CSci 5451 Spring 2021
Introduction to parallel computing

General Information

This course is an introduction to parallel computing. It covers parallel architectures, parallel algorithms and their analysis. It will also introduce you to programming on parallel platforms. The main programming medium used for the labs will in C /C++. OpenMP, MPI, and CUDA for the NVIDIA Graphics Processing Units (GPUs), will also be covered, along with a quick overview of PGAS, Pthreads, and openACC. The course blends theory [complexity of parallel algorithms and their efficiency] with practical issues such as parallel architectures and parallel programming. A small part of the course will be devoted to the emerging paradigm of quantum computing.

- **Class Schedule:** MW 08:15 AM - 09:30 AM — Online
- **Instructor:** Yousef Saad ≪ saad@cs.umn.edu ≫ `http://www.cs.umn.edu/~saad`
  Office: Keller H. 5 -225B – Office Phone: (612) 624 – 7804.
- **Teaching Assistant:** James Mooney ≪ moone174@umn.edu ≫
- **Office hours:** Posted on the class web-site
- **Class Web-site:** Lecture notes (only) will be posted here:
  `www-users.cselabs.umn.edu/classes/Spring-2021/csci5451/`

  Detailed schedule, Homeworks, labs, grades, will be posted on canvas. Submissions of labs and homeworks on canvas.

  It is your responsability to check both Canvas (especially for homeworks) and the cselabs website (for lecture notes) on a regular basis.

Texts

With so much available on the internet, there is no need for a formal textbook. The closest references for some (not all) the material covered in the course are the following:

- *"Introduction to Parallel Computing, 2nd edition"*, by V. Kumar, A. Grama, A. Gupta, and G. Karypis (2003). ISBN-13: 978-0201648652; ISBN-10: 0201648652
- *"Introduction to Parallel Programming"*, by Peter S. Patcheco, Elsevier (2011). ISBN 9780080921440, 0080921442
- *"Programming Massively Parallel Processors, Third Edition: A Hands-on Approach"* by David B. Kirk and Wen-mei W. Hwu. (2017) – ISBN-13: 978-0128119860

The first reference is on theoretical aspects and the other 2 on parallel programming (2nd one mostly on CUDA). Links to additional material will be also provided as the course progresses.

Lecture Notes

Lecture notes will be posted regularly on the CSE class web-site (see above – not on canvas). Click on the "Lect. Notes" tab in the menu. These notes will be posted by topic rather than lecture by lecture, and they are usually posted prior to the lectures.
Also note that the videos of the lectures will be recorded and posted on canvas – usually on the same day as the lecture.

Evaluation

Your evaluation for this class will be based on 2 homeworks, 3 Labs, 4 quizzes (4 best out of 5 given), and a participation score.

- 2 times 10% = 20% for 2 homeworks
- 3 times 12% = 36% for 3 labs
- 4 times 10 % = 40% for 4 quizzes.
- 4% for participation.

There are actually 5 quizzes scheduled but only the 4 best will be taken into account to calculate your score for the quizzes portion. There will be no make-up quizzes. The quizzes will be 30mn each and will be online. There is no final exam for this class.
The participation score will be mostly based on class attendance and participation. If for any reason you cannot attend a class, you can send me an e-mail. *If you are not able to attend class on a regular basis due to other committments (e.g., work) you need to contact me at the start of the semester so you wont be penalized by a poor participation grade.*
Final grades will be decided based on the following scale, where T is the total score (out of 100) you achieved in the class.

A : $100 \geq T \geq 94$	**A-** : $94 > T \geq 88$	**B+** : $88 > T \geq 82$
B : $82 > T \geq 77$	**B-** : $77 > T \geq 72$	**C+** : $72 > T \geq 65$
C : $65 > T \geq 60$	**C-** : $60 > T \geq 55$	**D+** : $55 > T \geq 50$
D : $50 > T \geq 40$	**F** : $40 > T$	

If you are taking the class on an S-N basis your total score must be at least 60% in order to get an S grade.

Grading

Grades will be posted immediatly after each Homework/ Lab or test is graded. This will usually take about one week. It is important that you check your grades regularly. If you see a discrepancy between your grades and the grades posted, you need to alert the TA immediatly. You have one week after the homework/ test is returned for requesting a change. Details on this can be found in the general **policy on homeworks and tests** which is posted in the class web-site.

Cheating

All homeworks labs, and tests, must represent your own individual effort.

> *It is very important to understand what constitutes cheating. If you have questions please ask. I hate doing this but I have not hesitated in giving fails for cases of cheating in the past. It is *not worth it*. In addition note that it is not difficult to spot a cheating situation. Absolute no-no: exchanging codes or text for hw. Discussions of HW and Labs should be limited to the general understanding of the ideas.*

Cheating cases will be dealt with in a very strict manner. Violators of this policy will fail the course and will have their names recorded. For additional information please consult the student code of conduct which can be found here: `https://regents.umn.edu/policies/index`

Overview of topics to be covered

[Tentative! – this will be revised prior to 1st class –]

- Introduction; Historical Perspective; Types of parallelism; parallel algorithms and parallel computing.
- Parallel computing platforms, Taxonomy, Pipelined-, Vector-, superscalar. Examples of parallel platforms.
- Memory and cache performance issues, Hierarchical memories, Latency, bandwidth, Caches, How do caches work, Examples.
- Parallel algorithms, design. Parallel performance metrics (Efficiency, load balancing, scalability, ..)
- Programming shared memory machines. openMP, Posix threads, PGAS.
- Programming GPUs, CUDA, openACC.
- Static networks; Linear arrays rings and meshes; Hypercubes; Fat Trees; Graph embeddings; Routing.
- Basic communication operations. Programming with MPI.
- Programming distributed systrems. MPI
- Numerical parallel algorithms: (Dense matrix computations, Sparse matrix algorithms)

- Non-numerical parallel algorithms; searching, sorting, graph algorithms.
- Introduction to Quantum Computing. Quantum computers, gates, Cirq and Qiskit development kits, Examples (Deutsch-Josza, QFT, Shor).