Math 252: Mathematics of Public-Key Cryptography — Spring 2019

Instructor: Nathan Pflueger (pronounced "fleeger")
email: npflueger@amherst.edu
office: SMUD 401

Office hours (tentative):

Tuesday	1:45-3:15
Wednesday	1:45-3:15
Friday	1:30-2:30

(or by appointment)

Course webpage: `http://npflueger.people.amherst.edu/252/`

Times and locations: MTWF 9:00-9:50 SMUD 014

Come to office hours! I am happy to answer your questions and also talk about the course in general. Even if you don't have specific questions, you can come to review material, listen to other students' questions, or just to chat.

email policy: The best way to reach me with course questions (besides office hours) is by email. I generally reply to email within 24 hours. However, I often do not reply to email on weekends. I will also reply less quickly on Thursdays, which is the day I devote primarily to research.

What is this course all about? Math 252 is first and foremost a math course: the topics include a variety of notions from number theory, abstract algebra, and algorithms, and the heart of the course is learning aspects of these topics in an applied setting. The unifying theme of the course is the construction of the most commonly-used **public-key** cryptographic algorithms, and certain attacks against them that must be anticipated.

A wonderful feature of this subject (in my mind) is that it provides an excellent platform to practice **problem-solving with code**. Students will write code to try the algorithms in practice, perform experiments, and design and implement variations of the algorithms discussed in class. The official programming language for the course is Python, though everything we study would transfer readily to any other language.

No prior background in programming, number theory, or abstract algebra is expected. The course can serve as an introduction to these topics.

Course topics: We will cover mathematical problems underlying public-key ciphers and digital signatures, as well as algorithms to solve them. The subject presents an appealing introduction to several notions from number theory, abstract algebra, and algorithms. Topics include:

1. The discrete logarithm problem and Diffie-Hellman key exchange.
2. Integer factorization and the RSA cryptosystem.
3. Digital signatures.
4. Elliptic curves and related cryptosystems.
5. The NTru lattice-based cryptosystem.

Throughout the course, we will discuss both *cryptography* (encryption and signing algorithms) and *cryptanalysis* (algorithms to break cryptosystems).

ALICE SENDS A MESSAGE TO BOB SAYING TO MEET HER SOMEWHERE.
UH HUH.
BUT EVE SEES IT, TOO, AND GOES TO THE PLACE.
WITH YOU SO FAR.
BOB IS DELAYED, AND ALICE AND EVE MEET.
YEAH?
I'VE DISCOVERED A WAY TO GET COMPUTER SCIENTISTS TO LISTEN TO ANY BORING STORY.

Expectations: Students should expect to spend 8 to 12 hours per week outside of class on this course, between studying their notes and the textbook, working on written homework, and designing and writing code. If you are new to programming, working on that skill will account for a lot of your time (but it will pay very high dividends, as my previous students repeatedly tell me!). Some assigned problems will be quite challenging, and **you do not need to complete all problems to earn a good grade in the course**. However, I recommend attempting all assigned problems, and studying the solutions after they are posted.

You will sometimes be expected to take the initiative in looking up information about the tools you need, especially when programming. If you can't find what you're looking for, though, I will of course be happy to help.

The nature of a course like this, which connects to several different disciplines, is that some students may have an easier time with the material than others (e.g. if they already know how to program, or already know some abstract algebra). If it seems like the course is easier for some of your classmates, do not be discouraged! This just means that you have a lot to gain from the course. I hope that by the end of the course all students will reach the same place, and you will all be versatile and valuable problem-solvers wherever you take your skills from this course.

Textbook: *An Introduction to Mathematical Cryptography, Second Edition*, by Hoffstein, Pipher, and Silverman. With Amherst credentials, you can download the entire book for free, or to print a cheap paperback copy, at the following link.

`https://link.springer.com/book/10.1007/978-1-4939-1711-2`

Prerequisites: A course with proofs, such as Math 220/221 or 271/272, or instructor permission.

Tips:

- **Come to office hours!** I am happy to answer your questions and also talk about the course in general. Even if you don't have specific questions, you can come to review material, listen to other students' questions, or just to chat.
- **Review early and often.** You should constantly be looking over your notes and keeping the big picture in mind. Arrive each day in class with a sense for where we are.
- **Keep a positive attitude.** Learning is a long process, and you will struggle often. Remember that struggle and difficulty is how you grow. Don't be afraid to talk to me about whatever difficulty you're facing. I want all of my students to be successful and deepen their mathematical skill and appreciation.
- **Practice, practice, practice.** Start early on homework, and let hard problems simmer in your head. Try unassigned problems in addition to homework. Read the book, and *read actively*, always questioning, summarizing, and interpreting what's on the page.

Structure and grading: There will be weekly homework assignments, two midterm exams, and a final exam. The dates of all exams, and their share of your final grade, are listed below. There is no set curve or grading cutoffs, but most likely the median grade will be around a B.

Written homework	20%	
Programming homework	20%	
Midterm 1	15%	Wednesday 3/6
Midterm 2	15%	Wednesday 4/24
Final exam	20%	Date/time TBA
Your best exam	10%	(added to its original weight)

Late homework: Both written and programming homework will be **due at 10pm** on Wednesday evenings, except on exam days, via an online system called Gradescope. To allow for technical difficulties or other last-minute issues, Gradescope will allow you submit homework after the deadline, however your score will be reduced by 2% per hour after the deadline (scaled continuously, e.g. being fifteen minutes late results in a 0.5% deduction). Please try to turn in your work by 10pm (I don't want to be responsible for lost sleep!), but don't worry about short delays.

I do not grant extensions for any reason. However, to compensate for illness and other emergencies, your **lowest two written homework scores, and your lowest two programming homework scores, will be dropped**. If you cannot make a due date due to an emergency, my advice is to simply skip the assignment, study and understand the posted solutions to catch up, and focus on keeping up with the new material in the course. You do not need to apologize or provide any reasons for skipping an assignment or turning it in unfinished; please choose what is best for your time, health, and well-being.

Missed exams: if you are ill or an emergency arises near an exam, notify me as soon as possible. If you have a time conflict with an exam, notify me as soon as possible, and **at least one week in advance** (exam dates are listed above).

Accommodations: I strive to make this course welcoming to all students. If you would like to discuss your learning needs with me, please schedule a meeting so that we can work together to support your academic success. Anyone who may require an accommodation based on the impact of a disability should contact me to make arrangements. I rely on Accessibility Services for assistance in verifying the need for accommodations and developing accommodation strategies, so I encourage you to contact them at `accessibility@amherst.edu` or 413-542-2337.

Intellectual responsibility:

- **Homework:** Mathematics is a collaborative subject; open and generous communication is one of its core values. Therefore you are strongly encouraged to work with other students, ask many questions, and learn from as many people as possible. However, you must write up the solution yourself. **All your submitted work must be your work, written in your own words,** with the exception of Mathematica labs, which are written in small groups. Copying solutions from other students, solutions manuals, or online databases is plagiarism; such copying will result in a 0 on the assignment and will be reported to Community Standards. You are also expected to **list each person your worked with** on the front of your homework assignment.

- **Exams:** You will be allowed **one page of notes (front and back)** for each exam. No calculators or other aids are permitted. Cell phones should be stowed out of sight during exams. Use of cell phones or other devices during the exams (except in emergencies) will be grounds to receive a 0 on the exam. You are bound by the college's honor code, and all work must be entirely your own on exams.