Duke University TIP Summer Studies Program 2016

Cryptography, Codebreaking, and the Mathematics of Spying
Rice University, Term I
Instructor: Kenan Ince
TA: TBA

Required Texts: (provided by Duke TIP)
Coursepack: Cryptography, Codebreaking, and the Mathematics of Spying
Enigma by Robert Harris (Ballantine, 1996)
The Shadow Factory by James Bamford (Doubleday, 2008)
Learn to Program: Second Edition by Chris Pine (Pragmatic, 2009)

We will mainly be working with Simon Singh's The Code Book and the CryptoNotes coursepack. You should feel free to read out of The Code Book as much as you like. I won't have time to cover all of it. (Of course, you have no required homework in this class.)

The mathematics of the course will mainly come from the Coursepack. I still expect you to take notes in the class, because the act of taking notes makes it easier to remember things and because I will combine the Coursepack with other resources.

Course Description and Objectives:
The purpose of this course is to introduce you to the complex and exciting world of secret communication – and the discovery of secrets through cryptanalysis! Beginning with the simplest ciphers of antiquity, we will trace the historical development of cryptography (the science of enciphering messages) and cryptanalysis (the science of breaking ciphers and decrypting secret messages) through the medieval period, the age of radio communications, and the computer age.

Today, encryption plays a central role in the security of distributed computer networks, such as the Internet, and in our ability to conduct commerce and communicate securely over great distances. In this course you will develop a hands-on understanding of how computers handle encryption and explore the mathematical foundations of modern cryptology.

The objectives of this course are for students:

• to learn the ins and outs of various cipher systems and analyze their strengths and weaknesses;
• to acquire a strong mathematical background in a range of topics applicable to cryptography;
• to explore the ethics involved in cryptology from a historical and a practical perspective;
• to gain firsthand experience translating algorithms into working computer code; and,
• to have fun exploring the world of codes and ciphers.

Disability Support: If you have a documented disability that will affect your work in this class, please contact me within the first week to discuss accommodations.
Assignments and Evaluation:
You will receive a grade based on your participation in this course. Your grade will also depend upon your performance on written problem solving, computer programming exercises, group codebreaking activities, journaling, and reading fiction and nonfiction about cryptography and codes. There will be one exam to evaluate your progress. You will be allotted time to complete an independent project on a topic of your choice.

Electronic Devices
I believe it is near-impossible to take math notes on a laptop unless it is a tablet equipped with a pen (which is allowed). For this reason, I don’t allow non-tablets to be used in class. You won’t need a computer to solve the problems in this course.

Taking Notes
Recent research has found that taking notes word-for-word on what is done in class leads only to surface-level learning, which disappears when new information is introduced. If you’ve ever completely forgotten what you’ve “learned” in a class months after taking it, you’ve experienced this firsthand. However, taking notes in a manner that allows you to process the information you take in allows you to learn the material in lasting ways, even if you never go back to review those notes in the future (although, of course, you should). I highly recommend paraphrasing in your notes what is said in class, how a problem is solved, etc. For these reasons, taking notes is a requirement of this course.

Basic Materials We’ll Need:
- 3-ring binder or folder with pockets for organizing handouts and assignments
- a notebook with at least 100 pages of 4×4 square per inch graph paper (for note-taking and cipher exercises)
- a college- or standard-ruled notebook with at least 50 pages (for journaling)
- basic scientific calculator
- pens, pencils, and eraser(s)
- USB flash drive for saving programs and other data

Journals
- Every day, the first 5-10 minutes of evening session will be devoted to journaling. You'll be asked to copy a specific question and respond to it. If you finish before time is up, keep writing on any topic you choose.
- Although we'll take up your journals frequently, read the entries, and respond to them, the writings in journals will not be graded or used in grading.
- You can fold the pages of journal entries you want to keep private.
- I will usually pick your journals up at the beginning of class every few mornings, but today I'll pick them up after you do your first assignment and pre-assessment. I want you to do these activities not in your groups (this is probably the only thing I won't have you do in your groups!).
- When I pick up assignments/papers, I'll have you put them inside your journals when I pick them up.

Rules for group discussion
Research shows that group work is a highly effective way to spend class time. Benefits include:
• Everyone has more opportunity to participate.
• Students can learn from each other in ways they can’t from a textbook or instructor.
• The best way to learn something is to try to explain it to someone else.
• Talking about problems with your colleagues makes you more comfortable with the language of mathematics.
• Math is more fun this way!

Ground Rules for Groups: You will get the most out of your group work if you stick to these rules.

1. Criticize only ideas, not people. And criticize constructively! Respect each other’s thoughts.
2. Be a team player. Question and participate.
3. Be willing to make mistakes or have a different opinion. Don’t belittle others who make a mistake.
4. We are each responsible for what we get from the group experience. And everyone is responsible for the success of the group!
5. Don’t let others do all the work.
6. Don’t do all the work for everyone else.
7. One person talks at a time.
8. Keep to the current topic.
9. Take notes, even if there is a scribe.
10. It is dishonest to pretend that you understand or agree when you don’t.

Course Outline (subject to change based on goals/needs of the class)

Week One

**Monday, June 12: Ancient and Elementary Ciphers (c. 500 – 1 B.C.)**
Early communication, steganography, and telegraphy
The Greek scytale: simple transposition
Julius Caesar’s shift cipher
Simple substitution ciphers
Algebra review

**Tuesday, June 13: Elementary Cryptanalysis (c. A.D. 900)**
The Arab invention of frequency analysis
Solving simple substitution ciphers
Sir Arthur Conan Doyle’s “The Adventure of the Dancing Men”
**Intro to programming Solving shift ciphers using programming**
**Note:** No prior programming experience is assumed by this course.

**Wednesday, June 14: Polyalphabetic Systems (c. A.D. 1550)**
Vigenère ciphers
Solving Vigenère ciphers with known key length
_Krypto_: an unsolved mystery
**Programming Vigenère Ciphers**
Thursday, June 15: Advanced Cryptanalysis (c. A.D. 1850)
Solving Vigenère ciphers using Kasiski's method
Implementing Vigenère ciphers using modular arithmetic
**Programming and Cryptanalyzing Vigenère Ciphers**

Friday, June 16: Modern (20th Century) Cryptography
*Modern Marvels: Codes* (short film)
Reading: *Enigma*
One-time pad cryptosystems

Saturday, July 17
Encoding affine ciphers
Factoring, prime factorization, and greatest common factors (the Euclidean algorithm) Modular inverses

**Course Outline: Week Two**

Monday, June 19
Independent project selection/work
The extended Euclidean algorithm
Modular inverses and their use in solving a congruence

Tuesday, June 20
Solving modular congruences
Solving systems of congruences
Cryptanalyzing affine ciphers
Enigma machines
Independent project work
**Breaking Master Locks using modular arithmetic**

Wednesday, June 21
Independent project work
Probability and its importance in cryptology
Friedman’s index of coincidence (determining plaintext language, polyalphabeticity, and key length)
Reading: *Enigma*
Review for Exam 1
Solving Vigenère ciphers using the Friedman test
**Cryptanalyzing Vigenère ciphertext using index of coincidence Enigma cipher challenge**

Thursday, June 22
**Exam 1**
Independent project work
Navajo code talkers
Computer cryptography
Binary codes
Programming: Friedman test

Friday, June 23
Independent project work
Computer security, encryption, and authentication
Genetic fingerprinting
Diffie-Hellman key exchange
Movie: WarGames

Saturday, June 24
Group presentations

Course Outline: Week Three

Monday, June 26
Introduction to public key encryption
Euler’s phi function
Multiplication by repeated squaring
Reading: Enigma

Programming: Modular inverses and GCD

Tuesday, June 27
Intro to mathematical induction
Fermat’s Little Theorem
The NSA and surveillance

Computer Programming: Finish Inverse and GCD

Wednesday, June 28
Euler’s Theorem
How RSA encryption works
Debate: is the NSA’s surveillance of Americans justified?

Computer programming: modular exponentiation

Thursday, June 29
Computer Programming: RSA Algorithm
Matrices
Reading: Enigma
The Hill cipher
Cryptanalysis of the Hill cipher

Friday, June 30
Individual conferences with instructor
Crypto Treasure Hunt
Additional topics (based on time and student interest)
Letter frequency