上文

上午

1h GPU 介绍

2h CPU架构和GPU架构

CUDA架构-编程模型，存储模型

0.5h CUDA开发环境搭建和工具配置

下午 承上启下，印证前文，铺垫后文，

CUDA 编程 + 上机实验

CUDA 编程技巧及注意问题 + 上机实验

自我介绍

2006承接OpenGL编程项目，涉及GLSL/CG等早期GPU编程；

2010涉足CUDA编程；

2014加入周斌老师科研和培训团队，任职助理研究员和CUDA培训师。

培训风格：周老师之中英混合、孙鑫之边讲边练、NEHE之增量学习、GIT之版本递进；

新事物，不是为了废掉旧的，而是为了成全旧的，使旧的变得完全。

回顾上文

大数据通用计算的需求，GPU基础架构高性能的天然能力。

前几代GPU技术的发展。共同造就了CUDA。

CUDA语言继承C/C++，有细微延伸和扩展。

C程举例

例1：计算点到多边形的距离，算法公式：d=sqrt( (x1-x2)^2 + (y1-y2)^2 )；

移植CUDA——新的编程模型

移植预备，将for循环封装成嵌套函数；

规范化，根据新模型规范微调程序。

0.关键词：函数类型、变量类型

1.线程配置与索引；

2.显存分配；

3.函数声明与调用；

4.参数输入与结果输出；

新的存储模型

上一例子用到，global memory和register memory，常用的还有constant memory、shared memory以及texture memory，下面着重演示constant memory、shared memory的用法。

例1升级，被共享的点坐标先后存到constant memory、shared memory。

技巧Tips

1.线程配置

A．线程数目不是32的倍数怎么办？

B．线程数目阈值检测？最高支持T级别( 1<<(9+16+16) ~= 2^40 )？一维grid线程数目超过阈值怎么办？

2.如何精确测时？

A．调用helper\_timer.h

B．注意Synchronize

C．线程数目(问题规模)多大时，GPU发挥的性能赶超CPU？

3.移植三步曲

循环变嵌套函数、索引数组变量化、新模型改写；

4.其它

A．结果验证

B．错误定位

C．driver编写

预告下文

程序延伸，在例1的基础上求最短距离（reduction）？

新例子——例2，矩阵相乘。C算法，CUDA算法初步，C改进算法-block。

下理

明天

CUDA 进阶

CUDA 扩展，分析、调试

CUDA 基本优化

后天

CUDA 深入优化

考试