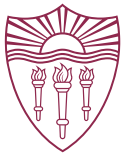


EE/CSCI 451: Parallel and Distributed Computation

Lecture #1

8/18/2020

Viktor Prasanna

prasanna@usc.edu

ceng.usc.edu/~prasanna

University of Southern California

USC University of Southern California

Outline

- Course information
- Parallelism everywhere, technology trends
- Data science and big data
- Course outline

Policies and Procedures

- All classes are recorded and accessible on Blackboard
- Recordings should be used appropriately (https://policy.usc.edu/scampus-part-c/)
- Specify your current time zone (and your background info.)
  - Please fill the Google form: https://forms.gle/yqxoUa7NC7bMq5yv6

Course Info. (1)

- When & Where
- Lecture: Tuesday, Thursday 3:30 – 4:50 PM, Online at: https://usc.zoom.us/j/92090885346?pwd=Ty9PZmFxL2dPVDR3ekNZNmdnVFJ0QT09
  - Meeting ID: 920 9088 5346
- Lab: Friday, 3:30 – 4:50 PM, Online at: https://usc.zoom.us/j/93866326251
  - Meeting ID: 938 6632 6251
- Office Hours
  - Tuesday 11:00-12:00 Noon
  - Thursday 10:00-11:00 AM
  - Meeting ID: usc.zoom.us/my/prasanna.zoom

Course Info. (2)

- TA & TA's office hours
  - Meng Yuan
    - Email: ymeng643@usc.edu
    - Office hours:
      - Friday 11:00 AM-1:00 PM or By appointment
      - Meet Yuan at: https://usc.zoom.us/j/8629150353
        - Meeting ID: 862 915 0353

Course Info. (3)

- **Prerequisite:** Some exposure to High level programming
- Textbook: Introduction to Parallel Computing (2nd Ed.), Grama, Gupta, Karypis, Kumar, Addison-Wesley
  - Options to buy or rent
    - Amazon: https://www.amazon.com/Introduction-Parallel-Computing-Ananth-Grama/dp/0201648652

- Reference book: Introduction to Parallel Computing, Zbigniew Czech, Cambridge
  - Free online access
    - Cambridge: https://www.cambridge.org/core/books/introduction-to-parallel-computing/F2170BB15F769C874CD62B3DB5255080

- Hands-on programming exercises
- Students are responsible for understanding all the materials covered in the class (only)

Course Info. (4)

- Course Objectives
  - Understand the key architectural concepts of multi-core, many core and GPU platforms for parallel programming
  - Develop simple parallel algorithms to solve computational problems
  - Implement key algorithms in the field on multi-core, many-core and GPU platforms
  - Understand and determine the computational complexity of simple parallel algorithms
  - Write parallel programs using message passing and shared memory paradigms

Course Info. (5)

- Course Objectives (cont.)
  - Select an appropriate basic data structure (e.g., arrays) and access methods (e.g., pointers) to optimize performance
  - Understand communication and coordination issues in parallel computing
  - Understand basic principles of Cloud computing and Data Science processing
  - Perform Data Science analytics using distributed computing frameworks

Course Info. (6)

- Grading
  - Homework 10%
    - Homeworks must be done independently
    - **10% late penalty per day** will be assessed with no credit received after the third day
  - Programming Assignments 10%
  - Course Project 15%
  - Midterm I (Sept 25 in lab session, 2 hours) 20%
  - Midterm II (Oct 23 in lab session, 2 hours) 20%
  - Final Exam 25%

Course Info. (7)

- Class Participation
  - Attend the class
  - Visit during office hours
  - Participate in online discussion forum
  - Asking or answering questions

Course Info. (8)

- Academic Integrity
  - Cheating will not be tolerated
  - Grade of F will be assigned
  - Cheaters will be reported to USC Student Judicial Affairs and Community Standards (SJACS)

Course Info. (9)

- Communicating with me
  - Visit during office hours via zoom
  - Via email prasanna@usc.edu
    - Subject field: EE451

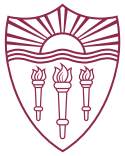
Course Info. (10)

- Lab
  - Time: 3:30 – 4:50 PM, Friday
  - Discussion of material covered in the lecture
  - Homework
  - Programming languages
  - Programming assistance
  - Course project discussion
  - Midterms (Sep 25 and Oct 23 in lab session, 330-530pm)
  - Course project presentation

Course Info. (11)

- Piazza
  - EE/CSCI 451 Parallel and Distributed Computation Fall 2020
  - Enroll via piazza.com/usc/fall2020/eecsci451
- Lecture notes (ppt) will be uploaded to Piazza one hour before the lecture
- Homework assignments will be uploaded to Piazza
- Please upload finished homework on Blackboard

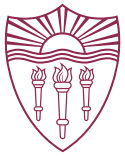
Programming Languages and Platforms



- Lower level languages (?)
  - Pthreads
  - MPI
  - OpenMP
  - CUDA/OpenCL
- Platforms
  - Intel/AMD Multicore
  - Nvidia GPU
  - Microsoft Azure Cloud
  - Amazon EC2 Cloud

- Domain Specific Languages
  - Giraph (Graph Processing)
  - TensorFlow (Machine Learning)
  - PyTorch (Machine Learning)
- High Level Languages
  - MapReduce
  - Hadoop
  - Spark

Sample Programming Assignments



- Dense and sparse matrix computations on multi-core and GPU platforms
- Parallel sorting on multi-core
- Real-time rendering on GPU
- Social Network analytics using Hadoop
- Parallelizing signal processing kernels (FFT, dense algebra,..)
- Graph embedding using graph convolutional network on GPU
- …

Course Project (1)

- Large software project
  - Scientific computing
  - Graph analytics
  - Data science
  - …
- Sample course projects
  - Multi-core implementation of data plane kernels for software defined networking (e.g., traffic classification, packet classification, etc.)
  - Accelerating Deep Neural Networks (CNN, LSTM, etc; Inference, Training, etc.) using GPU
  - Big data analytics using Spark (e.g., graph analysis, log processing, etc.)
  - …
- Students can work in teams

Course Project (2)

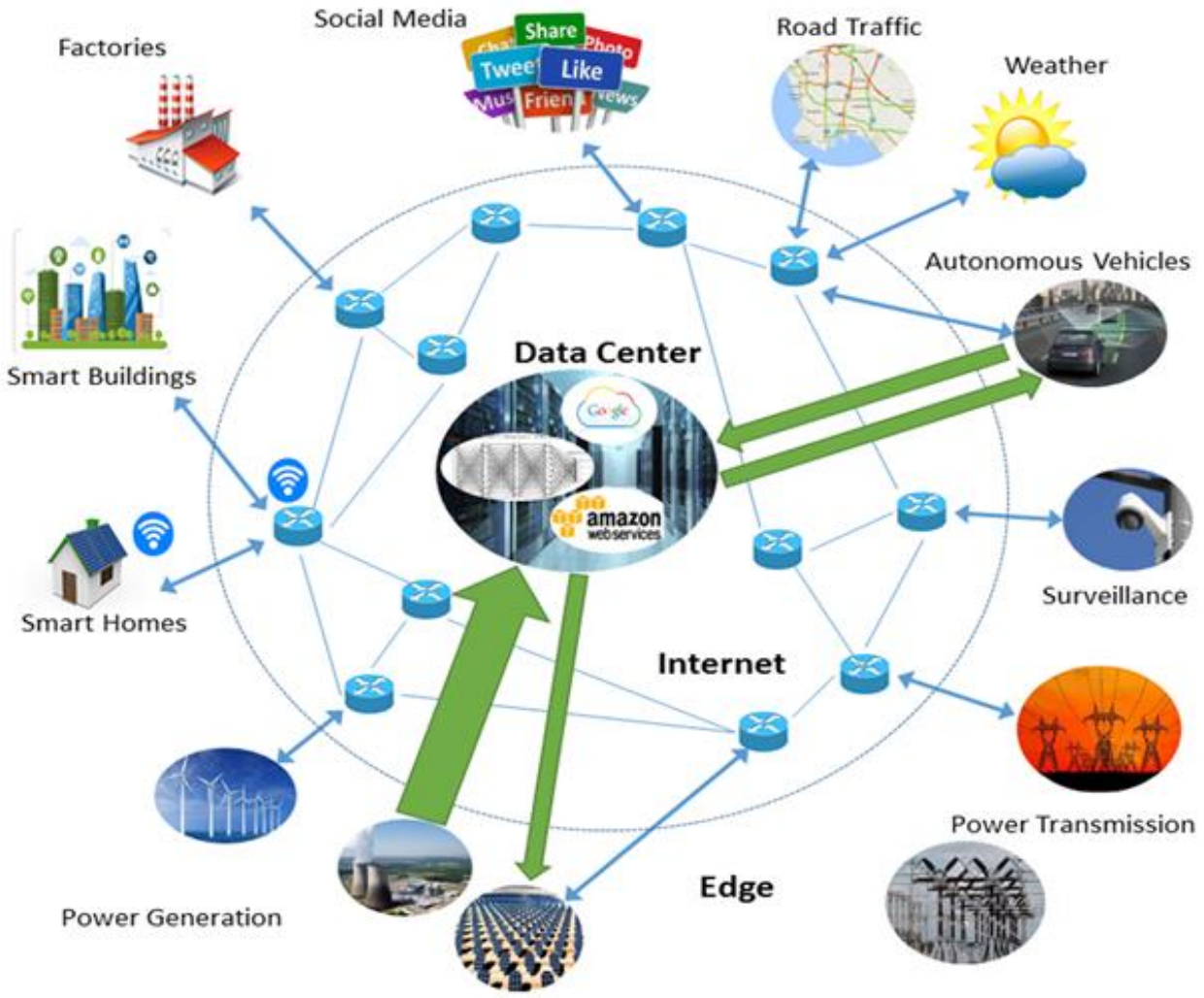
- Project timeline
  - Week 5-8: Identify team members and project topic
  - Week 9: Project proposal due
  - Weeks 12-13: Student Project Presentations
  - Week 13: Project final report due (Sunday Midnight AOE)

- Grading breakdown for the course project
  - Project Proposal: 25%
  - Project presentation: 25%
  - Project final report: 50%

Jobs

- Data Centers, HPC Centers, Networking, Communications equipment vendors, Data Scientists…
- Opportunities at
  - IBM
  - Nvidia
  - Intel
  - AMD
  - Microsoft
  - Amazon
  - Facebook
  - Google
  - National Labs
  - Electronic Arts
  - Juniper
  - Cisco
  - Xilinx
  - Micron

Parallelism Everywhere (0)



(Parallel) Computing Platforms

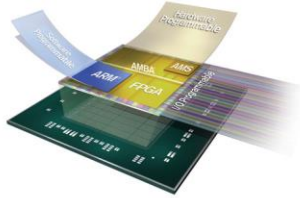
- Edge Devices

- Multi-core processors

- Graph Processing Units

- FPGAs

- System-on-Chip



- Supercomputers

Parallelism Everywhere (1)

- Cellphone processors

— Apple A13
  - iPhone 11, 11 Pro/Pro Max, SE (2nd generation)
  - Up to 2.65 GHz ARMv8.4-A six-core CPU
  - Semiconductor: 7nm (N7 Pro)
  - L1 Cache: 128KB (Data)/128KB (Instruction)
  - L2 Cache: 8 MB
  - Memory Technology: LPDDR4X

— Snapdragon 865 (Qualcomm)
  - Galaxy S20/S20+/S20 Ultra
  - Up to 2.84 GHz eight-core ARM cortex-A77/A55
  - Semiconductor: 7nm N7P
  - L1 Cache: 256KB/512KB
  - L2 Cache: 1.8 MB
  - Memory Technology: LPDDR4X-2133

Parallelism Everywhere (2)

- Multi-core processors

— Intel

  - Ex. Intel Xeon Platinum 9282
    - Cores: 56
    - Threads: 112
    - Frequency: 2.6 GHz
    - L1 Cache: 1.75 MB (Data)/1.75 MB (Instruction)
    - L2 Cache: 56 MB
    - L3 Cache: 77 MB
    - DDR4: 2 TB
    - Maximum memory bandwidth: 262 GB/s
    - Thermal Design Power: 400 W

Parallelism Everywhere (3)

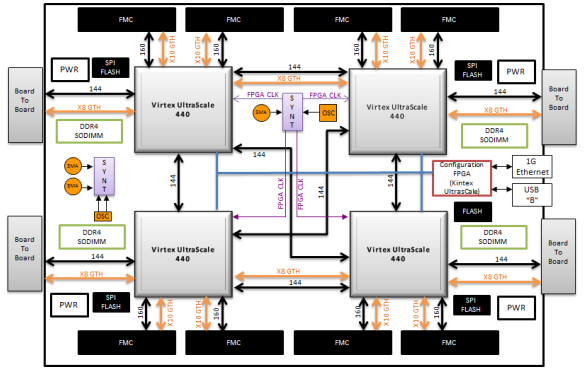
- FPGA family – Xilinx

— Virtex Ultrascale+ family

- 9 million logic cells
- 16 nm FinFET
- Up to 500Mb of on-chip memory integration
- Up to 128-33G transceivers deliver 8.4 Tb of serial bandwidth
- 460GB/s HBM bandwidth, and 2,666 Mb/s DDR4 in a mid-speed grade

— For high-performance computing

- 5.5 M logic cells for massively parallel processing
- DSP48E slices for fixed- and floating-point acceleration
- 150 Gb/s chip-to-chip interconnect

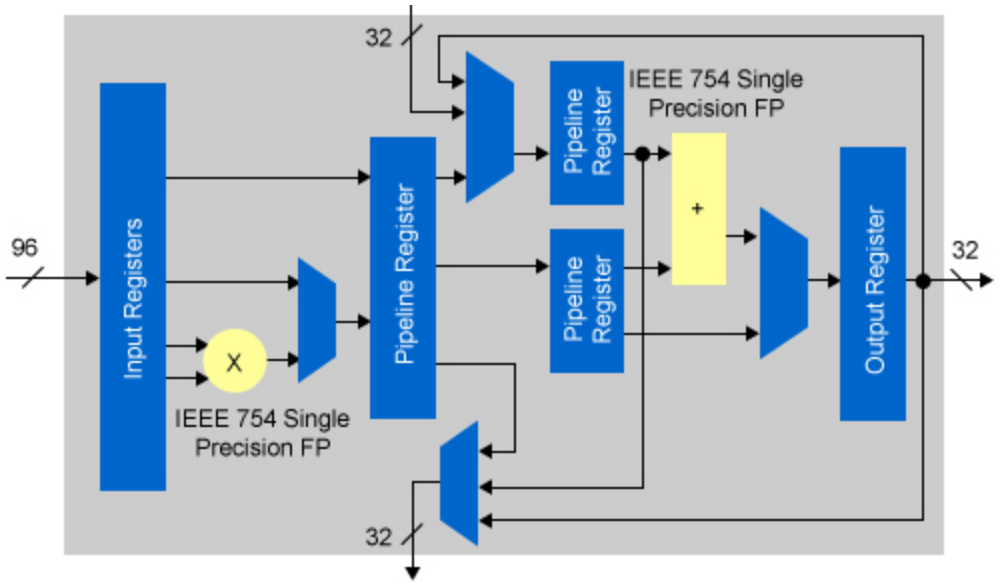


Parallelism Everywhere (4)

- Programmable SoC family – Zynq UltraScale+ MPSoC
  - Efficient ARM + FPGA
  - Processing System
    - Quad-core ARM Cortex-A53 processor
    - Running at 1 GHz
    - 71 dedicated DDR controller I/Os
    - 54 dedicated memory/peripheral I/Os
    - L1 Cache – 32KB/32KB (per Core)
    - L2 Cache – 512KB Unified
  - Programmable Logic
    - 16 nm Xilinx FPGA fabric
    - Virtex UltraScale+ or Kintex UltraScale+ based
    - Up to 16 12.5Gbps high-speed transceivers
    - Up to 150 1.8V high performance I/Os

Parallelism Everywhere (5)

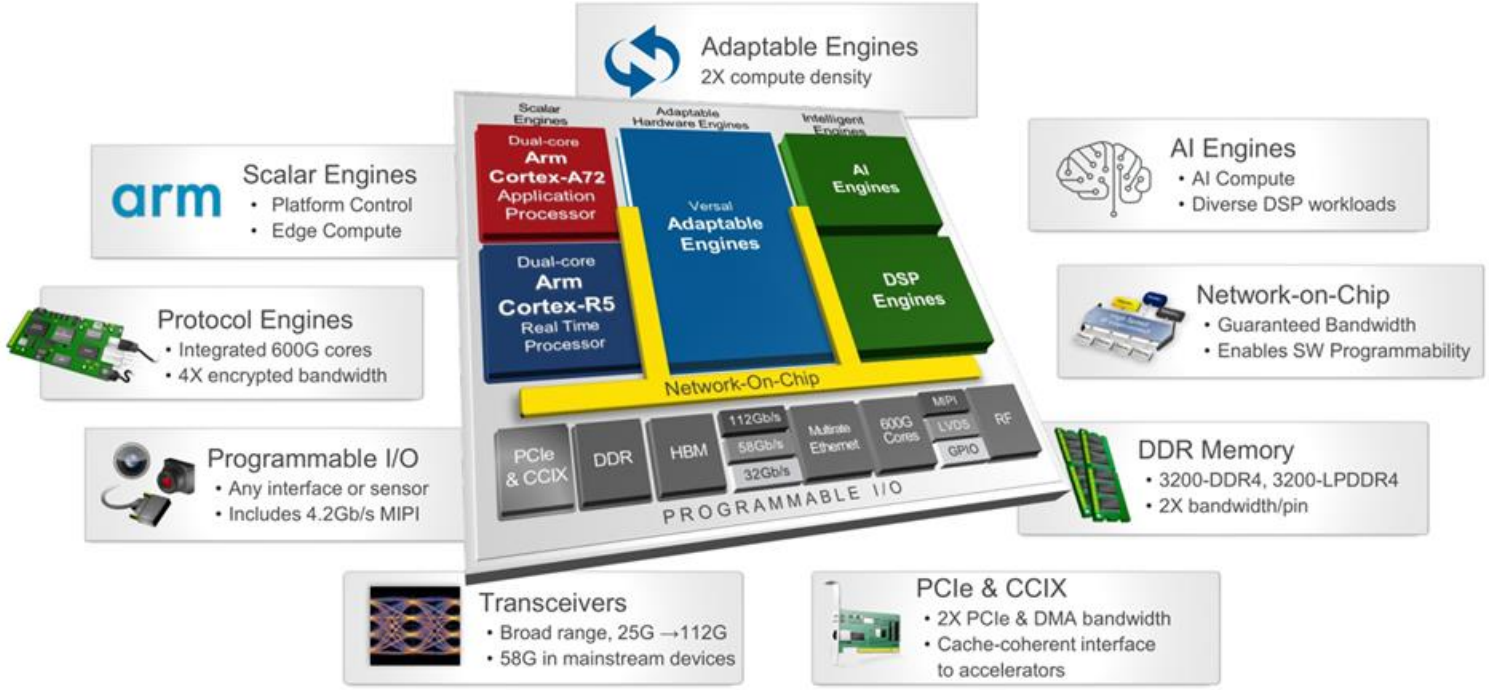
- Floating-point FPGA – Stratix 10
- First delivery of hardened floating-point operators within a DSP block
- Up to 10 tera floating point operations per second (TFLOPS)
- Applications
  - Industrial video
  - Broadcast systems
  - Wireless systems
  - Medical imaging
  - Military radar
  - High-performance computing



Stratix 10 FPGA and SoC Variable-Precision DSP Block, Floating-Point Mode

Parallelism Everywhere (6)

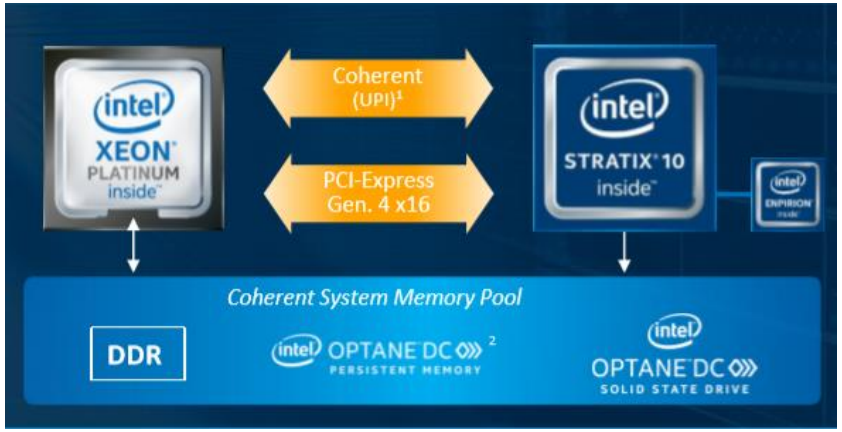
- ACAP
  - Vector and scalar processing unit tightly coupled with programmable logics, tied with high bandwidth Network-on-Chip
- Arm Cortex processors
- DSP Engines
- High-bandwidthnnetwork-on-chip
- AI Engines
  - Array of VLIW SIMD vector processors programmable using C/C++

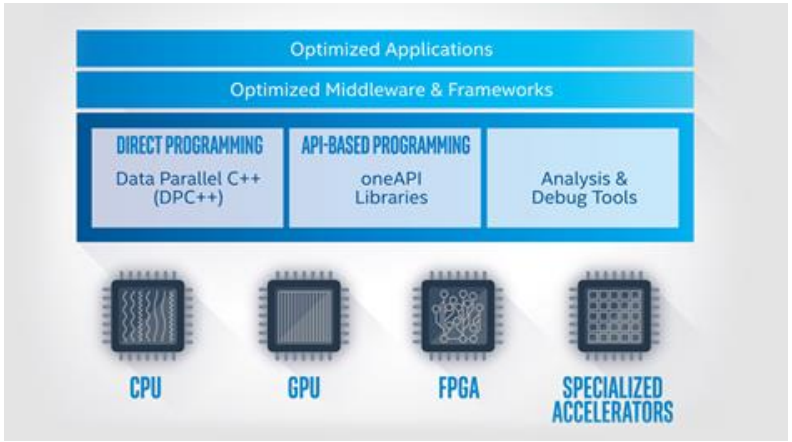


Vissers, Kees. "Versal: The Xilinx Adaptive Compute Acceleration Platform (ACAP)." *Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. ACM, 2019.

Parallelism Everywhere (7)

- Heterogeneous Computing Platform
  - CPU+FPGA (e.g. Intel Xeon Server processor + Stratix 10 data-center FPGA)
  - Cache Coherent Interconnect (e.g. UPI, CXL)
  - Unified Programming Interface: OneAPI
    - Deploy applications across heterogeneous architectures: CPUs, GPUs and FPGAs
    - Data Parallel C++ Compiler and libraries



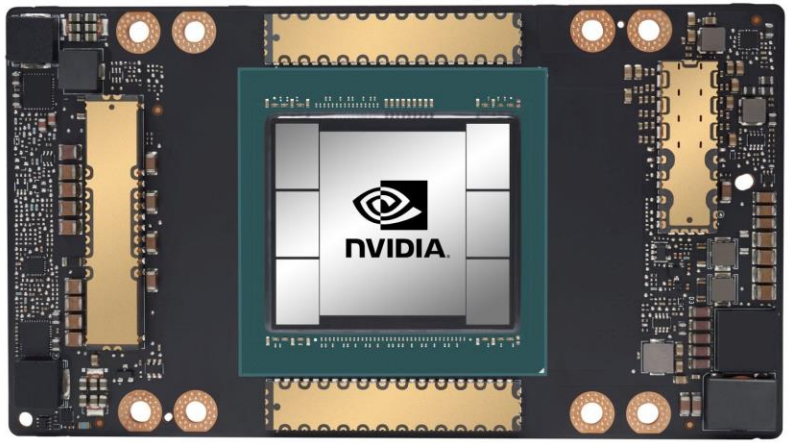


Parallelism Everywhere (8)

- GPU family – Nvidia

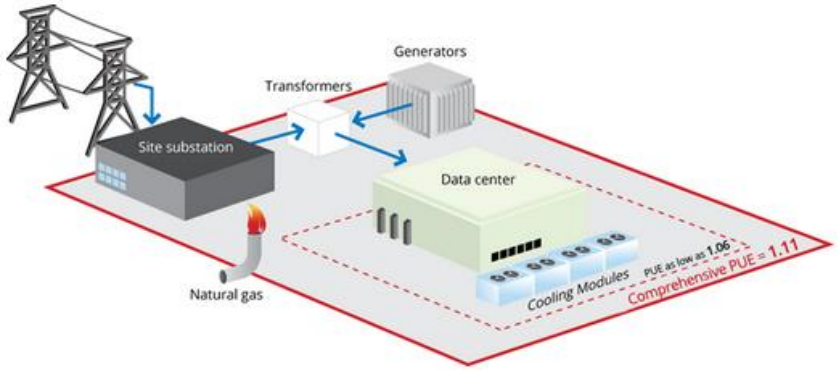
— A100 Tensor Core GPU

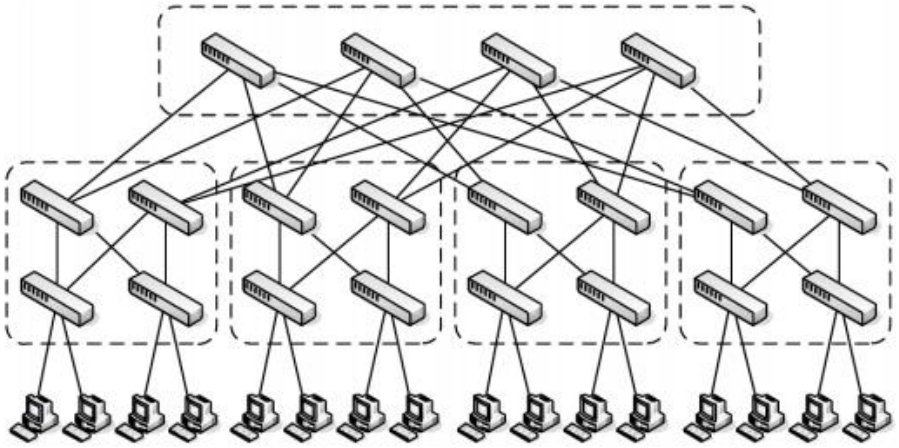
  - 108 Streaming Multiprocessors
  - 40 MB L2 cache
  - 40 GB high bandwidth memory
  - 1.6 TB/s peak bandwidth
  - 600 GB/s inter-GPU communication bandwidth
  - 54.2 billion transistors



Parallelism Everywhere (9)

- Data center
  - — Servers
  - — Storage
  - — Network devices
  - — Power distribution systems
  - — Cooling systems
  - — Ex. Google data centers
    - > 2.5 million servers
    - 13 in the US
    - 1 in South America
    - 5 in Europe
    - 2 in Asia





Parallelism Everywhere (10)

- Supercomputer
— Fastest high-performance system
— Used to solve problems which are
- too massive for standard computers
— Typically multi-core+GPU
— Ex. Columbia supercomputer (2004-2013)
  - 10,240 processors
  - 20 TB of memory
  - 440 TB of online storage
  - 10,000 TB of tape data storage

Parallelism Everywhere (11)

- Fugaku
  - The world's fastest supercomputer (as of Jun. 2020)
  - Location: RIKEN Center, Japan
    - Processor: A64FX 48C 2.2GHz
    - 7,299,072 cores
  - Power: 28.33 MW
  - Memory: 4.9 PB
  - Performance: 416 PFLOPS
    - Mega 10^6
    - Giga 10^9
    - Tera 10^{12}
    - Peta 10^{15}

Data Science and Big Data

- Big data is a term to describe the large volume of data – both structured and unstructured – that are too large or complex for traditional data processing applications to deal with
- Data mining huge amounts of data:
  - Google: **5.8 billion** searches per day
  - Facebook: **70 billion** photos are uploaded per year
- Data science refers to the techniques and systems to extract knowledge and insights from big data
- Key components
  - Modeling: Consumer behavior
  - Classification: Label consumers
  - Prediction: Energy consumed over next few hours
  - Optimization: Minimize total energy consumed
  - Data Mining: Fraud detection
  - Machine Learning: Image classification

Outline

- Course information
- Parallelism everywhere, technology trends
- Data science and big data
- Course outline

33

Course Outline (1)

	Topics/Lecture Activities	HW/Exam Due Dates
Week 1	Course overview (Aug. 18) Parallel computing architectures (Aug. 20)	
Week 2	Parallel computing architectures (Aug. 25) shared memory programming model (Aug. 27)	Homework 1 out
Week 3	Shared memory programming model and OpenMP (Sept. 1 and 3)	Homework 1 due Homework 2 out
Week 4	Message Passing programming model	Homework 2 due Homework 3 out
Week 5	Interconnection networks	Homework 3 due Homework 4 out
Week 6	Communication cost in parallel machines, message passing, routing in interconnection networks, Program and data mapping: graph embedding	Midterm 1 (Sept. 25) 2 hrs. Homework 4 due Homework 5 out

Course Outline (2)

	Topics/Daily Activities	HW/ Exams Due Dates
Week 7	Analytical Modeling of Parallel Systems Communication Primitives P1	Homework 5 due Homework 6 out
Week 8	Communication primitives P2 GPU Architecture CUDA P1	Homework 6 due Homework 7 out
Week 9	GPU Architecture CUDA P2 Parallel algorithm design, dependency graph	Homework 7 due Homework 8 out
Week 10	Graph partitioning, mapping, parallel algorithm models Map Reduce, Distributed Computing on Cloud	Midterm 2 (Oct 23) 2 hrs. Homework 8 due Homework 9 out
Week 11	Data Science, Parallel Graph Analytics & Parallel Sorting, Supercomputers (Oct. 27 and 29)	Homework 9 due Homework 10 out
Week 12	PRAM Part 1 & Part 2 (Nov. 3 and 5)	Homework 10 due Homework 11 out
Week 13	Project Presentations	Homework 11 due
FINALS WK	Final Exam	Nov 17 - Nov 24

Course Outline (3)

	Topics/Lab Activities	Readings and Homework
Week 1	Account setup and Lab overview (Aug 21)	Programming HW 1 out
Week 2	Pthreads Part1 (Aug 28)	Programming HW 1 due
Week 3	Pthreads Part2, Data Science Basics (Sep 4)	Programming HW 2 out
Week 4	OpenMP Part1 (Sep 11)	Programming HW 2 due
Week 5	OpenMP Part2 (Sep 18)	Programming HW 3 out
Week 6	Midterm 1 2 hours	Programming HW 3 due
Week 7	MPI (Oct 2)	Programming HW 4 out
Week 8	Project Discussion, CUDA (Oct 9)	Programming HW 4 due
Week 9	CUDA (Oct 16)	Programming HW 5 out
Week 10	Midterm 2 2 hours	Programming HW 5 due
Week 11	Spark, MapReduce (Oct 30)	Programming HW 6 out
Week 12	Course project presentation (Nov 6)	Programming HW 6 due
Week 13	Course project presentation (Nov 13)	

Acknowledgement

- Lecture notes compiled over the years
- My research team
- Course TA and mentors during Spring ‘14, ‘15, ’16, ‘17, '18, '19
- EE451 Students in Spring ‘14, ‘15, ‘16, ‘17, '18, '19
- Introduction to Parallel Computing (2nd Ed.), Vipin Kumar, Ananth Grama
- ……

Research

- Research Projects
  - High Performance Networking
  - Big Data Analysis
  - Social Network Analysis
  - Energy Efficient Computing
  - Accelerated Machine Learning
- Webpages
  - ceng.usc.edu/~prasanna/
  - http://fpga.usc.edu/
  - http://dslab.usc.edu/