Theory of Computation

CPSC 439-01,-02

Spring 2024

Description & Objectives

Introduction to the theory of computation. Automata theory; finite state machines, context free grammars, and Turing machines; hierarchy of formal language classes. Computability theory and undecidable problems. Time complexity; P and NP-complete problems. Applications to software design and security.

Prerequisites

CPSC 121 or MATH 320; MATH 270B or MATH 280; or Computer Science or Computer Engineering graduate standing.

G.E. Requirements

This class does not meet any CSU General Education requirements.

Instructor

Professor Kenytt Avery

Email: kavery@fullerton.edu

Office: Zoom meetings accessed through Canvas

After class in EC 007 (Monday), EC 032 (Wednesday), EC 044 (Thursday)

Office Hours:

- Tuesday 14:00-13:30 - “Walk-in” consultation via Zoom

Use the link labeled “Office Hours” in Canvas.

- Tuesday 15:30-16:15 - Scheduled Appointments via Zoom

    1. Sign up for a 15-minute slot using the Canvas calendar. See Scheduling private appointments for Office Hours for more information.

    2. Use the link labeled “Office Hours” in Canvas to enter the Waiting Room.

Meeting Information

Section 01:	Monday and Wednesday	13:00-14:15	CS 110B
Section 02:	Monday and Wednesday	14:30-15:45	CS 102B

Learning Goals

This course will use mathematical tools to explore the limits of computation, asking questions such as:

- What exactly does it mean to compute?
- How does a set of Boolean circuits turn into a computer?
- Are there alternative mechanisms for computation?
- Are some computational mechanisms more powerful than others?
- Is there anything a computer can't do?
- Are some computational problems harder than others to solve?
- How can we tell if this is a harder or easier problem?
- Are there limits to how quickly we can solve a problem?
- What if we found an algorithm that could solve hard problems easily?

Course Outline

Major topics are listed here. Consult the Course Website for details, including assigned reading, exercises, and projects.

1. Introduction.
2. Representing mathematical objects as bits.
3. Cantor's theorem and diagonalization.

4. Defining computation. Straight-line programs and Boolean circuits.
5. Representing programs as bits.
6. Functions with infinite domains. Finite automata.
7. Turing machines.
8. Lambda calculus.
9. Alternate models of computation. Universality.
10. Uncomputability.
11. Spring Recess.
12. Turing completeness as a bug. Language-theoretic security.
13. Introduction to complexity
14. Polynomial-time reduction.
15. NP, NP-Hard, and NP-Complete
16. P vs. NP.

Important Dates

CSUF's Academic Calendar documents all holidays and planned campus closures.

CSUF's Registration Calendars contain all major dates with respect to adding, dropping, and withdrawing from your classes.

March 23	Project 1 due
April 1, 3	Spring Recess - No class
April 13	Project 2 due
April 27	Project 3 (Written Assignment 1) due
May 4	Project 4 due
May 18	Project 5 (Written Assignment 2) due

Textbooks

Required

Introduction to Theoretical Computer Science, Boaz Barak. Available from «https://introtcs.org/».

Optional

The Algorithm Design Manual, Third Edition, Steven S. Skiena, Springer, ISBN: 978-3030542559

Understanding Computation, Tom Stewart, O'Reilly, ISBN: 978-1449329273

The New Turing Omnibus, Enlarged and Updated Ed., A. K. Dewdney, Holt, ISBN: 978-0805071665

Additional Reference Material

Many popular technical books may be read online through the campus's subscription to the O'Reilly Learning Platform. To access the library,

- Visit «https://libraryguides.fullerton.edu/oreilly».

  From outside the campus network, you may be prompted to log in through the library's WWW proxy.

- When you reach the Welcome page, choose "Not listed? Click here." from the drop-down menu, then enter your `@csu.fullerton.edu` email address.

In addition, the librarians at the CSUF Pollack Library have developed the Computer Science LibGuide to support students studying computer science.

Programming Language and Platform

The instructor will use the latest version of Tuffix to evaluate programming projects, and will detail any additional libraries or tools required for each assignment. It is the student's responsibility to ensure that the assignments execute to his or her satisfaction on the instructor's grading platform.

Announcements & Communication

Ordinarily, announcements will be made in class and be accompanied by an update posted to the course website.

Urgent announcements (i.e. regarding events before the next regular class meeting) will be made via Canvas. By default, Canvas will forward announcements to your `@csu.fullerton.edu` email address.

Please check regularly, read, and understand all messages sent to your `@csu.fullerton.edu` address and posted to the learning management system.

Email Communication

All email to the instructor **must** include your name, class, and section number, and should come from your `@csu.fullerton.edu` email account. You do not need to include your CWID.

The email account provided by CSUF is the only way the instructor may reach the student outside of class meeting times. Additionally, it is the only way the instructor knows that he is corresponding with a student enrolled in the class. Please refrain from using your personal email address and use your `@csu.fullerton.edu` email.

Typically, the instructor shall respond to your emails during business hours within 2 business days (i.e. Monday through Friday). If you have not received a response, it may be because you have not identified yourself adequately, or because the instructor believes that your concern has already been addressed (e.g., verbally, or as part of class discussion). If you are expecting a response and do not receive one within this timeframe, please double-check the requirements above, then re-send your request.

Grading

Plus and minus grading will not be used when determining final grades.

Graduate students that use this course on a graduate study plan must perform additional work and will be evaluated on a separate grading scale vis-à-vis their undergraduate counterparts.

Final grades are computed by first finding the average score in each category described in the second table below. All scores are normalized to a scale of 0 to 100 before being averaged. The average score for each category is then used to compute the weighted average according to the weights in the second table below.

At least one of the lowest quiz scores will be dropped.

Grade	% of Total Points
A	90% - 100%
B	80% - 89%
C	70% - 79%
D	60% - 69%
F	Below 60%

Category	% of Final Grade
Exercises	20%
Quizzes	30%
Group Projects	30%
Written Assignments	20%

Assignments

Assignments will be posted to the course website and discussed in class prior to the due date. There are approximately:

- 14 weeks of reading assignments
- 14 weeks of exercises

- 3 group projects
- 2 written assignments

Reading assignments are outlined on the course website, and it is the responsibility of the student to stay up to date with the reading.

Exercises such as solving problems or proving theorems will be assigned each week. Answers will be submitted through an online discussion forum, and you will be able to compare your work with classmates. Graduate students will occasionally be assigned an additional exercise to be completed at the same time. The instructor will review solutions in class the following week.

Major projects and written assignments are to be completed in groups. Each project description will include the assignment's grading rubric. Presentation, spelling and grammar can be worth up to 30% of a written assignment's grade. At least one project will include an additional section specifying work to be completed by graduate students in your group.

Late Submissions

All work must be submitted by the deadline posted in Canvas. If you have not finished an assignment, submit whatever work you have completed by the deadline for partial credit.

The following kinds of projects cannot be evaluated, and will be assigned a zero score:

- Late submissions.
- Email submissions.
- Source code that cannot be executed successfully.
- Input/output that is falsified or does not match the submitted source code.
- Submissions that are plagiarized or otherwise violate the collaboration guidelines.

Extra Credit

If any opportunities for extra credit arise, they will be detailed along with the applicable assignments, and will be offered to all students who meet the requirements.

Quizzes

Quizzes will take place at the end of major topics. All quizzes are to be completed individually, in-person, in class. No one will be allowed to begin taking the quiz after the first student has completed the quiz.

In general, missed quizzes cannot be made up, but at least one of the lowest scores will be dropped. If you miss additional quizzes, another quiz will be imputed as the average score of all quizzes that have not been dropped.

Attendance Policy

While strongly encouraged, attendance is not required. If you are sick, contact the instructor as soon as possible, but **do not** attend class.

Religious/Cultural Observance

Students who have religious or cultural observances that coincide with this class should let me know by e-mail. I strongly encourage you to honor your cultural and religious holidays. However, if I do not hear from you, I will assume that you plan to attend all class meetings.

Academic Integrity

Academic dishonesty will not be tolerated. You are encouraged to assist one another and discuss the course materials with your peers, but it is your responsibility to be aware of and follow the spirit of UPS 300.021 Academic Dishonesty.

Academic dishonesty includes such things as cheating, inventing false information or citations, plagiarism, and helping someone else commit an act of academic dishonesty.It usually involves an attempt by a student to show a possession of a level of knowledge or skill, which they in fact do not possess.

When you use information from outside sources, you must acknowledge the original author or source following standard scholarly practice. **You are not allowed to use any material from any website that provides solutions to the assignments given in class**.

By submitting work for evaluation, you acknowledge that you have adhered to the spirit of the university's academic integrity policy and that your submission is an original work by you or by other members of your group, if you were directed to work in groups.

- Third party work (code, artwork, etc.) may not be used in student work without prior instructor consent. Failure to gain and document instructor consent will be construed as willful academic dishonesty.
- When a third party's work is incorporated into student work after gaining instructor consent, failure to wholly document the work's origin, copyright and license will be construed as willful academic dishonesty.

Failure to follow the spirit of the academic integrity policy shall result in the following consequences:

- A zero grade for the assignment at a minimum, and a failing grade ('F') for the course in the case of repeated violations.
- Your name, your actions, and the evidence collected submitted to Student Affairs and the department chair.
- In addition, Student Affairs may levy their own disciplinary action.

Collaborating with other groups

Projects will be completed in groups as assigned by the instructor. You may work freely with your fellow group members, but must limit the input you get from sources *outside* your group:

- You may help each other understand the assignment and brainstorm general solutions, but each group must develop and submit their own distinct work.
- You may give each other technical support, for instance troubleshooting software installation.
- You must separate to develop your own detailed solution to the problem, and type in your own source code and project report.
- Given these requirements, any submissions with identical excerpts, or excerpts that are identical up to superficial rearrangements, will be considered highly suspect of plagiarism.

Responsible use of Generative AI

The following statement on responsible use of generative AI was partially generated by the AI assistant Claude from Anthropic PBC:

Generative AI tools like ChatGPT and GitHub Copilot offer useful capabilities for generating text, code, and other content. However, these tools have limitations, including the potential for factual errors, logical gaps, and other deficiencies in the generated output. Computer Science students in particular have a special responsibility to use them appropriately.

When using generative AI tools for course assignments:

- You must clearly identify any large blocks of generated text or code that you include in your submissions. This allows the instructor to distinguish your original work from the AI-generated portions.

- For tools like GitHub Copilot that generate small code snippets, you should make a reasonable effort to identify code that may have been generated. However, it is understood that it can be difficult to track Copilot's contributions when integrated into your normal coding workflow.
- Nevertheless, you take full responsibility for vetting, editing, and confirming the accuracy of any generated output. You must thoroughly review suggestions and rewrite/refactor as needed. If errors, bugs, or other issues are present, you will be graded accordingly.
- You may not present content generated by AI as your own original work or understanding. The goal is to assess your knowledge and skills, not those of the AI system.
- Overuse of generative AI, where it substitutes for doing your own work to gain understanding, will negatively impact your learning. While this may not immediately affect your grades, it will become apparent in the workplace when you lack the knowledge and skills that you were expected to gain from this course.

Relying too heavily on AI generation will lead to gaps in your skills and understanding that could embarrass you in job interviews and set you up for failure in roles that require the expertise implied by graduating with a degree in Computer Science.

The intelligent use of AI can augment your productivity and creativity, but they are a supplement to your own critical thinking and efforts to master the material, not a replacement. You remain solely responsible and accountable for the quality of work you submit.

Recording & Transcription of Class Content

Students are prohibited from recording class activities (including class lectures, office hours, advising sessions, etc.), distributing class recordings, or posting class recordings. This includes audio and video recording, as well as photography, scanning, and screen capture (including photos and screen captures of boardwork).

The only exceptions to this policy are accommodations mandated by the Americans with Disabilities Act or other federal or state laws. Eligibility for accommodations must be verified through the office of Disability Support Services as detailed in UPS 330.230 Recording and Transcription of Class Content by Students. It is the responsibility of students who require accommodation to notify the instructor.

This policy is in place to protect the privacy of students in the course, as well as to maintain academic integrity through reducing instances of cheating. Students who record, distribute, or post class materials in violation of this policy may be referred to the Office of Student Conduct. and be subject to student discipline, up to and including expulsion.

Acknowledgments

Portions of this syllabus are drawn from syllabi authored by Professors Michael Shafae and Anand Panagadan.

Appendix: Tips for Success

Taking Notes

As Computer Science students, this probably runs counter to your intuition, but it's true: the best way to take notes is to write them by hand on paper. For details, see Note-taking by hand: A powerful tool to support memory by Hetty Roessingh of the University of Calgary.

In order to encourage you to take notes by hand, the instructor does **not** provide copies of slides, class notes, or boardwork. If you wish to have a permanent record of anything that is said in class or appears on the board, you are expected to take notes unless you require accommodation for notetaking through Disability Support Services.

If you take good notes, please consider volunteering as a Notetaker. Not only might you be able to help a classmate, but incentives from DSS may include priority registration.

Getting Help

The best time to ask a question is in class or during office hours. If you don't feel comfortable asking a question directly, you can use email, online chats, and the LMS to help you ask your question. All questions are good questions and all questions deserve a good answer.

When writing an email, please use your first and last name in emails and include the course and section number to help your instructor understand the context of your question. Include your GitHub username if you would like your instructor to view your GitHub repository. (Remember to commit and push all the changes.)

Keep a notebook handy at all times because you never know when a question might come up. Write down your questions and bring them with you to class or office hours.

Provide as much detail as possible such as the title of the homework assignment, the command that you're trying to use, the output from the computer that doesn't make sense, and what operating system you are using. Note that if you are not using Tuffix, the amount of help that the instructor is able to provide may be limited.

Working with groups

For advice on dealing with differing levels of contribution in a group, read the following short article, especially the section "What this Group Should Have Done: Mirroring":

Barbara Oakley. 2003. Coping with hitchhikers and couch potatoes on teams [Student's Corner]. *IEEE Engineering in Medicine and Biology Magazine* 22, 5 (2003), 9–10. https://doi-org.lib-proxy.fullerton.edu/10.1109/MEMB.2003.9625655

Metacognition

The workload required of full-time adjunct faculty is such that grading will often be significantly behind. As a senior or graduate student, you are expected to be able to reflect on your own learning, self-assess your performance on the assigned work, and take corrective action when necessary (a process called *metacognition*, literally "thinking about thinking").

For more information about this process, see the following articles:

- Metacognitive Study Strategies from The Learning Center, University of North Carolina at Chapel Hill
- Better Grades Through Metacognition from the Office of Teaching, Learning, and Assessment, The Pennsylvania State University
- Thinking About Thinking: How Metacognition Can Help Your Grades by Olivia Croley and Dillon Murphy, Psychology in Action

Appendix: Department and College Information and Policies

Technical Proficiency

Technical proficiency in programming and software engineering should correspond to the prerequisite(s) of the course. Students are expected to be intimately familiar with their development platform of choice and be able to write and debug code in C++ or MATLAB at a level of proficiency that corresponds to the prerequisites of the course.

Students are expected to

1. Have basic computer competency which includes:
    a. the ability to use a personal computer to locate, create, move, copy, delete, name, rename, and save files and folders on hard drives, secondary storage devices such as USB drives, and cloud such as Google Drive (Titan Apps);
    b. the ability to use a word processing program to create, edit, format, store, retrieve, and print documents;
    c. the ability to use their CSUF email accounts to receive, create, edit, print, save, and send an e-mail message with and without an attached file
    d. and the ability to use an Internet browser to search and access web sites in the World Wide Web.
2. Have ongoing reliable access to a computer with Internet connectivity for regular course assignments
3. Access their CSUF student email account
4. Use Internet search and retrieval skills to complete assignments
5. Apply their educational technology skills to complete expected competencies
6. Utilize other software applications as course requirements dictate

Development Tool Resources

Students are strongly recommended to use the Computer Science Department's official GNU/Linux development environment, Tuffix. Tuffix is Tuffy the Titan's Linux distribution.

Professor Shafae has made a series of captioned videos to guide a student through the process of installing and using Tuffix.

Students using Tuffix should join the CSUF TUFFIX Slack workspace. Please use the `#general` channel to ask about troubleshooting, installing, and using Tuffix.

You can use your own computer, or borrow a computer from CSUF for free through the Long-Term Device Checkout process.

Students who need immediate access to computer resources may do so through the Virtual Computing Lab. Professor Shafae has recorded a video demonstrating how to access and use the service.

A general purpose CentOS (Linux) shell server is available through secure shell (ssh) and secure file transfer protocol (sftp). The hostname is `ecs.fullerton.edu`.

Three specialized GPU servers are available. The systems are named `aries.ecs.fullerton.edu`, `prudence.ecs.fullerton.edu` and `turing.ecs.fullerton.edu`. Access to these systems may be limited to the campus network; you may use a VPN to access them off campus. Each system has at least one Nvidia Titan X GPU. Please use Docker to create containers for your project. This is the only mechanism for GPU users to install software for their projects. These GPU shell servers are not available off campus. To log into them from outside CSUF's network, first login to ecs.fullerton.edu and then log into the GPU shell server of your choice.

To login to any of the shell servers, all you need is your CSUF Portal login and password. If your email address is `taylor@csu.fullerton.edu`, then your username is `taylor`. If you are using a command-line ssh client, then your command to connect to `ecs.fullerton.edu` will be ssh `taylor@ecs.fullerton.edu`; replace `ecs.fullerton.edu` with the appropriate target hostname. Your password is the same password as your CSUF Portal password.

Student Resources

Any student who wishes to discuss any concern may contact the assistant deans of the college. Assistant deans are student advocates who will help you navigate the university's policies and procedures and assist with resolving any conflict.

Assistant Director of Academic Advising - Maria Organista
CS-206A (657) 278-4110 morganista@fullerton.edu

College International Advisor - Sandra Boulanger
CS-206A (657) 278-3119 sboulanger@fullerton.edu

Appendix: University Information and Policies

Students with Special Needs

Please inform the instructor during the first week of classes about any disability or special needs that you may have that may require specific arrangements related to attending class sessions, carrying out class assignments, or writing papers or examinations. According to California State University policy, students with disabilities must document their disabilities at the Disability Support Services (DSS) Office in order to be accommodated in their courses. Additional information can be found at Disability Support Services, by calling 657-278-3112 or email dsservices@fullerton.edu.

Emergency Procedures

For your own safety and the safety of others, each student is expected to read and understand the guidelines published at Emergency Preparedness. Should an emergency occur, follow the instructions given to you by faculty, staff, and public safety officials. An emergency information recording is available by calling the Campus Operation and Emergency Closure line at 877-278-1712.

Instructional Continuity

Due to an event such as an epidemic or a natural disaster that disrupts normal campus operations, students must monitor the course website and their campus email address for any instructions and assignments that the instructor announces.

Laboratory Safety

Safety is no accident. Learning and following the appropriate safety practices and protocols is an integral part to all laboratory courses. Following the appropriate safety practices and protocols minimizes the chances of repetitive stress injuries, mishandling hazardous materials, and injury to self and others. Additional campus laboratory safety information regarding hazardous materials is online at Chemical/Laboratory Safety.

Code of Conduct

We shall endeavor to question, discuss, disagree, and debate without resorting to tactics of intimidation, harassment, or personal attack. Insensitive language, harassment, or disruptive

behavior shall not be tolerated. The University expects students to know the rules specified at «https://www.fullerton.edu/deanofstudents/conduct/students.php» and abide by them.