Astronomy 415 - Fall 2022
"Computational Astrophysics"

Instructor

Prof. Massimo Ricotti
E-mail: ricotti@aumd.edu
Phone: (301) 405 5097
Office: PSC 1156
Office hours: Friday 11am-12pm or by appointment
Class web page: http://www.astro.umd.edu/∼ricotti/NEWWEB/teaching/ASTR415-22.html

TA

Haoying Dai
E-mail: dhy@terpmail.umd.edu
Office: ATL 1351
Office hours: Friday 1-2pm or by appointment

Class Schedule

Lectures on Tuesday and Thursday from 2:00pm to 3:15pm
Room CSS 2428 or Computer Lab (CSS 0224)

SYLLABUS

This course does not require previous programming experience but if you never learned a compiled programming language, by the end of this course, you will.

I discourage the use of any programming language other than Fortran77, Fortran90, C, C++. The use of python can be discussed. Interpreted programming languages like Matlab, Matematica and IDL can be used for plotting data but not to solve the homework. The reason is twofold:

i) The term "computational astrophysics" typically refers to running computationally intensive applications that require high-level efficiency and parallelization. These codes need to run on supercomputers and are typically written in C or Fortran and are parallelized using OpenMP or MPI. Recently, has also become possible to run supercomputer codes on GPU using CUDA or OpenCL languages (we will touch on that).

ii) The course focuses on understanding the mathematical methods involved in writing efficient numerical algorithms. Your goal is to write these algorithms (almost) from scratch as a learning process. However, several interpreted programming languages have these algorithms already built in: they work like "black boxes" with an input and an output, thus are not suited for teaching the numerical methods hidden inside the “black box”.

My “mother tongue” is Fortran, but I can also speak C as a second language, and if necessary I can understand some Java or python. Solutions to the homework will be in C.

The level of the class and the topics covered will partially depend on the student’s previous familiarity with low-level programming languages. Along the way I may complement the lectures with power point presentations on topics that the class is especially interested in. I will post homework on ELMS but I will also keep the webpage updated and link all the course material there. Laptop computers are allowed and even encouraged in class, especially during the first part of the course.

Course Description

This course will provide students with a basic knowledge of numerical methods commonly used in astrophysics and, more generally, in physics. This process will be motivated by concrete examples of modern problems in astrophysics that demand numerical approaches.

As mention above, the material covered will depend on the existing level of computer sophistication among the class participants. However, in broad outline the major course topics will include linear algebra, root finding, least-square fitting, Monte Carlo methods, numerical integration, N-body methods, fluid dynamics, FFTs and time-series analysis.

Learning Outcomes

By the end of the course students should be comfortable working in a Unix environment, compiling and running codes, employing a variety of numerical methods to solve scientific problems and visualization techniques to analyze the results.

Recommended Texts

There is no required text for this course. The following recommendations may be helpful to you. Note that much of the course material will follow, *Numerical Recipes*, which is available online. Most in-class programming examples will be in C, but you are free to chose from any suitable languages for completing the assignments.

Aarseth, S. J. 2003, “Gravitational -body Simulations: Tools and Algorithms,” Cambridge Univ. Press.

Hockney, R. W., and J. W. Eastwood 1988, “Computer Simulation Using Particles,” Hilger. [Out of print?]

Kernigan, B. W., and D. M. Ritchie 1988, “The C Programming Language” (2nd ed.), Prentice-Hall.

Peek, J. D., et al. 1997, “Learning the Unix Operating System (Nutshell Handbook)” (4th ed.), O’Reilly.

Prata, S. 1998, “The Waite Group’s C Primer Plus” (3rd ed.), Howard W. Sams & Co.

Press, W.H. et al. 1992, “Numerical Recipes in Fortran [or C or C++]” (2nd ed.),

Cambridge Univ Press - visit the website at *http://www.nr.com/*.

Yee, H.C. 1989, "A class of High-resolution Explicit and Implicit Shock-Capturing Methods", Tech. Report Lecture Series 1989-04, von Karman Institute of Fluid Dynamics [difficult to find?]

Course Grading

A+	$\geq$ 96%
A	from 93% to 95.9%
A-	from 90% to 92.9%
B+	from 86% to 89.9%
B	from 83% to 85.9%
B-	from 80% to 82.9%
C+	from 76% to 79.9%
C	from 73% to 75.9%
C-	from 70% to 72.9%
D+	from 66% to 69.9%
D	from 63% to 65.9%
D-	from 60% to 62.9%
F	below 60%

There will be no exams in this course. Grades will be determined by homework assignments (6 of them) plus one term paper with the following weights:

Homework	80%
Term paper	20%

There will be no curve on the final grades. There may need to be some adjustment to scores depending on the class average; however, any adjustment will be to lower the percentages given above, never to raise them.

Assignments

All assignments involve programming exercises. To make evaluating your work easier, you must provide me with a *printed solution* of the problem and upload on ELMS a single "stand-alone" file containing all your work by the start of class on the day the assignment is due. The file (e.g., a gzip tar archive or a zip file) when uncompressed must produce a folder (named by your last name), containing a suitable formatted response (PDF is best, as Word has compatibility issues especially with embedded figures) to the questions posed in the assignment, along with a description of the remaining contents of the file, including, as needed, instructions on compiling and running any source code. Ideally a Makefile should be provided. Any static graphical output (plots, etc.) should be embedded in the response document.

The TA will compile and run your code with a set of test parameters to ensure correct functionality and error handling. Toward the end of the course, we will also consider your coding style when evaluating your work. Assignments that are late will automatically incur a 10% penalty unless there are extenuating circumstances. The penalty for late assignment will be commensurate to the number of days the homework is late (e.g., $\sim$ 5% per day). Late assignments must be completed before the solutions are handed in to get any credit.

You may work in groups to discuss programming strategy, but you must submit your own solution to each assignment. Note that, just as for written prose, it is necessary to cite the source of any algorithms you use in completing assignments. This includes Numerical Recipes routines that you use.

Term Project

In addition to the homework I will assign a term project. You can work on the term project in small groups or individually. We will discuss the details and project ideas during the course, but you should expect that you will turn in a writeup of the project and present it in class during one of the last 3 classes of the semester.

Course Policies

It is our shared responsibility to know and abide by the University of Maryland's policies that relate to all courses, which include topics like: academic integrity, student and instructor conduct, accessibility and accommodations, attendance and excused absences, grades and appeals, copyright and intellectual property.
Please visit www.ugst.umd.edu/courserelatedpolicies.html for the Office of Undergraduate Studies' full list of campus-wide policies and follow up with me if you have questions.

The Honor Code

The University of Maryland, College Park has a nationally recognized Code of Academic Integrity, administered by the Student Honor Council. This Code sets standards for academic integrity at Maryland for all undergraduate and graduate students. As a student you are responsible for upholding these standards for this course. It is very important for you to be aware of the consequences of cheating, fabrication, facilitation, and plagiarism. For more information on the Code of Academic Integrity or the Student Honor Council, please visit http://www.shc.umd.edu/. To further exhibit your commitment to academic integrity, remember to sign the Honor Pledge on all examinations and assignments: "I pledge on my honor that I have not given or received any unauthorized assistance on this examination (assignment)." (NOTE: for the assignments in this course, you may include this statement in your e-mail to me, or the enclosed writeup.)

Resources and Accommodations

Accessibility and Disability Services

The University of Maryland is committed to creating and maintaining a welcoming and inclusive educational, working, and living environment for people of all abilities. The University of Maryland is also committed to the principle that no qualified individual with a disability shall, on the basis of disability, be excluded from participation in or be denied the benefits of the services, programs, or activities of the University, or be subjected to discrimination. The Accessibility & Disability Service (ADS) provides reasonable accommodations to qualified individuals to provide equal access to services, programs and activities. ADS cannot assist retroactively, so it is generally best to request accommodations several weeks before the semester begins or as soon as a disability becomes known. Any student who needs accommodations should contact me as soon as possible so that I have sufficient time to make arrangements. For assistance in obtaining an accommodation, contact Accessibility and Disability Service at 301-314-7682, or email them at adsfrontdeskumd.edu. Information about sharing your accommodations with instructors, note taking assistance and more is available from the Counseling Center.

Student Resources and Services

Taking personal responsibility for your own learning means acknowledging when your performance does not match your goals and doing something about it. I hope you will come talk to me so that I can help you find the right approach to success in this course, and I encourage you to visit UMD's Student Academic Support Services website to learn more about the wide range of campus resources available to you. In particular, everyone can use some help sharpening their communication skills (and improving their grade) by visiting UMD's Writing Center and schedule an appointment with the campus Writing Center. You should also know there are a wide range of resources to support you with whatever you might need (UMD's Student Resources and Services website may help). If you feel it would be helpful to have someone to talk to, visit UMD's Counseling Center or one of the many other mental health resources on campus.

Basic Needs Security

If you have difficulty affording groceries or accessing sufficient food to eat every day, or lack a safe and stable place to live, please visit UMD's Division of Student Affairs website for information about resources the campus offers you and let me know if I can help in any way.

Veteran Resources

UMD provides some additional supports to our student veterans. You can access those resources at the office of Veteran Student life and the Counseling Center. Veterans and active duty military personnel with special circumstances (e.g., upcoming deployments, drill requirements, disabilities) are welcome and encouraged to communicate these, in advance if possible, to the instructor.

Participation

Given the interactive style of this class, attendance will be crucial to note-taking and thus your performance in this class. Attendance is particularly important also because class discussion will be a critical component for your learning.

- Each student is expected to make substantive contributions to the learning experience, and attendance is expected for every session.
- Students with a legitimate reason to miss a live session should communicate in advance with the instructor, except in the case of an emergency.
- Students who miss a live session are responsible for learning what they miss from that session.
- Additionally, students must complete all readings and assignments in a timely manner in order to fully participate in class.

Course Evaluation

Your participation in the evaluation of courses through CourseEvalUM is a responsibility you hold as a student member of our academic community. Your feedback is confidential and important to the improvement of teaching and learning at the University as well as to the tenure and promotion process. CourseEvalUM will be open for you to complete your evaluations for 2022 Fall semester from early October through December. Please go directly to the website (http://www.courseevalum.umd.edu/) to complete your evaluations. By completing all of your evaluations each semester, you will have the privilege of accessing online, at Testudo, the evaluation reports for the thousands of courses for which 70% or more students submitted their evaluations.

Tentative Course Outline

Date	**Lecture**	**Reading**	**(NRiC)**	**HW due date**
#1	Aug 30	Introduction to the course and survey	–	
#2	Sep 01	Computer architecture	–	
#3	Sep 06	Introduction to UNIX	tutorial	
#4	Sep 08	Introduction to C	1.1-1.2, tutorial	
#5	Sep 13	Examples in C and debugging	1.1-1.2, tutorial	
#6	Sep 15	Parallel Computing (CPU and GPU)	tutorial	
#7	Sep 20	Data representation	1.3	
#8	Sep 22	Linear algebra, part 1 (Gauss-Jordann)	2.0-2.3	HW1 due
#9	Sep 27	Linear algebra, part 2 (LU & SVD dec.)	2.4-2.6	
#10	Sep 29	Root finding in 1-D	9.0-9.1, 9.4, 9.6	
#11	Oct 04	Root finding in multi-D, num. derivative	5.7	HW2 due
#12	Oct 06	Statistics and the K-S test	14.0-14.3	
#13	Oct 11	Least-squares fitting	15.0-15.2, 15.4-15.5	
#14	Oct 13	Random numbers and cryptography	7.0-7.2	HW3 due
#15	Oct 18	Numerical integration	7.6, 4.0-4.4, 4.6	
#16	Oct 20	Integration of ODEs, part 1 (IVPs)	16.0-16.1	
#17	Oct 25	Integration of ODEs, part 2 (leapfrog)	–	HW4 due
#18	Oct 27	Integration of ODEs, part 3 (stiff ODEs)	16.6, 17.0	
#19	Nov 01	Integration of ODEs, part 4 (2-pt BVPs)	16.6, 17.0	
#20	Nov 03	N-body techniques, part 1	–	HW5 due
#21	Nov 08	N-body techniques, part 2 (PP)	–	
#22	Nov 10	N-body techniques, part 3 (PM and tree)	19.0, 19.4-19.6	
#23	Nov 15	Integration of PDEs, part 1 (ell & hyp)	19.0-19.1	
#24	Nov 17	Integration of PDEs, part 2 (hyp & par)	19.2	HW6 due
#25	Nov 22	Fluid dynamics, part 1 (eqns)	–	
–	Nov 24	no class (Thanksgiving)	–	
#26	Nov 29	Fluid dynamics, part 2 (methods)	19.3	
#27	Dec 01	Term project presentations	–	Project due
#28	Dec 06	Term project presentations	–	
#29	Dec 08	Term project presentations	–	

Note: check online for up to date Course outline.