Syllabus for:
Parallel Computing for Science and Engineering
SDS 374c/394c, unique: 57795/57910

Victor Eijkhout and Lars Koesterke

Spring 2021

Time and place	online	12:30 – 2:00 TTh
Instructor:	Victor Eijkhout	`eijkhout@tacc.utexas.edu`
Instructor:	Lars Koesterke	`lars@tacc.utexas.edu`
Office Hours:		after class or by appointment
TA:	TBD	

1 Checklist

1. Get yourself a TACC account at `https://portal.tacc.utexas.edu/` if you don't already have it (section 7.2).
   Use Canvas to send your

   Real name, EID, TACC username

   to instructors **and** TA(s).
2. Download a copy of the two textbooks; section **??**.
3. Join the Slack workspace: the link will be given through Canvas.
4. If you are not familiar with Linux/Unix, do the tutorial in the IHPSC textbook.
5. End of March: think of a final project, get approval from the instructor(s).

2 Rationale; Course Aims and Objectives

Parallel computing has become the dominant method for achieving high performance in computational science, and engineering research and development. Computer systems containing from 10 to 100,000 CPUS, based on commodity processors from Intel, AMD, and IBM, are now commonplace in academia, government, and industry. The latest generation of microprocessors exposes an even greater opportunity for parallelism through technologies such as Hyper-Threading, multi-core and vector processors. Hence, parallel computing has become an important component of university curricula for scientists and engineers. (Parallel and concurrent computing is also an important topic in branches of computer science as different as web servers and operating system. Those typically have a very different flavor, and we will not go into them.)

This course is intended to teach the fundamentals of parallel programming for scientific and engineering applications. It includes the study of general parallel computing principles, including asymptotics; current computer architectures and how parallelism manifests itself in them. However, the main focus will be on building active command in the two of the main parallel programming systems in science and engineering: MPI (Message Passing Interface) and OpenMP. (The topic conspicuously not discussed is programming of GPUs, through such languages as CUDA.)

3 Format and Procedures

The course is structured around lectures, in-class labs, assignments (both theory and programming), quizzes, and one course project.

3.1 Tentative Course Schedule

The following topics will be taught, though not necessarily strictly in this sequence:

1. Overview of Parallel Computing. Parallel computing concepts, essentials of parallel computer architectures and hardware, and standard programming models for parallel computers. Account setup, hands-on introduction to large-scale system environments for development and execution of parallel applications.
2. Shared Memory Parallel Programming with OpenMP. OpenMP directives, syntax, and operation; parallelize serial codes with OpenMP; control and synchronization; performance issues, scalability, and new features of OpenMP.
3. Distributed Memory Programming with the Message Passing Interface (MPI) library. MPI communication operations and data structures, syntax and features; parallelizing serial codes with MPI, performance issues, future of MPI.
4. Parallel Algorithm Design, Optimization, Hardware, and Scientific Applications Case Studies. The basics elements of parallel algorithm design, optimization and hardware operations for parallel programming will be presented.

3.2 Programming Projects and Topics

There are many practical areas of Parallel Computing to explore, ranging hardware to software. These include topics such as interprocessor communication, data-intensive computing, shared memory, vectorization, and threading in multi-core CPU systems and many-cores accelerator systems (GPU, ARM and MIC). Select, propose, or use your present research project for the basis of a computational project that explores, tests or develops a parallel algorithm, application, or concepts. This is a wide open adventure that may involve hardware, standard programming languages, parallel optimization, parallel I/O, etc. A portfolio of subjects and potential ideas will help jumpstart everybody, so that students can be working on a project by the 4th Section.

Requirements: Must be computational (not just academic theory), use parallel techniques/hardware, relevant to High Performance Computing. Students are expected to turn in code as well as an approximately 5 page report.

4 Assumptions and prerequisites

- Programming experience using at least one of C++/C/Fortran is required. Proficiency equivalent to SDS 322 or COE 322 is expected.
  Not sufficient are: Java, Matlab, Python.
- Students need to be familiar with the basics of the Linux/Unix operating system.
- Familiarity with matrix linear algebra is necessary, for instance 408D or 408M; some familiarity with calculus and multi-variate calculus is desirable.
- Experience with developing scientific codes is helpful but not required. Each student is expected to employ one or more of the parallel concepts and develop a course project. For this it is acceptable to improve or add functionality to on an existing research project. Otherwise, suggestions and guidance will be provided by the instructor.

5 Course Readings/Materials

The lecture content is derived from many different sources, including hardware manuals, journal articles and conference presentations and papers. The suggested texts are a good introduction to the material; however, the following references should be consulted for an in-depth understanding of certain subjects.

Lecture slides and such All lecture slides will be distributed through the repository https://github.com/TACC/sds374394spring2021.

Reference works The MPI standard is online: http://www.mpi-forum.org/docs/; index of all routines: http://www.mcs.anl.gov/research/projects/mpi/www/www3/. The OpenMP standard is likewise electronically available: http://openmp.org/wp/openmp-specifications/. It is a good idea to bookmark these pages.

Required download The following materials should be downloaded from the above Github location.

- *Introduction to High-Performance Scientific Computing*, Victor Eijkhout, contains much background material on computer architectures and computing. The class will regularly refer to this book. It also contains tutorials that will be used in class.
- *Parallel Computing for Science and Engineering*, Victor Eijkhout, is a textbook about MPI and OpenMP.

These books are by the class instructor; he greatly appreciates receiving comments and corrections on this text.

6 Assignments, Assessment, and Evaluation

Since this course aims to instill active command of the programming systems MPI and OpenMP, there is a strong emphasis on programming assignments.

These exercises and assignments will typically only use a small part of the programming systems. Therefore, the semester concludes with a larger programming project that allows the student to show their skill in putting together a complete parallel application. The student is expected to deliver both code and a writeup, in the style of a scientific article, and the quality of this will partly determine the grade for the project.

There will also be a small component of the theory of parallelism.

The final grade is a weighted average of these components: MPI 30%, OpenMP 30%, project 30%, theory 10%.

7 Electronic resources

7.1 Class materials

Class materials, including syllabus, textbooks, lecture slides, will be disseminated through a github repository: `https://github.com/TACC/sds374394spring2021`

Students are strongly recommended to learn minimal git commands, rather than to rely on a browser interface. Signing up for a github account and following this repository is also a good idea.

7.2 Discussion through Slack

Students will have access to a Slack workspace for discussions. The link will be given out through Canvas.

7.3 Computing Resources

Students are free to use personal computers and laptops (or machines belonging to any department they have access to) for developing codes. In that case they need a C/C++ or Fortran compiler with OpenMP and an MPI installation.

However, assignments will ultimately have to be performed on TACC's Stampede or Frontera cluster.

If you don't already have a TACC account, go to `https://portal.tacc.utexas.edu/` and create an account there. Send a note through Canvas to the instructor(s) and TA(s) what login name is you have created. If you already have a TACC account, for instance through working with an advisor, mail us that.

7.4 Canvas

UT Canvas will be used for posting announcements and recording grades.

8 Formal and informal policies

Class attendance and participation policy

We expect students to attend and participate in class in accordance with the UT Honor Code (see below). Students are encouraged to ask questions, especially relating to material used in their projects.

This class will have lecture and lab sessions. During the lectures there are to be no laptops open, as these are distracting to the student, as well as others who can see the screen. Tablets with non-raised screens are allowed.

Academic Integrity

University of Texas Honor Code
The core values of The University of Texas at Austin are learning, discovery, freedom, leadership, individual opportunity, and responsibility. Each member of the university is expected to uphold these values through integrity, honesty, trust, fairness, and respect toward peers and community.

Each student in this course is expected to abide by the University of Texas Honor Code. Any work submitted by a student in this course for academic credit will be the student's own work. Collaborations will be allowed for the course project.

You are encouraged to study together and to discuss information and concepts covered in lecture and the sections with other students. You can give ”consulting” help to or receive ”consulting” help from such students. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an e-mail, an e-mail attachment file, a diskette, or a hard copy.

Should copying occur, both the student who copied work from another student and the student who gave material to be copied will both automatically receive a zero for the assignment. Penalty for violation of this Code can also be extended to include failure of the course and University disciplinary action.

During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result in failure of the exam, and may lead to failure of the course and University disciplinary action.

Religious Holy Days

By UT Austin policy, you must notify the instructors of your pending absence at least fourteen days prior to the date of observance of a religious holy day. If you must miss a class, an examination, a work assignment, or a project in order to observe a religious holy day, they will give you an opportunity to complete the missed work within a reasonable time after the absence.

Use of E-mail for Official Correspondence to Students

All students should become familiar with the University's official e-mail student notification policy. It is the student's responsibility to keep the University informed as to changes in his or her e-mail address. Students are expected to check e-mail on a frequent and regular basis in order to stay current with University-related communications, recognizing that certain communications may be time-critical. It is recommended that e-mail be checked daily, but at a minimum, twice per week. The complete text of this policy and instructions for updating your e-mail address are available at `http://www.utexas.edu/its/help/utmail/1564`.

Documented Disability Statement

Any student with a documented disability who requires academic accommodations should contact Services for Students with Disabilities (SSD) at (512) 471-6259 (voice) or 1-866-329-3986 (video phone). Faculty are not required to provide accommodations without an official accommodation letter from SSD. (Note to Faculty: Details of a student disability are confidential. Faculty should not ask questions related to a student's condition or diagnosis when receiving an official accommodation letter.)

Please notify me as quickly as possible if the material being presented in class is not accessible (e.g., instructional videos need captioning, course packets are not readable for proper alternative text conversion, etc.).

Please notify me as early in the semester as possible if disability-related accommodations for field trips are required. Advanced notice will permit the arrangement of accommodations on the given day (e.g., transportation, site accessibility, etc.).

Contact Services for Students with Disabilities at 471-6259 (voice) or 1-866-329-3986 (video phone) or reference the SSD website for more disability-related information: `http://www.utexas.edu/diversity/ddce/ssd/for_cstudents.php`

Behavior Concerns Advice Line (BCAL)

If you are worried about someone who is acting differently, you may use the Behavior Concerns Advice Line to discuss by phone your concerns about another individual's behavior. This service is provided through a partnership among the Office of the Dean of Students, the Counseling and Mental Health Center (CMHC), the Employee Assistance Program (EAP), and The University of Texas Police Department (UTPD). Call 512-232-5050 or visit `http://www.utexas.edu/safety/bcal`.

Q drop Policy

The State of Texas has enacted a law that limits the number of course drops for academic reasons to six (6). As stated in Senate Bill 1231: Beginning with the fall 2007 academic term, an institution of higher education may not permit an undergraduate student a total of more than six dropped courses, including any course a transfer student has dropped at another institution of higher education, unless the student shows good cause for dropping more than that number.

Emergency Evacuation Policy

Occupants of buildings on the UT Austin campus are required to evacuate and assemble outside when a fire alarm is activated or an announcement is made. Please be aware of the following policies regarding evacuation:

- Familiarize yourself with all exit doors of the classroom and the building. Remember that the nearest exit door may not be the one you used when you entered the building.
- If you require assistance to evacuate, inform me in writing during the first week of class.
- In the event of an evacuation, follow my instructions or those of class instructors.
- Do not re-enter a building unless you are given instructions by the Austin Fire Department, the UT Austin Police Department, or the Fire Prevention Services office.