Computer Science 171: Cryptography

Course Format. Two classes of 1 hr and 30 minutes each every weak. Discussions: 1 class of 1 hr every week.

Course Textbook: The prescribed textbook for this course is Katz and Lindell's text Introduction to Modern Cryptography (not free, some copies available in the library). Here are some other excellent (and mostly free) resources:

- Boneh and Shoup's upcoming book: A Graduate Course in Applied Cryptography
- Lecture notes by Pass-Shelat, Bellare-Goldwasser and Bellare-Rogway
- Goldreich's book: Foundations of Cryptography Vol. 1 and Vol. 2 (some fragments are freely available)
- Nigel Smart's book: Cryptography, An Introduction
- Menezes, Oorschot, and Vanstone's book: Handbook of Applied Cryptography
- Rosulek's book: The Joy of Cryptography
- Ostrovsky: http://web.cs.ucla.edu/ rafail/PUBLIC/OstrovskyDraftLecNotes2010.pdf

Course Outline. Here is the list of topics covered in this class (schedule according to Spring 2019 semester):

	Date	Topic
1	Week 1	Introduction and overview. Private-key cryptography. The syntax of private-key encryption. The shift cipher.
2	Week 1	Elementary cryptanalysis and frequency analysis. Principles of Modern Cryptography.
3	Week 2	Modern cryptography: definitions, assumptions, and proofs. Perfect secrecy. The one-time pad. Limitations of perfect secrecy. A computational notion of security. Pseudorandomness and pseudorandom generators.
4	Week 2	Pseudorandom generators and stream ciphers. The pseudo-OTP. Proofs by reduction, and a proof of security for the pseudo-OTP.
5	Week 3	Security for multiple encryptions. Drawbacks of deterministic encryption. Chosen-plaintext attacks and CPA-security. Pseudorandom Functions.
6	Week 3	CPA-security from stream ciphers and block ciphers.
7	Week 4	Encrypting arbitrary-length messages: block-cipher modes of operation.
8	Week 4	Chosen-ciphertext attacks.
9	Week 5	Midterm I.
10	Week 5	Message integrity and message. authentication codes (MACs). Defining security for MACs. Constructing MACs.
11	Week 6	CBC-MAC. Authenticated encryption and generic constructions.
12	Week 6	Secure sessions. CCA-security. Hash functions and collision resistance.

13	Week 7	Birthday attacks on hash functions. Additional applications of hash functions.
14	Week 7	One-Way Functions and Implications.
15	Week 8	Practical constructions of stream ciphers.
16	Week 8	Practical constructions of block ciphers. Substitution-permutation networks (SPNs).
17	Week 9	Fiestel Networks. The data encryption standard (DES).
18	Week 9	Midterm II.
19	Week 10	Group theory. Cyclic groups. Hardness assumptions in cyclic groups: the discrete-logarithm assumption and Diffie-Hellman problems. Concrete Parameters.
20	Week 10	Drawbacks of private-key cryptography. The Diffie-Hellman key-exchange protocol and the public-key setting. Public-key encryption: syntax and definitions of security.
21	Week 11	Definitions of security for public-key encryption. Hybrid encryption and the KEM/DEM paradigm. El Gamal encryption.
22	Week 11	Digital signatures. The hash-and-sign paradigm. RSA-based signatures. DSA. Certificates and public-key infrastructures.
23	Week 12	Special Topic I: Identity Based Encryption.
24	Week 12	Special Topic II: Zero-Knowledge Proofs.
25	Week 13	Special Topic II: Zero-Knowledge Proofs.
26	Week 13	Special Topic II: Zero-Knowledge Proofs.
27	Week 14	Special Topic III: Cryptocurrency.
28	Week 14	Special Topic III: Cryptocurrency.
	Final Week	Final

Course requirements. The course requirements and weights for different parts of the grade:

- Homeworks (every week) - 20%
- Midterm I - 25%
- Midterm II - 25%
- Final - 30%

Prerequisites. The only formal prerequisite is a course on discrete mathematics (CS70). Specifically, we will assume familiarity with basic (discrete) probability and modular arithmetic. Students enrolled are also expected to have had some exposure to algorithms, mainly to be comfortable reading pseudocode and to be familiar with big-O notation.

Collaboration. You are encouraged to work on homework problems with fellow students; however, you must always write up the solutions on your own. Similarly, you may use books or online resources to help solve homework problems, but you must always credit all such sources in your write-up and you must never copy material verbatim. However, as a general rule of thumb, you should never possess solutions to exact homework questions other than those solutions you have written yourself. We realize that it is sometimes possible to stumble upon solutions on accident. If this happens, please cite the source and write up solutions in your own words.

We believe that most students can distinguish between helping other students and cheating. Explaining the meaning of a question, discussing a way of approaching a solution, or collaboratively exploring how to solve a problem within your group are types of interaction that we strongly encourage. But you should write your homework solution strictly by yourself so that your hands and eyes can help you internalize this material. At no time should you be in possession of another student's solution. You may discuss approaches but your solution must be written by you and you only. You should explicitly acknowledge everyone whom you have worked with or who has given you any significant ideas about the homework.

Further, it is your responsibility to ensure that your solutions will not be visible to other students. If you use Github or any other source control system to store your solutions electronically, you must ensure your account is configured so your solutions are not publicly visible. Many popular version control systems provide free repositories to students; staff members or fellow students can help you obtain one.

As a final note, we'd like to point out that collaboration on homework, while encouraged, can be detrimental to your learning if misused. In particular, avoid collaborations where you do not contribute enough to your own satisfaction. Such a collaboration not only cheats you out of an opportunity to learn through homework, but can also affect your confidence. If you feel that you are not contributing enough to your group, then try to spend time thinking about the problems alone before working with your group. If you end up solving the problem all by yourself that's great! And if not, you'll still be prepared to better contribute to your group. If you're ever in doubt about what constitutes academic dishonesty, always ask a TA or on Piazza.

Warning: Your attention is drawn to the department's policy on academic dishonesty and the campus honor code. In particular, you should be aware that copying or sharing solutions, in whole or in part, from other students in the class (or any other source without acknowledgment) constitutes cheating. Any student found to be cheating risks automatically failing the class and being referred to the Office of Student Conduct.

Relation to MA116: MA116 covers cryptography from a mathematical perspective while CS171 covers it from a complexity-theoretic perspective. Specifically, CS171 focuses on modern cryptography, which is distinguished from classical cryptography, by its emphasis on definitions, precise assumptions and rigorous proofs of security. In contrast, MA116 (based on what I have read about the course online) focuses on the use of number theory for cryptography; which CS171 does not delve much into. So, while there is overlap in terms of applications of interest, there is very little overlap in terms of the technical content.

In more detail, besides the special topics, proposed CS171's core course content can be broken into three parts: (i) private-key encryption and message authentication codes: with a focus on definitions and provably secure constructions from simpler assumptions, (ii) practical and theoretical constructions of symmetric-key primitives, and (iii) Public-key encryption and digital signatures: with a focus on definitions and provably secure constructions from number-theoretic assumptions. The main overlap of CS171 with MA116 is in terms of the number-theory background covered in the third part of the course, which CS171 will covers very briefly (in one class). Of course, students interested in learning about the number-theoretic aspects will benefit from additionally taking MA116. Finally, there is also some overlap in the constructions covered in the context of public-key encryption and digital signatures (both of which will be covered in the proposed CS171 in three classes). Here again, unlike MA116, the treatment of these schemes in CS 171 is from a complexity perspective.