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 be 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 — ME 212
- **Instructor:** Yousef Saad ≪ saad@cs.umn.edu ≫ [http://www.cs.umn.edu/~saad](http://www.cs.umn.edu/~saad)
  Office: Keller H. 5 -225B – Office Phone: (612) 624 – 7804.
- **Teaching Assistant:** Tianshi Xu ≪ xuxxx1180@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-2019/csci5451/](http://www-users.cselabs.umn.edu/classes/Spring-2019/csci5451/)
  Detailed schedule, Homeworks, labs, grades, will be posted on canvas. Submissions of labs and homeworks on canvas.
  It is your responsibility 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:

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 class web-site (see above – not on canvas). Click on the "Lect. Notes" icon 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 this class is on Unite so you will have access to streaming video archives of class meetings on a TEN-DAY delay (7-DAY prior to exams) for the length of the semester.

**Evaluation**

Your evaluation for this class will be based on 2 homeworks, 3 Labs, 3 mid-term exams and a final.

- Homework and Labs: 46 %, consisting of 2 homeworks \((2 \times 8 \%)\) and 3 Labs \((3 \times 10 \%)\).
- Exams: 54\%, consisting of 3 mid-term exams \((3 \times 12 \%)\) and a final \((18\%)\).

Exams are all mandatory and *no make-up exams will be given*. If you miss any of the given exams and you have a *valid justification* you will need to bring a proof of the exceptional and/or emergency nature of the reason you missed the exam. If the circumstance is deemed valid then the mean of the other exams will be used to replace the missing grade so as to provide a "neutral" substitute. The mid-term tests may be 45mn or 60mn long. The final will be either 90mn or 120 mn long.

I may include a few bonus points (max of 3 points to be added to the final score) for a few students who show very active participation in class. There may be a few quizzes with the goal of improving class participation and discussions. These will not be graded. Final grades will be decided based on the following scale, where \(T\) is the total score (out of 100) you achieved in the class.

<table>
<thead>
<tr>
<th>Grade</th>
<th>Score Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>100 ≥ (T) ≥ 94</td>
</tr>
<tr>
<td>A-</td>
<td>94 &gt; (T) ≥ 88</td>
</tr>
<tr>
<td>B</td>
<td>82 &gt; (T) ≥ 77</td>
</tr>
<tr>
<td>B-</td>
<td>77 &gt; (T) ≥ 72</td>
</tr>
<tr>
<td>C</td>
<td>65 &gt; (T) ≥ 60</td>
</tr>
<tr>
<td>C-</td>
<td>60 &gt; (T) ≥ 55</td>
</tr>
<tr>
<td>D</td>
<td>50 &gt; (T) ≥ 40</td>
</tr>
<tr>
<td>F</td>
<td>40 &gt; (T)</td>
</tr>
<tr>
<td>B+</td>
<td>88 &gt; (T) ≥ 82</td>
</tr>
<tr>
<td>C+</td>
<td>72 &gt; (T) ≥ 65</td>
</tr>
<tr>
<td>D+</td>
<td>55 &gt; (T) ≥ 50</td>
</tr>
</tbody>
</table>

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 immediately after each Homework/ Lab or exam 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 immediately. You have one week after the homework/ exam is returned for requesting a change. Details on this can be found in the general policy on homeworks and exams which is posted in the class web-site.

Cheating

All homeworks labs, and exams, must represent your own individual effort. Cheating cases will be dealt with in a very strict manner. At a minimum, 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 systems. 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).