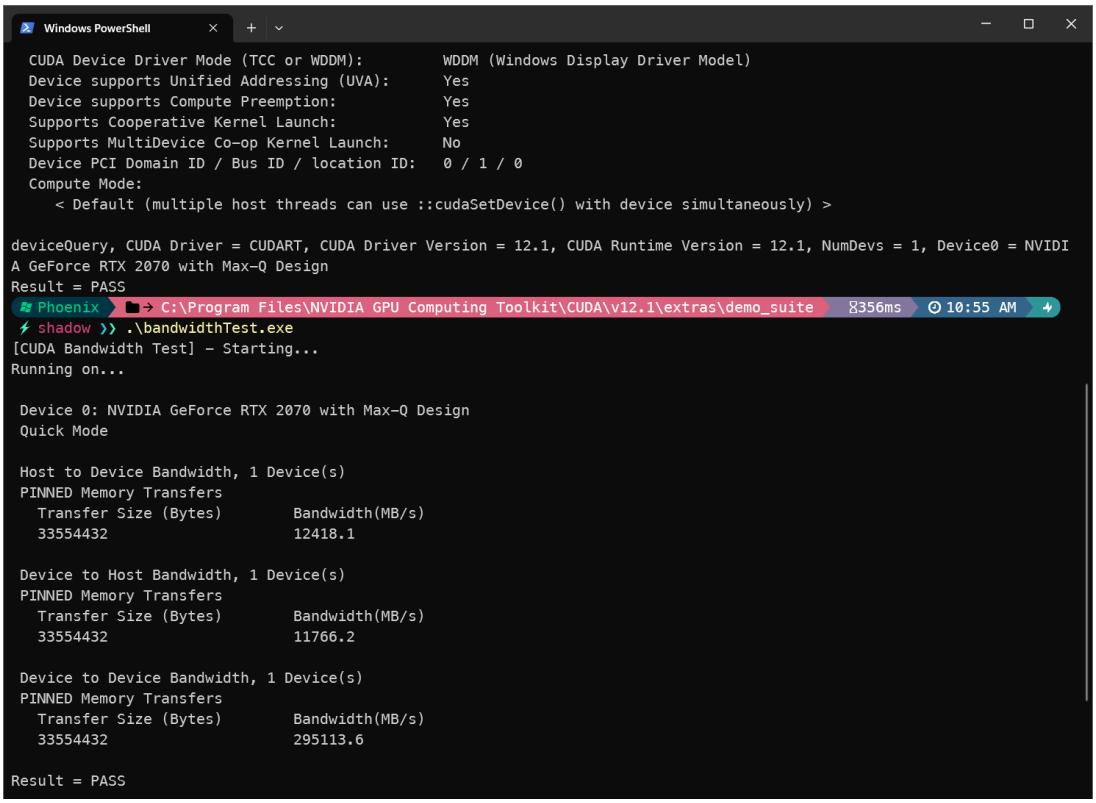
以及在 PATH 中加入 CUDA 的 BIN 路径：



验证安装成功

下面通过运行 `deviceQuery.exe`、 `bandwidthTest.exe` ，`result = PASS` 说明安装成功：





配置 CMake

在 CLion 中，选择项目为 CUDA 可执行文件，然后编写 CMakeLists.txt 如下：

```
cmake_minimum_required(VERSION 3.10)
project(cuda_examples CUDA)
set(CMAKE_CUDA_STANDARD 14)
```

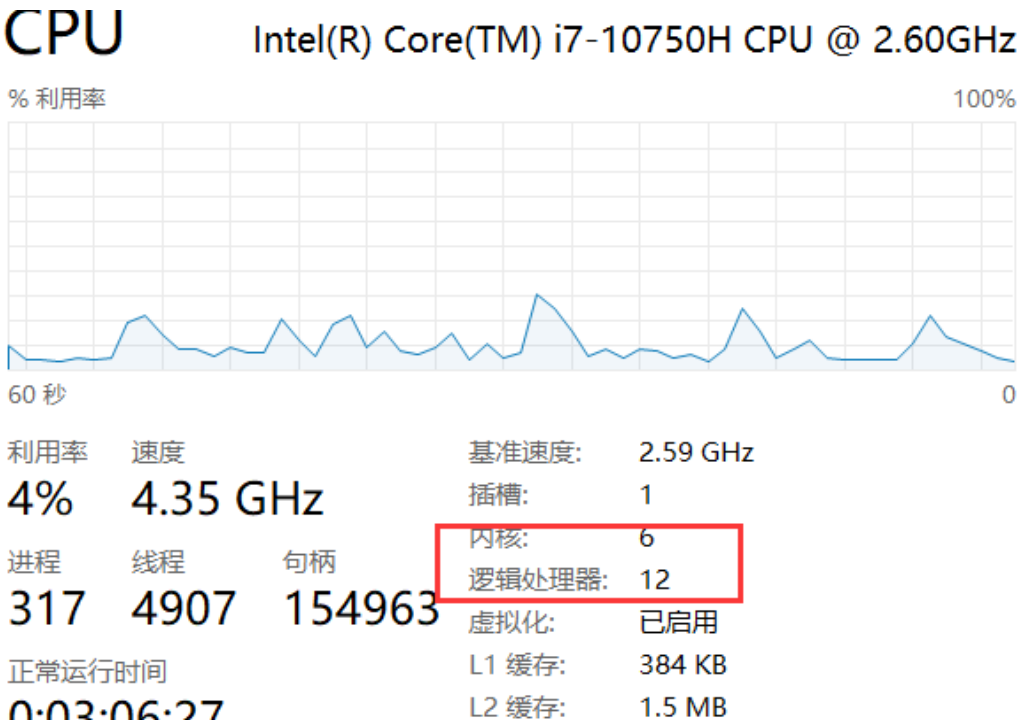
```
find_package(CUDA REQUIRED)
add_executable(cuda_examples main.cu)
```

这里以老师 PPT 上的乘积求和程序为例，输出结果如下，也验证了 CUDA 配置成功：

其它设备参数

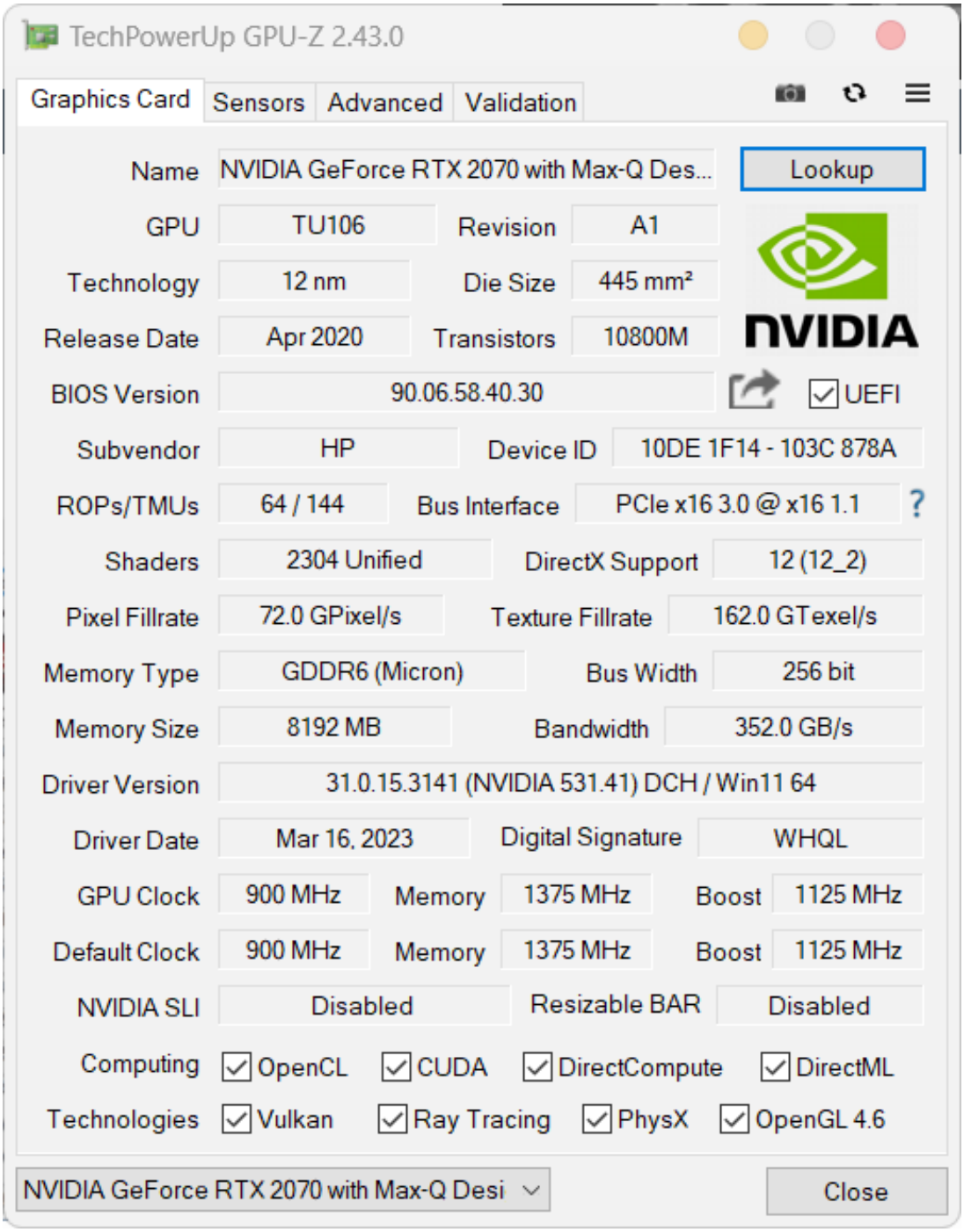
CPU 参数

通过任务管理器来查看， 我的电脑是 6 核 12 线程的：



GPU 参数

利用课程群中提供的 GPU-Z 查看 GPU 参数如下：



说明

实际上，在先前配置 CUDA 时候通过 `deviceQuery.exe` 也可以来获取相关参数 。