COSC 407 Introduction to Parallel Computing

Instructor: Yves Lucet, PhD, Unit 5 (Computer Science, Mathematics, Physics, Statistics), UBCO, ASC 350, ext 79505

Duration: 1 semester, 3 credits **Lectures:** 3 hours/week. **Laboratory:** 2 hours/week
Office Hours: Tuesdays, Thursdays 2-3pm.
Calendar Description: Design and implementation of parallel programs including theoretical computer models, parallel architectures (distributed, multicore, GPU), and standard parallel libraries.

Learning Objectives: The course will provide 3rd and 4th year students with an introduction to parallel computing. Upon completion of the course students will be able to understand parallel computing architectures and their limitations, create and implement parallel programs using various standard libraries, explain the limitation of the IEEE 754 floating point model, determine whether an undesirable output is due to floating point errors, and write parallel code.

Pre-requisite: 3rd year standing. Familiarity with a programming language (Java, C, or C++) is strongly recommended.

Textbook (Mandatory):

- Pacheco, P. S. An introduction to Parallel Programming, Morgan Kaufmann, 2011
- Kirk, D. B. & Hwu, W.-m. W. Programming Massively Parallel Processors: A Hands-on Approach, Morgan Kaufmann Publishers Inc., 2010

Additional references:

- Rauber, T. & Rünger, G. Parallel Programming: for Multicore and Cluster Systems, Springer Publishing Company Inc., 2010
- Eijkhout, V. Introduction to High-Performance Scientific Computing, available online at lulu.com, 2010.
- Sanders, J. & Kandrot, E. CUDA by Example: An Introduction to General-Purpose GPU Programming, Addison-Wesley Professional, 2010
- Burden, R. L. & Faires, J. D. Numerical Analysis, Brooks/Cole, 9th ed, 2010

Course content:

- **Parallel Programming:**
  - Theoretical computer models: sequential (von Neuman architecture, RAM) and parallel (PRAM, CRCW PRAM, BSP, LogP)
  - Efficiency metrics: MIPS, FLOPS, scalability
  - Theoretical limitations of parallelism: Hamdal's law, Guftasson's law
  - Parallel architectures: distributed, shared memory, GPU
  - Current standards: OpenMP, MPI, CUDA
  - Processes and Threads (PThreads, Win 32 API, Java theads)
  - Debugging parallel code: deadlock, race conditions
- **Scientific computing/numerical analysis:**
  - IEEE 754 floating point standard
  - Numerical stability: nonassociativity of floating point addition
  - Parallel linear algebra: matrix-matrix multiplication, parallel sort
  - Parallel scientific computing: solving linear systems in parallel

- **Tentative course outline:**

Week	Lecture	Date	Topic
1	1	Jan 07	Syllabus, intro: why parallel
	2	Jan 09	Intro to C
2	3	Jan 14	TBL, Grading, create teams
	4	Jan 16	OpenMP
3	5	Jan 21	OpenMP
	6	Jan 23	OpenMP
4	7	Jan 28	OpenMP
	8	Jan 30	CUDA
5	9	Feb 04	CUDA
	10	Feb 06	CUDA
6	11	Feb 11	CUDA
	12	Feb 13	Theory
		Feb 18	reading break
		Feb 20	reading break
7	13	Feb 25	Theory
	14	Feb 27	CUDA Floating point
8	15	Mar 04	Floating point, IEEE 754
	16	Mar 06	Theory
9	17	Mar 11	MPI
	18	Mar 13	MPI
10	19	Mar 18	MPI
	20	Mar 20	Threads
11	21	Mar 25	Threads
	22	Mar 27	Threads
12	23	Apr 01	Threads
	24	Apr 03	Review

Marking: There will be four individual + team quizzes, lab assignments, four 80 minute team midterms, and one 3 hour final exam. The grade for the course will be the weighted sum of the individual performance, team performance, and team maintenance with weights as determined by the class.

Grade Weight		Range
Individual performance		40-60
	individual readiness assessment tests	40
	Labs	40
	Final exam	20
Team performance		30-50
	team readiness assessment tests	20-50
	team midterm: in-class problem solving	50-80
Team Maintenance		5-15
	Peer evaluation	100

In addition, *you need to pass the final to pass the course*. Failing to pass the final will result in a maximum mark of 45%.

Sickness: In case you cannot be present at a midterm because of sickness, notify the instructor immediately and bring a note from your doctor at the next class. The weight of that midterm will be distributed among your other midterm and the final. If you cannot attend two midterms for sickness, both weights will be assigned to the final. If you cannot attend the final, see the associate dean.
Calendar Dates: See http://www.calendar.ubc.ca/okanagan/
Plagiarism and collaboration: The "default" assumption is that students will work on assignments independently . Students who complete assignments with the aid of collaborators or other sources (e.g. other text-books) must
(i) acknowledge this fact (including the name(s) of other sources) at the start of their homework submission (see above),
(ii) produce an independent writeup (copied submissions are not permitted),
(iii) be prepared to explain their solutions in further detail, if asked, and
(iv) be prepared to have the assignment grade adjusted accordingly.
Collaborating in groups of size greater than three is not permitted.
Plagiarism (the submission of work of another person as your own) and other anti-intellectual behaviour will not be tolerated. Your attention is directed to the "Student Discipline" section of the University Calendar as well as the UBC-V computer science Department Policy on "Plagiarism and Collaboration", available through the Undergraduate Web Page at http://www.cs.ubc.ca/our-department/administration/policies/collaboration. In particular, note that **it is not acceptable to make a solution available as an aid to others**.

Cooperation vs. Cheating: Working with others on assignments is a good way to learn the material and we encourage it. However, there are limits to the degree of cooperation that we will permit. Any level of cooperation beyond what is permitted is considered cheating.
When working on programming assignments, you must work only with others whose understanding of the material is approximately equal to yours. In this situation, working together to find a good approach for solving a programming problem is cooperation; listening while someone dictates a solution is cheating. You must limit collaboration to a high-level discussion of solution strategies, and stop short of actually writing down a group answer. Anything that you hand in, whether it is a written problem or a computer program, must be written by you, from scratch, in your own words. If you base your solution on any other written solution, you are cheating.
There will be random audit of assignment solutions through internet-based source code search engine: Any assignment found to be significantly similar to a publicaly available source code without the proper acknowlegdment will trigger an investigation for academic dihonesty in addition to any copyright violation.

If you have any doubt that an action you are considering might be construed, by anyone, as cheating, DON'T DO IT. Ask permission first.

Academic Integrity: The academic enterprise is founded on honesty, civility, and integrity. As members of this enterprise, all students are expected to know, understand, and follow the codes of conduct regarding academic integrity. At the most basic level, this means submitting only original work done by you and acknowledging all sources of information or ideas and attributing them to others as required. This also means you should not cheat, copy, or mislead others about what is your work. Violations of academic integrity (i.e., misconduct) lead to the break down of the academic enterprise, and therefore serious consequences arise and harsh sanctions are imposed. For example, incidences of plagiarism or cheating usually result in a failing grade or mark of zero on the assignment or in the course. Careful records are kept in order to monitor and prevent recidivism.
A more detailed description of academic integrity, including the policies and procedures, may be found in the calendar at http://okanagan.students.ubc.ca/calendar/index.cfm?tree=3,54,111,0.

If you have any questions about how academic integrity applies to this course, please consult with your professor.

Equity, Human Rights, Discrimination and Harassment
UBC Okanagan is a place where every student, staff and faculty member should be able to study and work in an environment that is free from human rights based discrimination and harassment. If you require assistance related to an issue of equity, discrimination or harassment, please contact the Equity Office, your administrative head of unit, and/or your unit's equity representative.
UBC Okanagan Equity Advisor: ph. 250-807-9291; email equity.ubco@ubc.ca
Web: www.ubc.ca/okanagan/equity
Unit Equity Representatives:
http://www.ubc.ca/okanagan/equity/programs/equityreps/unitcontacts.html

The present syllabus is subject to change according to the needs and interests of the class.