CSCI-561: Theoretical Foundations of Computer Science (I)

Fall 2019

1 Overview and Outcomes

Are there "laws of physics" for computing? Are there fundamental limits to what computers can do—and thus things computers cannot do? If so, what makes computational problems harder or easier, solvable or unsolvable? And when faced with a new computational problem, how can we determine its difficulty and solvability?

In this course, we will address such questions about the fundamental capabilities and limits of computation. In particular, we will answer the following:

- *What is a computer?*
  We will study different models of computation.
- *What can we compute?*
  We will define problems that are solvable/unsolvable using different models of computation.
- *How well can we compute?*
  We will analyze the performance capabilities and limits for various computational models and problems.

At many universities, courses on the *Theory of Computation* are purely theoretical, in essence, math classes. Here at Mines, we aim to blend theoretical rigor and practical application. Thus, in this course, we will both study fundamental results of computational theory and reduce theory to practice though projects that implement and apply key algorithms of theoretical computer science. Through the activities in this course, you will learn the following (Figure 1):

Remember: Know definitions of conventional objects in language and automata theory.
Example: Define a context-free grammar.

Understand: Describe computational problems using formal languages.
Example: Write a regular expression to find email addresses

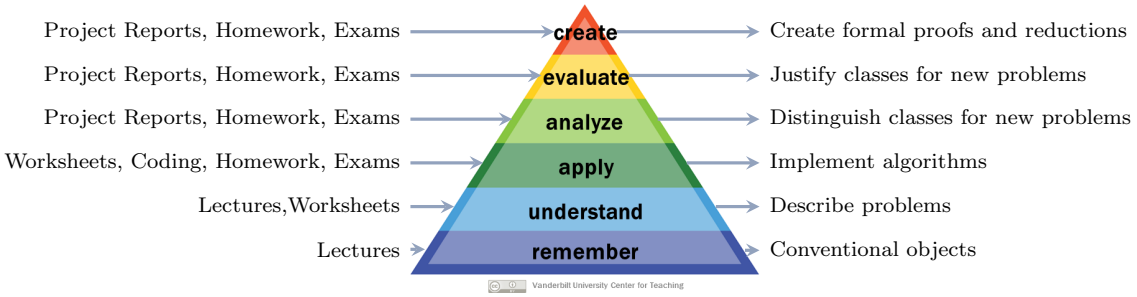

Figure 1: Bloom's Taxonomy of Learning Activities and Outcomes

Apply: Implement existing language theory algorithms
Example: Write code to convert a regular expressions to a finite automata.

Analyze: Distinguish suitable computational classes for new problems.
Example: Could we model some X as a regular language and/or solve via Boolean Satisfiability?

Evaluate: Justify the suitability of various computational classes for new problems.
Example: Why should we use context-free grammar vs. regular expressions to parse a particular file format?

Create: Develop proofs and reductions (algorithmic transformations) to characterize the required computation and/or solve a new problem.
Example: Create a formal proof that a file format cannot be parsed with regular expressions.

2 General Course Information

Instructor:	Neil Dantam	ndantam@mines.edu
Homework TA:	Hengrui Liu	liuhengrui@mymail.mines.edu
Project TA:	Matthew Schack	mschack@mymail.mines.edu

Textbook Michael Sipser. *Introduction to the Theory of Computation*.

Alternate References

- John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*.
- Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman. *Compilers: Principles, Techniques & Tools*.

Lisp References

- Peter Siebel. *Practical Common Lisp*. http://www.gigamonkeys.com/book/
- *Common Lisp HyperSpec*. http://www.lispworks.com/documentation/HyperSpec/Front/
- Paul Graham. *ANSI Common Lisp*.

Who should I email/contact?

- **Miscellaneous basic policy questions** (when is the midterm? when is an assignment due?): Re-read the syllabus, check Canvas for announcements and assignments, check the course website, and ask any additional questions on Piazza.
- **Help with assignments or course topics:** Piazza, TA office hours, or instructor office hours. Private post on Piazza if the matter should be hidden from other students (e.g., something about your code or questions about your grade)
- **Solutions to in-class exercises:** Slides with completed exercises will be posted to Canvas after the lecture.

- **Anything sensitive or confidential** (e.g., a health issue) Email the instructor about the issue and/or to schedule a meeting to discuss the issue.

3 Grading and Evaluation

The course score (percentage) will be computed as a weighted average of scores (points received over points possible) as follows:

Class Participation	5% (c)
Homeworks	10% (h)
Projects	35% (p)
Midterm Exam	20% (m)
Final Exam	30% (f)

$$\texttt{score} = .05\left(\frac{\texttt{c}_{\text{recv.}}}{\texttt{c}_{\text{poss.}}}\right) + .1\left(\frac{\texttt{h}_{\text{recv.}}}{\texttt{h}_{\text{poss.}}}\right) + .35\left(\frac{\texttt{p}_{\text{recv.}}}{\texttt{p}_{\text{poss.}}}\right) + .2\left(\frac{\texttt{m}_{\text{recv.}}}{\texttt{m}_{\text{poss.}}}\right) + .3\left(\frac{\texttt{f}_{\text{recv.}}}{\texttt{f}_{\text{poss.}}}\right)$$

Class Participation During most lectures, you will have a worksheet to practice the material. After the lecture is complete (i.e., we finish the set of slides corresponding to the worksheet), scan or photograph the worksheet and submit it on Canvas. Your participation grade will be based on making an honest effort on the exercises. You may omit submitting the greater of 2 worksheets or 10% of the total number of worksheets and still receive full credit for participation.

Midterm Exams A midterm exam will take place around the middle of the semester.

Final Exam A cumulative exam will take place during finals week.

Homeworks There will be several homeworks and exercises.

Projects There will a warmup plus two projects on applications of CS theory. The amount of code you will need to write is fairly small (a few hundred lines at most). However, you will need to think carefully about the relevant theory, math, and algorithms. Thus, **it is critical that you start projects early** so you have sufficient time to think through the required implementation and application (and ask questions if you get stuck).

Project 0 Warm-up project on programming environment and mathematical preliminaries.

Project 1 Finite Automata and Regular expressions.

Project 2 Propositional Logic and Boolean Satisfiability.

Letter Grades Letter grades will be based on a curve. It is expected—but not guaranteed—that score distributions will be normally distributed and letter grades will correspond to university and department norms. Assuming consistent, normal distribution of scores, the A/B cutoff will be approximately at the median score, and scores more than one standard deviation below the average may receive less than a B. However, skewed student effort or score distributions may result in correspondingly skewed letter distributions.

Written Work Format and submit your written work as follows. Improper submission or formatting may result in a penalty on assignments.

- For FERPA compliance, include a cover sheet on all written work that contains only your name and no answers or other work.
- Write your name on *every page* of all written work. If the work cannot be matched to you, you cannot receive credit for it.
- Include page numbers and total page count in written reports to ensure pages are properly ordered and no pages are overlooked.
- Handwritten work must be *clearly legible* to receive credit.
- Submit electronic reports, homeworks, etc. in PDF format. Do not submit word processor files because these are inconsistently formatted by different software.
- Work must be readable when printed in black and white.

Projects Expectations and Grading

- Projects will include a coding portion and a report portion.
- Code will be graded objectively. Code must produce the correct output to receive credit. Incorrect output, no output, compilation errors, or runtime errors will not receive credit. **Please double-check your submitted code to ensure that minor errors will not result in major test failures.**
- Code tests will include edge cases. Think through all possible conditions for your program.
- Report grading will evaluate your overall understanding for the project area.

Grading Corrections Grading changes will only be made for grading errors:

1. *Code:* An error in the grading environment or scripts incorrectly tested your code.
2. *Written:* The grader incorrectly understood your answer.

4 Laptop and Smartphone Policy

- Lecture slides will typically be posted in advance. You are welcome and encouraged to use your laptop or phone to follow along on the slides.
- Note-taking on laptops, tablets, etc. is welcome if you find it useful.
- Please refrain from using laptops, phones, etc. for non-class activities, e.g., email, web browsing, games, during classtime, as it is distracting to other students.

5 Tentative Schedule

(updated 2019-08-20)

Week	Date	Topic(s)
Week 1	Aug 20	Course Introduction and Math Review
Week 2	Aug 27	Common Lisp
Week 3	Sept 3	Finite Automata and Regular Expressions
Week 4	Sept 10	Regular Decision Properties
Week 5	Sept 17	Regular Closure Properties
Week 6	Sept 24	Application: Discrete Event Systems
Week 7	Oct 1	Context-Free Grammars
Week 8	Oct 8	Pushdown Automata
Week 9	Oct 15	Context-Free Languages
Week 10	Oct 22	Application: Context-Free Parsing
Week 11	Oct 29	Boolean Satisfiability
Week 12	Nov 5	Application: SATPlan
Week 13	Nov 12	Turing Machines
Week 14	Nov 19	Decidability
Week 15	Nov 26	Time and Space Complexity
Week 16	Dec 3	Lambda Calculus
Week 17	Dec 10	Finals Week

6 CS Department Course Policies

Academic Integrity All students are advised to be familiar with university policy on Academic Integrity. In addition, the following Collaboration Policy exists for all CS@Mines courses. This policy is a minimum standard; your instructor may decide to augment this policy.

1. If the project is an individual effort project, you are not allowed to give code you have developed to another student or use code provided by another student. If the project is a group project, you are only allowed to share code with your group members.

2. You are encouraged to discuss homework and final project assignments with other students in the class, as long as the following rules are followed:

   (a) You view another student's code only for the purpose of offering/receiving debugging assistance. Students can only give advice on what problems to look for; they cannot debug your code for you. All changes to your code must be made by you.

   (b) Your discussion is subject to the empty hands policy, which means you leave the discussion without any record (electronic, mechanical or otherwise) of the discussion.

3. Any material from any outside source such as books, projects, and in particular, from the Web, should be properly referenced and should only be used if specifically allowed for the assignment.

4. To prevent unintended sharing, any code stored in a hosted repository (e.g., on github) must be private. For group projects, your team members may, of course, be collaborators.

5. If you are aware of students violating this policy, you are encouraged to inform the professor of the course. Violating this policy will be treated as an academic misconduct for all students involved. See the Student Handbook for details on academic dishonesty.

Violations of this policy result in one of a range of punitive measures, from a zero score for an assignment, up to and including a course letter grade drop for all students involved. All issues of misconduct are reported to the Dean of Students. Academic misconduct associated with an exam grade will likely result in course failure.

Student Absences All students are advised to be familiar with university policy regarding the make-up of work missed due to excused absences. This policy may be found in the Bulletin.

Disabilities Accommodations The Colorado School of Mines is committed to ensuring the full participation of all students in its programs, including students with disabilities. The website http://disabilities.mines.edu outlines the university's disability services. Any student requiring accommodations must request Student Disability Services deliver each professor a *Confidential Letter of Required Accommodations* to ensure accommodations are met.

Discrimination & Harassment This course and all learning opportunities at Mines require a safe environment for everyone to be productive, develop professional practices, and to be able to share and learn without fear of discrimination or harassment. Discrimination or harassment of any type will not be tolerated. Sometimes harassment is unintentional, but regardless of intent the instructor will address any language or behaviors that might discriminate, stereotype, or promote harassment. If you witness discrimination or harassment of others, please bring it to the attention of Mines faculty so it can be addressed immediately.

Title IX is a federal law that protects individuals from discrimination based on sex and gender in educational programs or activities. Mines takes its Title IX obligations seriously and is committed to providing a campus community free from gender-based discrimination. Gender-based discrimination, including sexual harassment, sexual violence, stalking, and domestic violence, is prohibited within the Mines campus community. If these issues have impacted you or someone you know, you can appropriate resources here: http://inside.mines.edu/POGO-Title-IX. You can also contact the Mines Title IX Coordinator, Karin Ranta-Curran, at 303.384.2558 or krcurran@mines.edu for more information.

Learning Environment Fundamentally, I expect and require respect in this course for yourself, your classmates, and your instructor and TAs.

- Respect for yourself includes taking care of yourself physically and mentally and advocating for an environment that facilitates learning for you.
- Respect for your classmates includes recognizing and appreciating the diversity of backgrounds and experiences of your classmates and making it your interest to foster a learning environment for everyone; all are welcome.
- Respect for your instructors (as well as your classmates) includes not participating in disruptive or distracting behavior: talking, playing games, or web surfing during lecture, for instance, make it difficult for others to focus on the reason we are all here.

- Respect must be mutual to be effective; we (your instructors) and your TAs will be held to the same standards of respect.

Please let your instructor know if you become aware of an issue with the classroom (or out-of-classroom) environment with regards to these policies.