TIMETABLE: 39222 MWF noon–12:50pm in Lincoln Hall 107.

CREDIT HOURS: 4 hours.

PREREQUISITES: The catalog lists MCS 471 (numerical analysis) or MCS 571 (numerical analysis of partial differential equations) or consent of the instructor.
MCS 507 (scientific software) prepares also very well for this course.

COMPUTATIONAL SCIENCE: MCS 572 is one of the courses on the computational science prelim. In a broader context, MCS 572 fits in an interdisciplinary computational science and engineering (CSE) curriculum.

COURSE GOALS AND LEARNING OBJECTIVES: The main goal of MCS 572 is to study parallel algorithms and their realization on real machines. The learning objectives include the design and analysis of parallel programs, using distributed memory, shared memory parallel computers, and acceleration with graphics processing units. Another learning objective is the application of parallel software to scientific problems.

INSTRUCTOR: Jan Verschelde, Office: 1210 SEO.
E-mail: janv@uic.edu. URL: `http://www.math.uic.edu/~jan`.

OFFICE HOURS: At 1pm on Monday, Wednesday, and Friday, in office; or by appointment; via zoom `https://uic.zoom.us/my/profjanofficehour` or in person.

TEXTBOOK: The recommended textbooks are (1) *Parallel Programming. Techniques and Applications Using Networked Workstations and Parallel Computers* by Barry Wilkinson and Michael Allen, Pearson Prentice Hall, second edition, 2005; and (2) *Programming Massively Parallel Processors. A Hands-on Approach* by David B. Kirk and Wen-mei W. Hwu, Elsevier/Morgan Kaufmann Publishers, 2010; 4th edition, 2023, with Izzat El Hajj as 3rd author.

MCS 572 SITE: See `http://www.math.uic.edu/~jan/mcs572/index.html` for a copy of the syllabus, posting of slides, reference materials, and changes in the scheduling.
Backup site: `https://janv.people.uic.edu/mcs572`.

HOMEWORK: At every lecture, several exercises are listed. Some exercises provide inspiration for an interesting project. The collection of homework will be announced at least one week before the deadline.

PROJECTS: Three projects will be assigned during the semester. In the first project we will use MPI to experiment with the concepts and algorithms in the book. The goal of the second project is a computational detailed study of a parallel algorithm. In the third project we consider application fields, still to be determined, and depending on personal interests and preferences. The third project could be developed into a final project, instead of a final exam.

EXAMS: During the semester, there is one midterm on the first half of the course. As a take-home exam, the midterm functions as an important homework collection. Instead of a classical review week and a final exam, the last week of classes could be spent on project presentations, so the final grade is determined mainly by computer projects.

COMPUTERS: Access to a supercomputer is to be confirmed.
Access will be given to an multicore work station with an NVIDIA GPU.

POLICY FOR MISSED OR LATE WORK: Deadlines may be postponed. If you know you will be late, then it is better to apply for an extension of the deadline, instead of not submitting anything. For missed assignments, greater weight may be placed on the final project and/or final exam.

STUDENTS WITH DISABILITIES: UIC is committed to full inclusion and participation of people with disabilities in all aspects of university life. Students who face or anticipate disability-related barriers while at UIC should connect with the Disability Resource Center (DRC) at drc.uic.edu, drc@uic.edu, or at (312) 413-2183 to create a plan for reasonable accommodations. In order to receive accommodations, students must disclose disability to the DRC, complete an interactive registration process with the DRC, and provide their course instructor with a Letter of Accommodation (LOA). Course instructors in receipt of an LOA will work with the student and the DRC to implement approved accommodations.

CLASS ATTENDANCE: Students are expected to attend all class meetings. Any changes in this syllabus or in the scheduling of exams and other assignments will be announced during class meetings. We will also address the topics you need to implement the projects.

CLASSROOM CONDUCT POLICY: Laptops are permitted and even encouraged to run the posted programs and/or view the slides. Respect others in the class. No cellphone usage is permitted. No food and no drinks are allowed.

SOME IMPORTANT DATES:
Friday 30 August : last day to register, last day to withdraw without W grade.
Monday 2 September : Labor Day Holiday. No classes.
Wednesday 27 November : Student Wellness Day. No classes.
Thursday 28 – Friday 29 November : Thanksgiving holiday. No classes.

LAST REVISED: Tuesday 7 October 2024.

COURSE OUTLINE — subject to changes :

0. introduction

L-1	Mon 26 Aug	welcome to mcs 572 – supercomputing – measuring performance
L-2	Wed 28 Aug	scalability – types of parallel computing
L-3	Fri 30 Aug	high level parallel processing
	Mon 2 Sep	**Labor Day holiday. No classes.**

1. distributed memory parallel computing

L-4	Wed 4 Sep	basics of Message Passing (MPI) – broadcasting data
L-5	Fri 6 Sep	using MPI to write parallel programs
L-6	Mon 9 Sep	pleasingly parallel programs – Monte Carlo simulations
L-7	Wed 11 Sep	static and dynamic task assignments – load balancing
L-8	Fri 13 Sep	hands on supercomputing
L-9	Mon 16 Sep	partitioning and divide-and-conquer strategies

2. shared memory parallel computing

L-10	Wed 18 Sep	shared memory parallelism – an introduction to OpenMP
L-11	Fri 20 Sep	the work crew model with Julia, OpenMP, and pthreads
L-12	Mon 23 Sep	tasking with OpenMP – Bernstein's conditions – task dependence
L-13	Wed 25 Sep	tasking with Julia – parallel recursive functions
L-14	Fri 27 Sep	evaluating performance – metrics, task graph, isoefficiency, roofline
L-15	Mon 30 Sep	work stealing – threading building blocks

3. acceleration with Graphics Processing Units

L-16	Wed 2 Oct	a massively parallel processor: the GPU
L-17	Fri 4 Oct	programming GPUs with PyCUDA and with Julia
L-18	Mon 7 Oct	introduction to CUDA
L-19	Wed 9 Oct	data parallelism and matrix multiplication
L-20	Fri 11 Oct	device memories and matrix-matrix multiplication
L-21	Mon 14 Oct	thread organization and matrix multiplication
L-22	Wed 16 Oct	warps and reduction algorithms
L-23	Fri 18 Oct	review of the first 22 lectures
L-24	Mon 21 Oct	**midterm exam**

4. pipelining and synchronized computations

L-25	Wed 23 Oct	pipelining to create parallel algorithms
L-26	Fri 25 Oct	applying pipelining to sorting
L-27	Mon 28 Oct	solving triangular linear systems with a pipeline
L-28	Wed 30 Oct	synchronization with linear, tree, and butterfly barriers
L-29	Fri 1 Nov	parallel iterative methods to solve linear systems
L-30	Mon 4 Nov	heat distribution – domain decomposition methods
L-31	Wed 6 Nov	memory coalescing techniques
L-32	Fri 8 Nov	tensor cores
L-33	Mon 11 Nov	performance considerations

5. applications

L-34	Wed 13 Nov	parallel FFT and sorting
L-35	Fri 15 Nov	parallel Gaussian elimination
L-36	Mon 18 Nov	GPU accelerated QR
L-37	Wed 20 Nov	Case Study: Advanced MRI Reconstruction
L-38	Fri 22 Nov	multiple double arithmetic on the GPU
L-39	Mon 25 Nov	GPU accelerated Newton's method for Taylor series
	Wed 27 Nov	**Student Wellness Day. No classes.**
	Fri 29 Nov	**Thanksgiving holiday. No classes.**

6. review and/or final project presentations

L-40	Mon 2 Dec	final review and/or project presentations
L-41	Wed 4 Dec	final review and/or project presentations
L-42	Fri 6 Dec	final review and/or project presentations