GPGPU/CUDA/C Workshop 2012

Day-1:
GPGPU/CUDA/C and WSU

Presenter(s):
Abu Asaduzzaman
Nasrin Sultana

Wichita State University
July 10, 2012
Outline

- Introduction to the Workshop
- Computing: past, present, and future
- GPGPU/CUDA/C Technology and WSU
- Parallel Computing (by Nasrin)
- Practice:
  - C, C threads
  - OpenMP/C, Open MPI/C
  - Open MPI/C (SMP, MPI)

QUESTIONS?
Any time, please.
Workshop Introduction

- (Workshop) Objectives
  - To become a moderate to advanced level CUDA/C programmer
  - To prepare pedagogy for future CSE courses
  - To develop parallel computing research initiatives

- Methodologies
  - Discuss, study (book?), and practice
  - CUDA Educator from Nvidia

- (Workshop) Outcomes
  - Understand the needs and benefits of parallel programming
  - Write program in C, C thread, OpenMP/C, and Open MPI/C
  - Understand NVIDIA GPU/CUDA technology
  - Develop programs in CUDA/C for GPGPUs
# Workshop Introduction (2)

## Workshop Schedule

<table>
<thead>
<tr>
<th>Date</th>
<th>9:30 am to 12:00 noon session</th>
<th>1:30 pm to 4:00 pm session</th>
</tr>
</thead>
<tbody>
<tr>
<td>July/10/2012</td>
<td>• Introduction to the Workshop&lt;br&gt;• Computing: past, present, and future&lt;br&gt;• GPGPU/CUDA/C and WSU&lt;br&gt;• Parallel Computing (by Nasrin)</td>
<td>• Practice&lt;br&gt;• C, C threads&lt;br&gt;• Open MP/MPI&lt;br&gt;• Open MPI (SMP, MPI)</td>
</tr>
<tr>
<td>Tuesday</td>
<td>(Asaduzzaman/WSU)</td>
<td></td>
</tr>
<tr>
<td>(Asaduzzaman/WSU)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
(Workshop) Presenters

- Abu Asaduzzaman (Dr. Zaman)
  - Ast. Prof., EECS Dept. at WSU
  - Teaching computer systems and architecture courses
  - Research: parallel computing systems

- Mark Ebersole
  - NVIDIA CUDA Educator and parallel programming expert
  - Ten years of low-level systems programming experience
  - Working on device drivers and hardware diagnostic

- Chok Yip, EECS MS Student, WSU
- Nasrin Sultana, EECS MS Student, WSU
(Workshop) Participants

- Abu Asaduzzaman, EECS Ast. Prof., WSU
- Chok M. Yip, EECS MS Student, WSU
- Danny V. Nguyen, EECS BS Student, WSU
- Deepak A. Subbarayappa, Ph.D. (Math), WSU
- Kh. Nobin Rahman, EECS MS Student, WSU
- Merl Chrishanthas, EECS BS Student, WSU
- Nasrin Sultana, EECS MS Student, WSU
- Sreenivas Rao Gajula, EECS MS Student, WSU
Workshop Introduction (5)

Keywords

- **OpenMP (Open Multiprocessing)**
  - An API that supports multi-platform shared memory multiprocessing programming in C, C++, and Fortran.

- **Open MPI (Message Passing Interface)**
  - Supports shared memory as well as distributed systems.

- **GPU (Graphics Processing Unit)**
  - Aka, visual processing unit (VPU)
  - A specialized chip designed to rapidly manipulate memory to accelerate the building of graphic images

- **GPGPU**
  - General Purpose GPU (GPGPU, GPGP, or GP²U)
Keywords (cont’d)

- **CUDA (Compute Unified Device Architecture)**
  - A parallel computing architecture for graphics processing
  - CUDA is the computing engine in Nvidia GPUs that is accessible to software developers through variants of industry standard programming languages

- **OpenCL (Open Computing Language)**
  - A framework for writing programs that execute across heterogeneous platforms consisting of CPUs, GPUs, and other processors. It has been adopted by Intel, AMD, Nvidia, and ARM. OpenCL gives any application access to the GPU for non-graphical computing. Thus, OpenCL extends the power of the GPU beyond graphics.
Keywords (cont’d)

- **nvcc**
  - CUDA compiler
  - CPU code and GPU code

- **Kernel, Block, Grid, Thread, DIM**
  - Host → kernel → parallel Blocks
GPGPU/CUDA/C Workshop 2012

Outline

- Introduction to the Workshop
- Computing: past, present, and future
- GPGPU/CUDA/C Technology and WSU
- Parallel Computing (by Nasrin)

Practice:
- C, C threads
- OpenMP/C, Open MPI/C
- Open MPI/C (SMP, MPI)

Questions?
Any time, please.
Computing: past, present, future

- **CPU performance**
  - First PC in 1980s; 1MHz
  - Desktops in 2010s; 1GHz to 4GHz
  - Power consumption, heat dissipation, limit to transistor size

- **Supercomputer performance**
  - Tens to hundreds of processor cores; Roadrunner, BlueGene/L

- **Hyper-threading**
  - Intel Pentium 4
Computing: past, present, future (2)

- **PC/Workstation performance**
  - Add more processing cores
  - Multithreading, SMT

- **Year 2000 and beyond**
  - Multicore CPU (2-core netbook, 8-core workstations)
  - Parallel computing on mobile devices
  - “Command prompt is out; multithreaded GUI is in.”

- **Parallel/Concurrent processing**
  - Add more processing cores
  - Multithreading, hyper-threading, SMT
Parallel Processing – It is not fun!

- Paying the lunch bill together

<table>
<thead>
<tr>
<th>Friend</th>
<th>Before Eating</th>
<th>Total Bill</th>
<th>Return</th>
<th>Tip</th>
<th>After Paying</th>
<th>Total Spent</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>$10</td>
<td>$25</td>
<td>$5</td>
<td>$2</td>
<td>$1</td>
<td>$9</td>
</tr>
<tr>
<td>B</td>
<td>$10</td>
<td>$25</td>
<td>$5</td>
<td>$2</td>
<td>$1</td>
<td>$9</td>
</tr>
<tr>
<td>C</td>
<td>$10</td>
<td>$25</td>
<td>$5</td>
<td>$2</td>
<td>$1</td>
<td>$9</td>
</tr>
<tr>
<td>Total</td>
<td>$30</td>
<td>$75</td>
<td>$15</td>
<td>$6</td>
<td>$3</td>
<td>$27</td>
</tr>
</tbody>
</table>

- Started with $30; spent $29 ($27 + $2)
- Where did $1 go?
Computing: past, present, future (4)

- **Past (before year 2000)**
  - Single-core systems, sequential programming
  - Parallel computing on sequential systems
  - C

- **Present (2000 to present)**
  - Mostly multi-core systems, parallel programming
  - Parallel and distributed systems, GPU technology
  - Open MP/MPI, CUDA

- **Future?**
  - Cloud computing, networking, virtualization
  - GPGPU/CUDA Technology
GPGPU/CUDA/C and WSU

- **Nvidia market growth**
  - In 2010, Nvidia's year-to-year growth was 41.9%.
  - Q1 '11 was not good for Nvidia (AMD 15.4%, Intel 9.7%, but Nvidia -28.4%) due to GPU overheating debacle, issues with Apple. In August 2011, Nvidia predicted the growth of its revenues would be 4% to 6%.

- **Nvidia-Academia partnership**
  - Most top ranked universities including MIT and Stanford
  - CUDA Teaching Centers
  - CUDA Research Centers
GPGPU/CUDA/C and WSU (2)

NVIDIA's Academic Partnership

University of Southern California
University of Tennessee
University of Texas, Austin
University of Texas, Dallas
University of the Pacific
University of Toledo
University of Utah
University of Virginia
University of Washington
University of Wisconsin-Madison
University of Wyoming
Valdosta State University
Valparaiso University
Virginia Tech
Wake Forest University
Washington University in St. Louis
Wayne State University
Wentworth Institute of Technology
Western Michigan University
Wichita State University

Wichita State University
1845 Fairmount St #JB-253, Wichita, Kansas 67260-0083

Modern Computer Architecture and Programming
Course: CS 794
Contact: Abu Asaduzzaman

Zoom Here - more info
GPGPU/CUDA/C and WSU (3)
GPGPU/CUDA/C Workshop at WSU

Outline

- Introduction to the Workshop
- Computing: past, present, and future
- GPGPU/CUDA/C Technology and WSU
- Parallel Computing (by Nasrin)
- Practice:
  - C, C threads
  - OpenMP/C, Open MPI/C
  - Open MPI/C (SMP, MPI)

Questions?
Any time, please.
Outline

- Introduction to the Workshop
- Computing: past, present, and future
- GPGPU/CUDA/C Technology and WSU
- Parallel Computing (by Nasrin)

Practice:
- C, C threads
- OpenMP/C, Open MPI/C
- Open MPI/C (SMP, MPI)

Questions?
Any time, please.
Practice

Hello, WSU!

- C
- C thread
- Open MP/C
- Open MPI/C (SMP)
- Open MPI/C (MPI)
Practice (2)

Hello, WSU!

- C
  
  [0] Hi, WSU!
  [1] Hello, WSU!
  [2] Namaste, WSU!
  [3] Selamat pagi, WSU!

- Sample Codes
  
  - drzaman@kirk:~/CUDAWorkshop2012/Day_1/Hello$ pwd
  - /usr/users/User11/drzaman/CUDAWorkshop2012/Day_1/Hello
Matrix Multiplication

$$\begin{pmatrix}
a_{0,0} & a_{0,1} \\
a_{1,0} & a_{1,1} \\
a_{2,0} & a_{2,1} \\
a_{3,0} & a_{3,1}
\end{pmatrix} \times \begin{pmatrix}
b_{0,0} & b_{0,1} & b_{0,2} \\
b_{1,0} & b_{1,1} & b_{1,2}
\end{pmatrix}$$

Result

$$c_{0,0} = a_{0,0} b_{0,0} + a_{0,1} b_{1,0} \quad c_{0,1} = a_{0,0} b_{0,1} + a_{0,1} b_{1,1} \quad c_{0,2} = a_{0,0} b_{0,2} + a_{0,1} b_{1,2}$$

$$c_{1,0} = a_{1,0} b_{0,0} + a_{1,1} b_{1,0} \quad c_{1,1} = a_{1,0} b_{0,1} + a_{1,1} b_{1,1} \quad c_{1,2} = a_{1,0} b_{0,2} + a_{1,1} b_{1,2}$$

$$c_{2,0} = a_{2,0} b_{0,0} + a_{2,1} b_{1,0} \quad c_{2,1} = a_{2,0} b_{0,1} + a_{2,1} b_{1,1} \quad c_{2,2} = a_{2,0} b_{0,2} + a_{2,1} b_{1,2}$$

$$c_{3,0} = a_{3,0} b_{0,0} + a_{3,1} b_{1,0} \quad c_{3,1} = a_{3,0} b_{0,1} + a_{3,1} b_{1,1} \quad c_{3,2} = a_{3,0} b_{0,2} + a_{3,1} b_{1,2}$$

Dr. Zaman; WSU-5261
Practice (4)

Matrix Multiplication: \( C[r, c] = A[r, ?] \times B[?, c] \)

```c
C

int MatrixMultiplication(int m1[][2], int m4[][3], int m5[][3], int r1, int r4, int c4) {
    int i=0, j=0, k=0;

    for(i=0; i<r1; i++)
        for(j=0; j<c4; j++) {
            m5[i][j] = 0;
            for(k=0; k<r4; k++)
                m5[i][j] += m1[i][k] * m4[k][j];
        } /* end for… j */

    return (1);
} /* end MatrixMultiplication */
```
Matrix Multiplication: $C[r, c] = A[r, ?] \times B[?, c]$
Practice (6)

Matrix Multiplication: $C[r, c] = A[r, ?] * B[?, c]$

- OpenMP/C
Matrix Multiplication: $C[r, c] = A[r, ?] \times B[?, c]$

- Open MPI/C (SMP)
Matrix Multiplication: $C[r, c] = A[r, ?] \times B[?, c]$

- Open MPI/C (MPI)
Conclusions

Following topics are covered is Day-1:
- Overview of the GPGPU/CUDA/C Workshop
- Brief discussion on parallel computing
- Parallel computing and WSU
- Programming in C, C thread, OpenMP/C, Open MPI/C
- Performance analysis of various solutions

Topics for Day-2:
- Brief history of GPGPU and CUDA
- CUDA Arch and CUDA/C Programming for GPGPU
Questions?

☐ Any questions, comments, or suggestions?
Thank you.

Please send your feedback to:
Abu Asaduzzaman (Dr. Zaman)
Tel: +1-316-978-5261
E-mail: Abu.Asaduzzaman@wichita.edu