Teaching Parallel Computing Concepts through Metaphors

1. FUN : LEARNING :: THIS PRESENTATION : ?
   a) Nice Try
   b) Coffee Break
   c) Naptime
   d) You of Course!

Julianna Eum
Computer Architecture and Security Technology Lab (CASTL)
Columbia University, New York, NY 10027
jme2134@columbia.edu

How to Increase Computing Power:

- Solution 1: Do ONE Thing Very Fast
  - Increasing Frequency = Faster Computation
  - Increasing Frequency = Increased Heat

- Solution 2: Do MANY Things Relatively Fast
  - Subsequent tasks do not wait to start

The Parallel Computing Landscape

- Smartphones, Tablets, Media Players
  - iPad run by Apple A5 chip
  - Dual-core A4 custom chip on iPhone and iPod Touch
  - Motorola Droid X2 run by a 1GHz multi-processor

- Xbox 360, Playstation 3 and Beyond
- Personal Computers

  Intel Atom, Core i3/5/7, Itanium, Pentium 4 & Xeon processors utilize "hyper-threading"

http://www.apple.com/ipad/specs/

http://www.motorola.com/Consumers/US-EN/Consumer-Product-and-Services/Mobile-Phones/ci.DROID-X2-by-MOTOROLA-US-EN.alt

Why Learn Parallel?

Physics → Mathematical Vector Operations → Virtual Reality

Single Intel Processor Core

Multiple Cores with Multithreading

=

Shippy, Dave and Mickie Phipps. "The Race for a New Game Machine: Creating the Chips Inside the Xbox 360 & the PlayStation 3." Citadel Press, Kensington Publishing Corp, 2009.

Teach Parallel Earlier

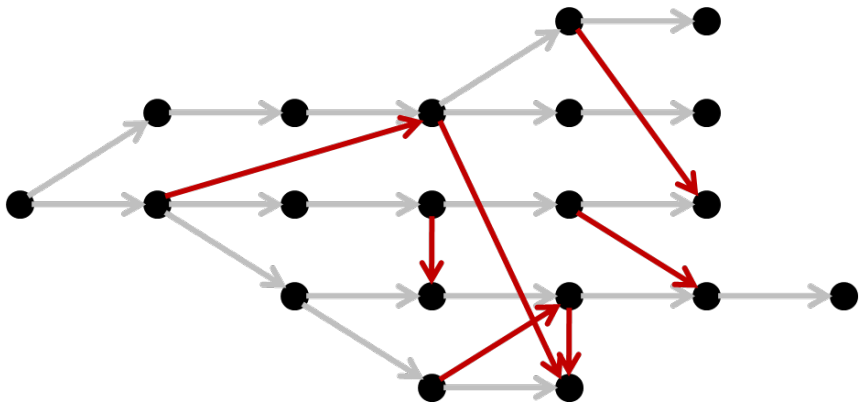
- Industry has overwhelmingly shifted to parallel computing while most CS and EE curricula are still adjusting.

- Students learn fastest while their brains are still plastic; most parallel concepts are taught after their paradigms have already formed.

How can Parallel Computing Concepts be Fun?

- Active learning increases attentiveness; games require such active participation
- Tap into students' existing knowledge
- Problem Solving and Competition
- Build a quick reference paradigm through games:

	3	4					9	
6					8	2	1	4
		1		2	7			
4		9		8	3	6	5	2
5	6						8	7
	2		5	6		4		1
			2	1		8		
8	5	7	9					6
	9					3	4	



Term Definitions

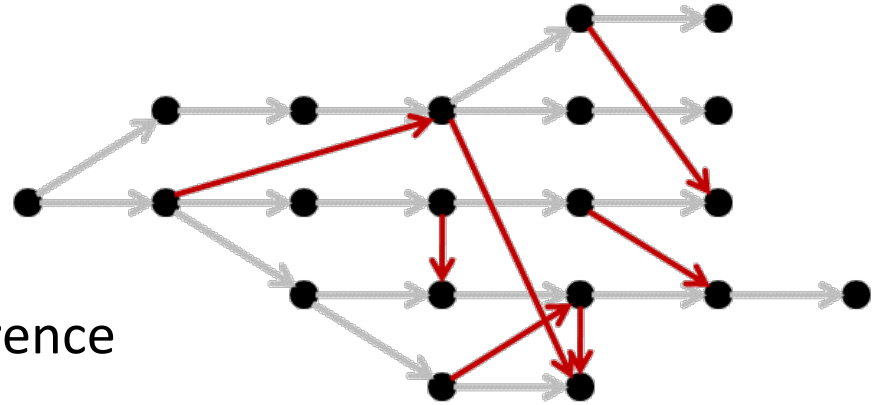
- **Decomposition**: break up the computation into tasks and sub-tasks
- **Assignment**: determine which processors get which tasks
- **Communication**: establish parameters and methods for processors to work together
  - **Shared Memory** ≈ Blackboard
  - **Message Passing** ≈ Getting "Assistance" on a Test

Parallel Software Concepts

- Students choose *Decomposition* & *Assignment*
  - Planning phase done prior to game time starting
  - 5 to a group avoids even choices
  - Consider load balancing
- *Communication* Pre-Determined:
  Shared Memory or *Message Passing*

Choose Your Path (CYP)

- Based on Choose Your Own Adventure
- Four Tasks:
  1. Write story sheets
  2. Link and join sheets
  3. Verify rule adherence
  4. Check & edit story coherence



- Four Parameters:
  1. 3 distinct branches each ≥ 5 sheets deep
  2. ≥ 1 sentences per sheet with ≥ 1 choice unless ending
  3. ≥ 7 joins equal or lower in level (no looping)
  4. Overall coherent narrative
- Grading on:

  0%: Meeting parameters (GO/NO GO - disqualified if not)

  50%: Time to completion

  50%: Story quality – subjective measure of content and coherence

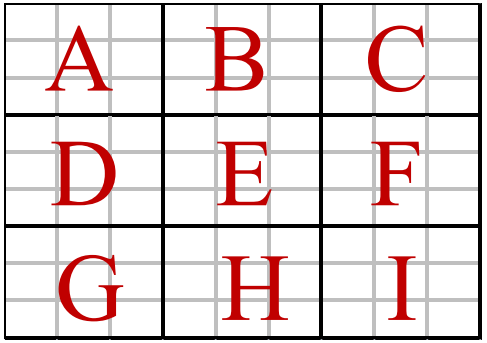
CYP Starting Point

- Subject: “A Day Off in NYC”
- Must Plan Task/Data Partitioning
  - 2 minutes for this prior to starting
  - May split up tasks or story sheets (data) among group but do not change tasks after starting – write names for assigned tasks
  - May create more sheets than the minimum for coherence but remember that sheet quantity is not a winning criteria
  - Groups are encouraged to draw a story graph in the space below:

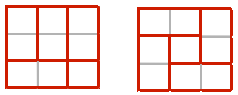
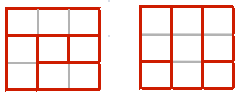
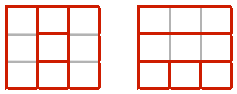
CYP Takeaways

- Concurrent Tasks are much faster when processed in parallel
- Load Balancing matters (prior planning)
- Poor locking can lead to overwriting data
  - Change end result
  - Invalidate work
- Synchronization is important

Sudoku



- Worldwide Familiarity
- Decomposition granularity at minigrid level
- Communication:
  - Shared Memory: Multiple Readers, One Writer
  - Message Passing: [*(x-coord, y-coord), number, confidence*]
- Scratch paper provided for students to execute their solving method
- Fastest group to correct solution wins (same initial board)

Decomposition	Analogy	Graphic
4x2 subsquares, 1x1 subsquare	Primarily Homogenous Processor	
1x3 subsquares, 2x2 subsquares, 2x1 subsquare	Heterogeneous Processor	
2x3 subsquares, 3x1 subsquare	Heterogeneous Processor	

Discussion and Questions

- Do you agree that these methods effectively teach the concepts enumerated? If not, how could they be adjusted?
- Could these methods be taught as early as middle school? Why/why not?
- Do the games go into too much or too little detail to provide effective learning platforms?

Future Applicability for You

- Contact Information:

Julianna Eum
Computer Architecture and Security Technology Lab (CASTL)
Columbia University
jme2134@columbia.edu

Simha Sethumadhavan
Assistant Professor of Computer Science
CASTL Director
Columbia University
simha@cs.columbia.edu
1.212.939.7062

- Please let us know if you would like to use either of the games and if we can help!