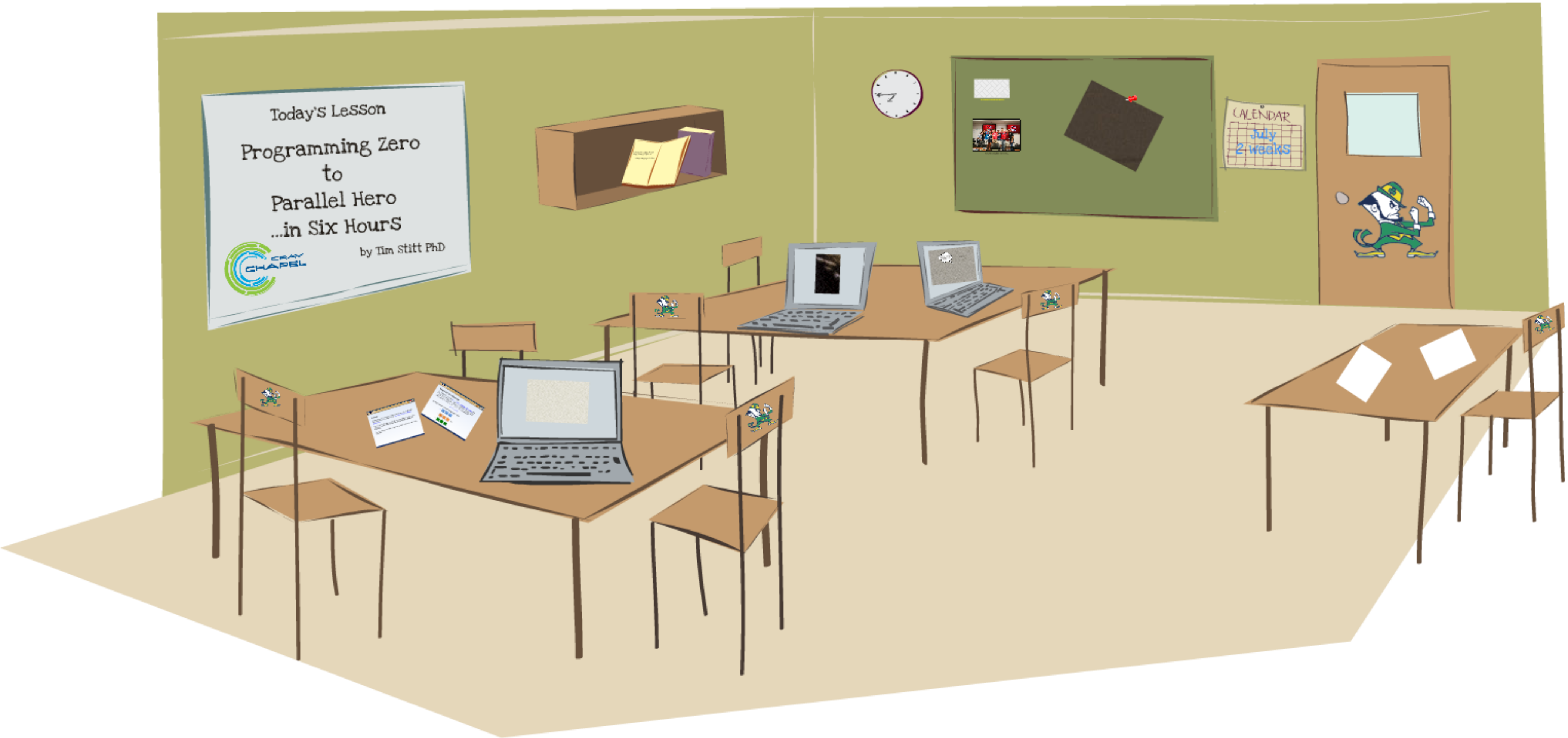
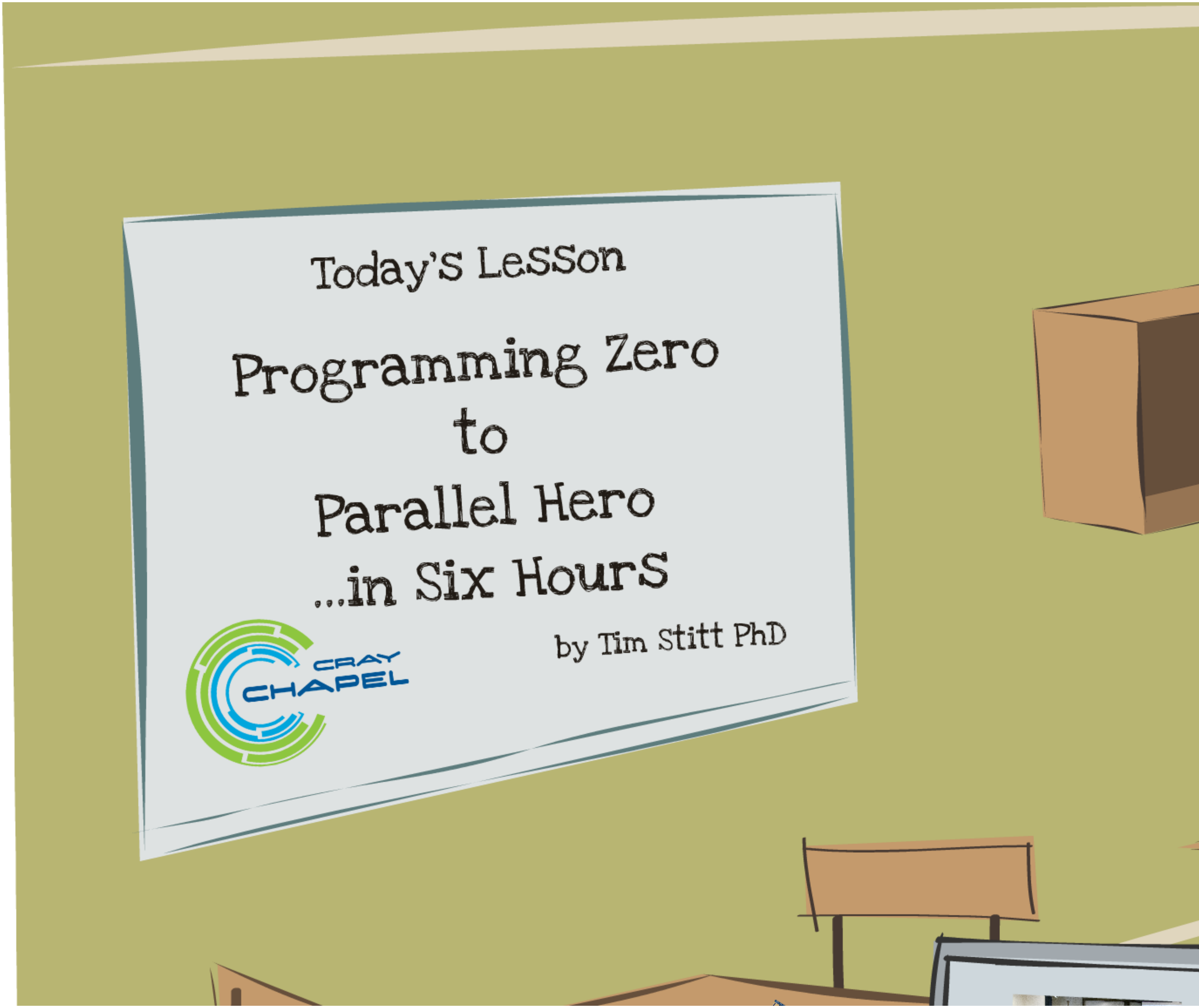
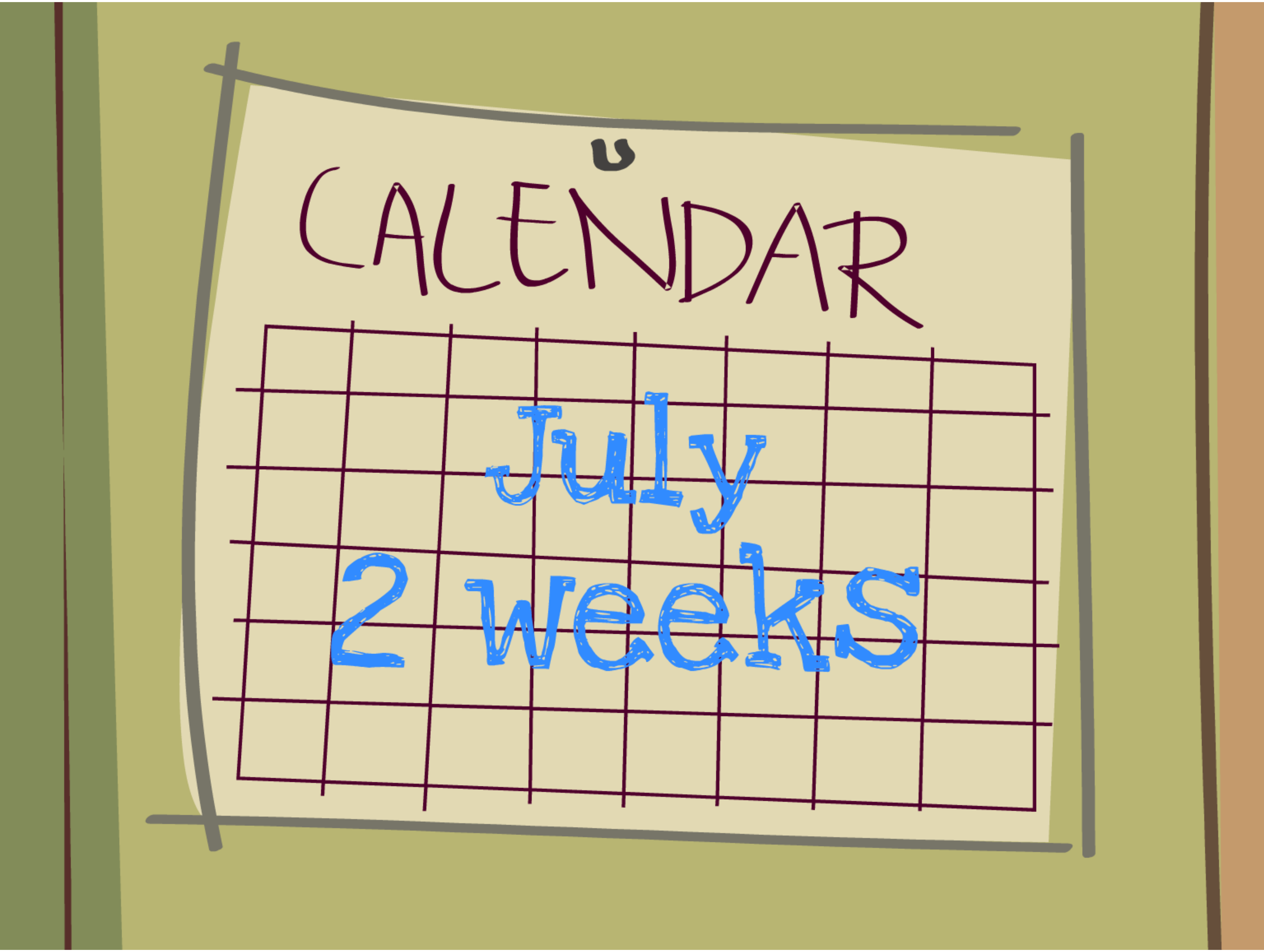


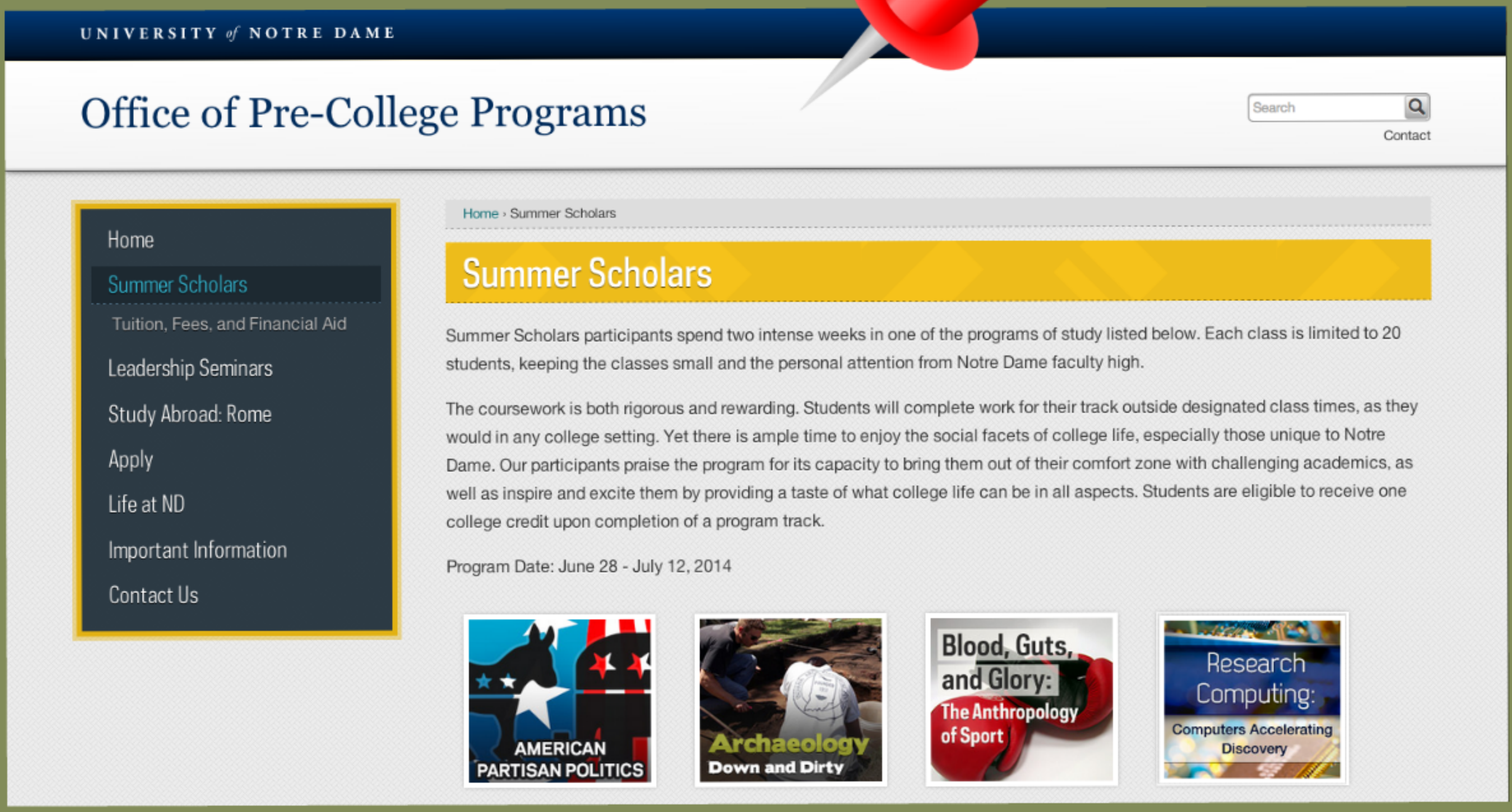
Today's Lesson

Programming Zero to Parallel Hero ...in Six Hours

by Tim Stitt PhD


Today's Lesson
Programming Zero
to
Parallel Hero
...in Six Hours
by Tim Stitt PhD
CRAY
CHAPEL


CALENDAR
July
2 weeks


UNIVERSITY of NOTRE DAME
Search
Contact
Home
Summer Scholars
Tuition, Fees, and Financial Aid
Leadership Seminars
Study Abroad: Rome
Apply
Life at ND
Important Information
Contact Us
Home › Summer Scholars
AMERICAN PARTISAN POLITICS
Archaeology Down and Dirty
Blood, Guts, and Glory: The Anthropology of Sport
Research Computing: Computers Accelerating Discovery

Office of Pre-College Programs

Summer Scholars

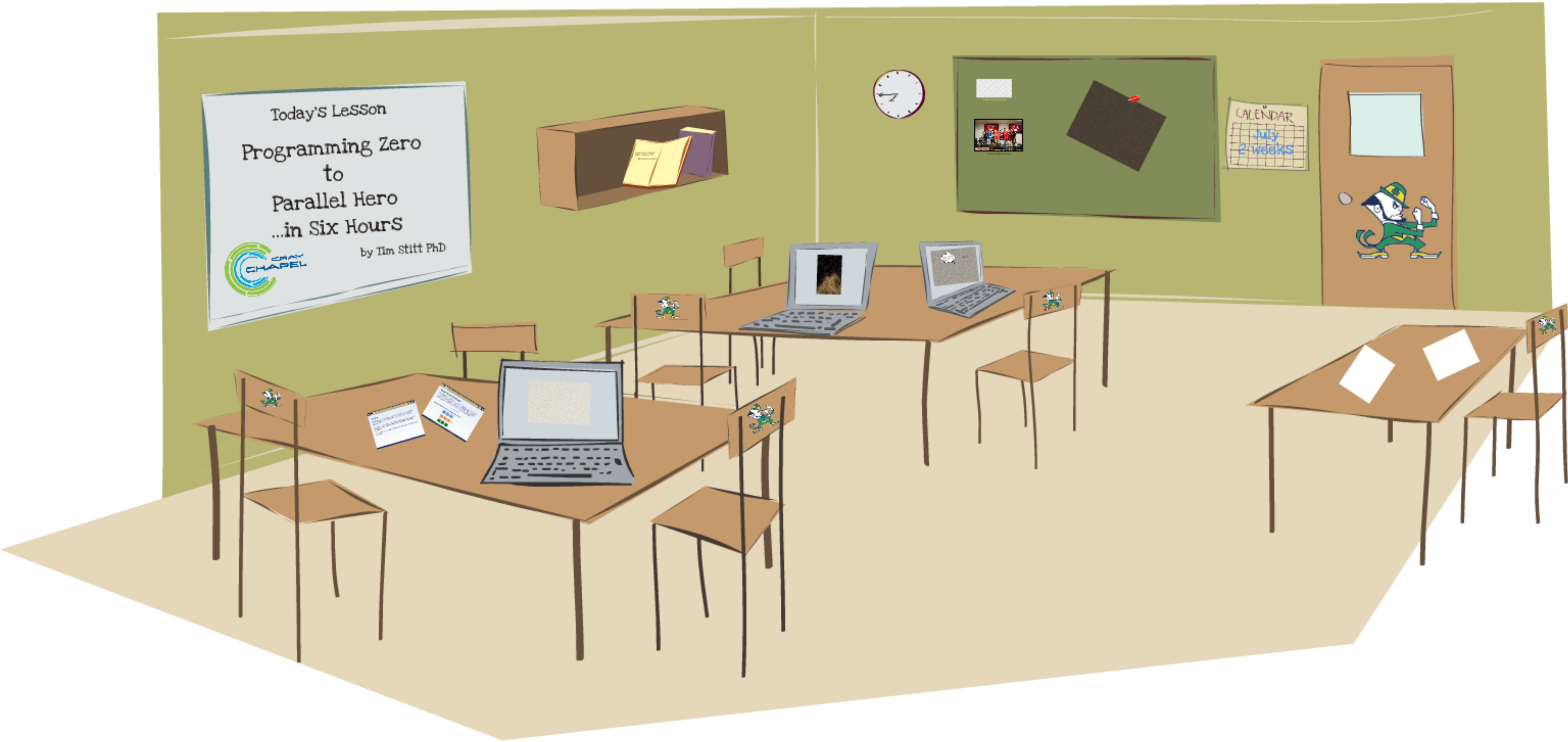
Summer Scholars participants spend two intense weeks in one of the programs of study listed below. Each class is limited to 20 students, keeping the classes small and the personal attention from Notre Dame faculty high.

The coursework is both rigorous and rewarding. Students will complete work for their track outside designated class times, as they would in any college setting. Yet there is ample time to enjoy the social facets of college life, especially those unique to Notre Dame. Our participants praise the program for its capacity to bring them out of their comfort zone with challenging academics, as well as inspire and excite them by providing a taste of what college life can be in all aspects. Students are eligible to receive one college credit upon completion of a program track.

Program Date: June 28 - July 12, 2014

http://precollege.nd.edu/summer-scholars/

Summer Scholars Group 2011


Today's Lesson
Programming Zero
to
Parallel Hero
...in Six Hours
by Tim Stitt PhD
CRAY
CHAPEL
CALENDAR
July
2 weeks

ALL PRESENTATIONS WILL BE HELD IN ROOM 302 (JORDAN HALL OF SCIENCE) UNLESS OTHERWISE NOTIFIED

Monday	Tuesday	Wednesday	Thursday	Friday	Saturd
July 1-12, 2013 – Research Computing Expeditions					
1st	**2nd**	**3rd**	**4th**	**5th**	**6th**
9:00 – 9:10 *Welcome* (**Dr. Jarek Nabrzyski**) **9:10 – 10:15** *Research Computing Intro* Research Computing: The Third Pillar of Science (**Dr. Tim Stitt**) **10:45 – 12:00** *Research Computing Hardware* Essential HPC Hardware and Communications (**Dr. Paul Brenner**)	**9:00 – 10:00** *Research Computing Software* Operating Systems for High-Performance Computing Environments **(Rich Sudlow, Steve Bogol)** **10:15 – 12:00** *Research Computing Application* *(LAB)* Student Projects (**Dr. Paul Brenner**)	**9:00 – 12:00** Field Trip to Union Station Data Center	***NO CLASSES***	**9:00 – 12:00** *Research Computing Software* *(LAB)* Programming Supercomputers Part I (**Dr. Tim Stitt**)	
LUNCH	**LUNCH**	**LUNCH**		**LUNCH**	
1:15 – 2:30 *Research Computing Application* Animation, Rendering and Visual Effects on Clusters (**Ramzi Bualuan**) **2:45 – 4:00** *Research Computing Application* “Finding genes associated with complex diseases and traits with HPC” (**Patrick Miller, Dan McArtor**)	**1:00 – 1:45** *Research Computing Hardware* HPC File-systems and Storage (**Serguei Federov**) **2:00 – 4:00** *Research Computing Application* Numerical Weather Prediction and Hurricane Modeling. (**Dr. Ed. Bensman**)	**1:00 - 4:00** *Cluster Building* *(LAB)* How To Build Your Own HPC Cluster **(Rich Sudlow, Steve Bogol)**		**1:00 – 4:00** *Research Computing Software* *(LAB)* Programming Supercomputers Part II (**Dr. Tim Stitt**)	

ALL PRESENTATIONS WILL BE HELD IN ROOM 302 (JORDAN HALL OF SCIENCE) UNLESS OTHERWISE NOTIFIED

Monday	Tuesday	Wednesday	Thursday	Friday	Saturd
8th	**9th**	**10th**	**11th**	**12th**	**13th**
09:00 – 12:00 *Research Computing Application (LAB)* A Introduction to Mathematical Modeling in MATLAB (**Aaron Donahue**) **LUNCH** **1:00 – 2:30** *Research Computing Software* GPGPU Programming/Visualization (**Chris/James Sweet**) **2:45 – 4:00** *Research Computing Application* Particle Physics on the Grid: The CERN CMS Experiment at Notre Dame (**Prof. Mike Hildreth**)	**9:00 - 12:00** *Research Computing Software (LAB)* A Crash Course in Python Programming (**David Pettifor**) **LUNCH** **1:00 - 4:00** *Research Computing Application (LAB)* Agent-Based Modeling with Repast (**Dr. Paul Brenner**)	**9:00 - 18:30** Argonne National Lab Tour and Navy Pier Visit	**9:00 – 10:30** *Research Computing Application* High Performance Computing in Finance and Economics (**Dr. Jeff Thurk**) **10:45 – 12:00** *Research Computing Application* Cloud Computing (**Dr. Jarek Nabrzyski**) **LUNCH** **1:00 – 2:15** *Research Computing Application* High Performance Computing in Fluid Mechanics (**Dr. Aleks Jemcov**) **2:30 – 4:00** *Research Computing Application* Tour of Education Technology Lab (**Paul Turner**)	**9:00 – 10:00** Tour of Nano-Fabrication Facility. (**Yaakov Sloman**) **10:00 – 11:00 (AGR-CRC)** Project Reports **11:00 – 12:00 (Debartolo)** Measuring Concussions on the iPAD (**Dr. Chris Poellabauer / Shane Quinlan, Nikhil**) **12:00** Wrap-Up ***NO AFTERNOON CLASSES***	STUDENTS DEPART

Procedures, by example

- Example to compute the area of a circle

```
def area(radius: real): real {
  return 3.14 * radius**2;
}

writeln(area(2.0));    // 12.56
```

```
def area(radius = 0.0) {
  return 3.14 * radius**2;
}
```

Argument and return types can be omitted

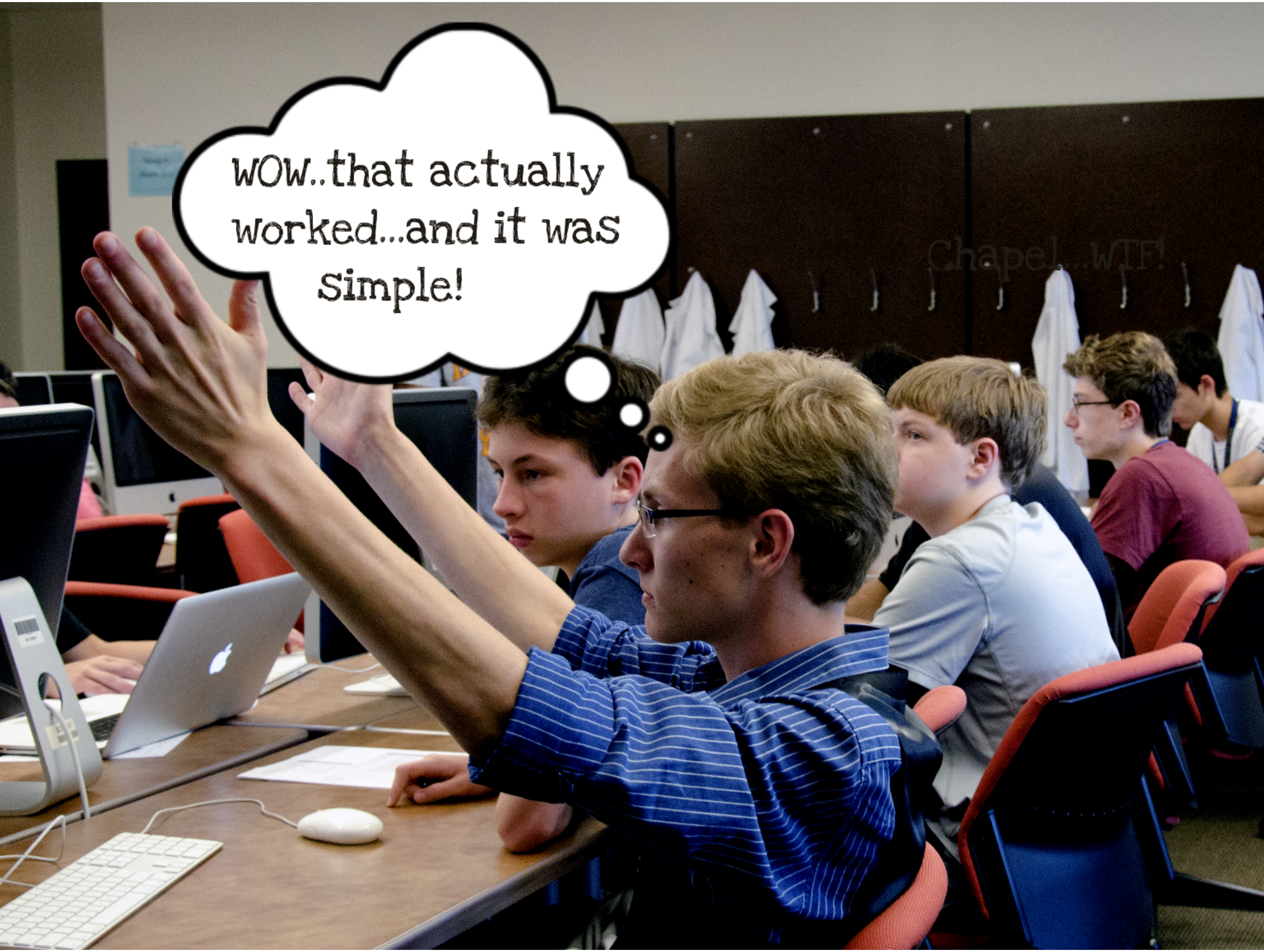
- Example of argument default values, naming

```
def writeCoord(x: real = 0.0, y: real = 0.0) {
  writeln((x,y));
}

writeCoord(2.0);            // (2.0, 0.0)
writeCoord(y=2.0);          // (0.0, 2.0)
writeCorrd(y=2.0, 3.0); // (3.0, 2.0)
```

Chapel Base Language

Group B
Room 302
ssh -X
NOTRE DAME

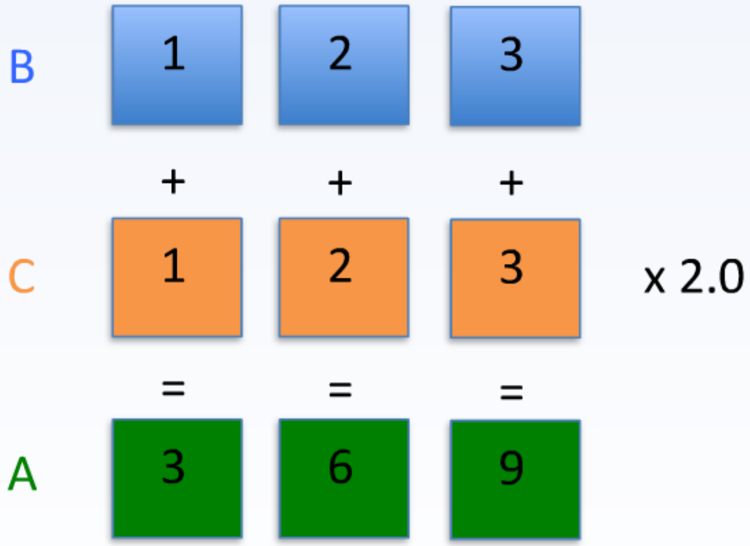

WOW..that actually worked...and it was simple!

Chapel Cluster Challenge

The STREAM benchmark (http://www.cs.virginia.edu/stream/) is a well-known measurement technique for evaluating high-performance computers. The STREAM benchmark performs the following simple mathematical operation on three (3) one-dimensional arrays:

$$A = B + \alpha \bullet C$$

Example (with 3 element arrays and α=2.0)



B	1	2	3	
	+	+	+	
C	1	2	3	x 2.0
	=	=	=	
A	3	6	9	

Challenge:

Parallelize the (non-parallel) Chapel implementation of the STREAM benchmark stored in the source file /usr/local/chapel/challenge/stream.chpl.

The winner will be based on the best timing achieved for the problem using your cluster. (*Note: the elegance of your solution will also be important*).

Your entry should consist of wallclock timings obtained across 1,2 and 4 Locales.

Compete at XSEDE12 Student Programming Competition

- Chapel language of choice