Applied Cryptography
CPEG 472/672
Lecture 1A

Instructor: Nektarios Tsoutsos

About the instructor

- Assistant professor
  - ECE (primary), CIS (joint)
- Research areas
  - Cybersecurity
  - Applied cryptography
  - Hardware security
  - Embedded systems
  - Trustworthy computing
  - Privacy outsourcing

TWC

Introduce yourselves

- Name?
- Degree/Academic Program?
- Advisor?
- Crypto background?
- Programming background?
- What are you hoping to learn in this course?
- What will be the biggest challenge?

Instructor Assistants

- Charles (Chaz) Gouert
  - PhD Candidate, ECE

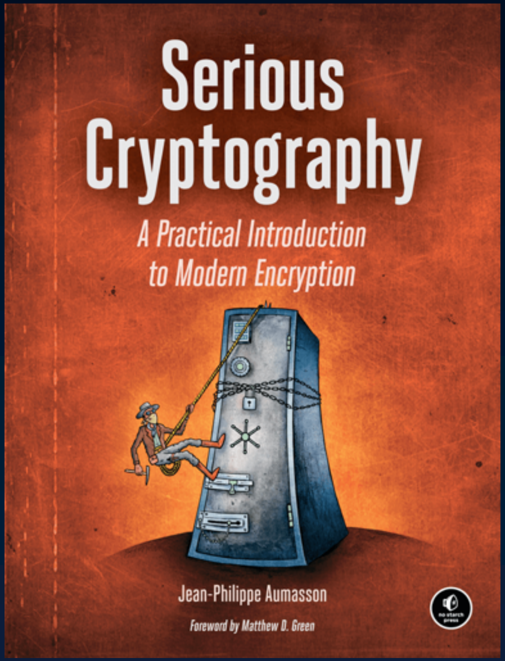
- Dimitris Mouris
  - PhD Candidate, ECE

Admin

- Lectures
  - Time: Tuesday & Thursday 2:00–3:15pm
  - Location: ISE 417
- In-class practice
  - Laptop required for hands-on exercises
- Reading
  - Review assigned material before class
- Office hours
  - By appointment: tsoutsos+crypto@udel.edu

Admin

- Textbook
  - Serious Cryptography by J.-P. Aumasson
    - ISBN: 9781593278267
    - Required textbook
  - Understanding Cryptography by C. Paar (optional)
    - ISBN: 9783642041006
  - Available at UD bookstore
- Online resources
  - CANVAS (courses/1496363)


Serious Cryptography
A Practical Introduction to Modern Encryption
Jean-Philippe Aumasson
Foreword by Matthew D. Green

Grades

- Final Exam: 25% (May 21, 2020)
- Midterm Exam: 15% (March 26, 2020)
- Homework Assignments: 50%
- Participation & in-class exercises: 10%
- Read the course policies
  - Late submission policy etc.
  - Academic integrity (very important)
- **Curved grading**

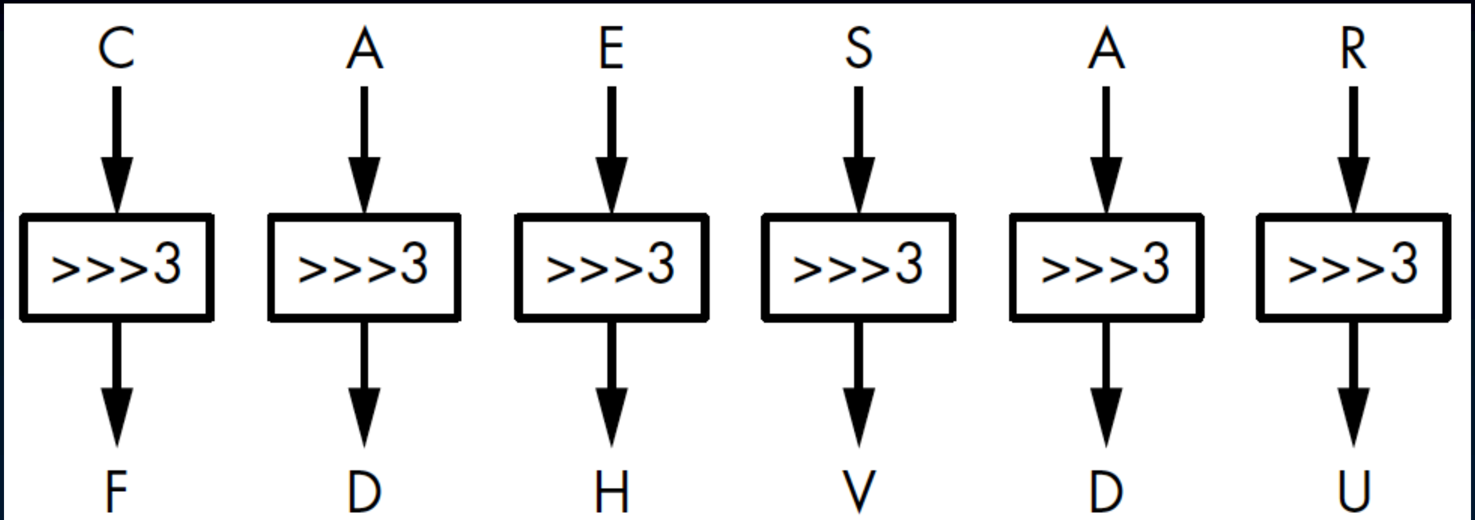
Syllabus

- In this course you will learn about:
  - Basics of encryption
  - Randomness generation
  - Security notions
  - Block and stream ciphers
  - Hash functions and keyed hashes
  - Authenticated encryption
  - Public key cryptography and elliptic curves
  - Homomorphic encryption
  - Key exchange

What is encryption?

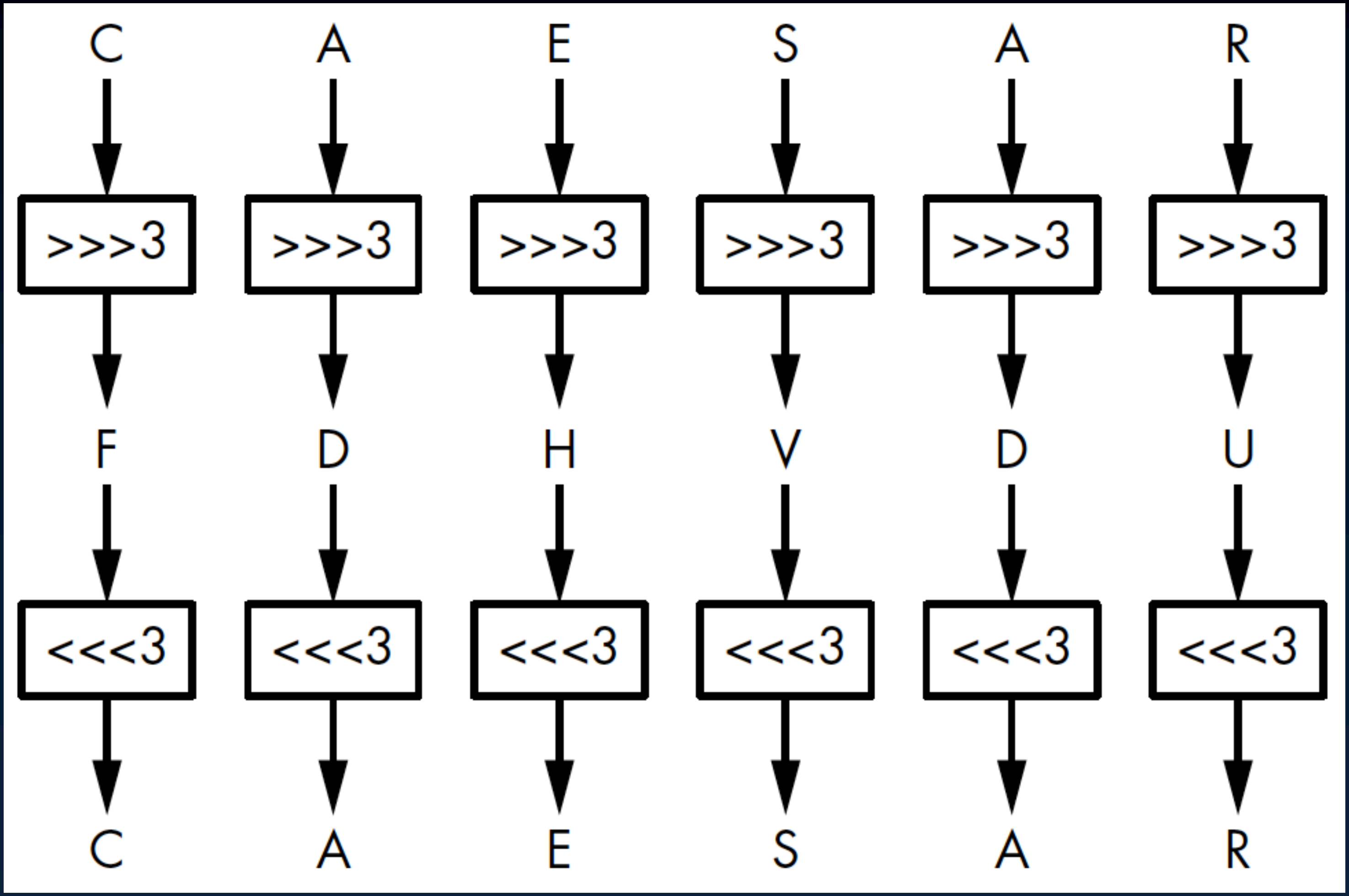
- Make data incomprehensible
  - Confidentiality
- Uses an algorithm called cipher
  - Inputs: Key (k), Plaintext (ptxt)
  - Output: Ciphertext (ctxt)
  - Symmetric, asymmetric (or public key)
- ctxt = Enc(k,ptxt)
- ptxt = Dec(k,ctxt)

Classical ciphers: Caesar cipher



- Encrypt: Rotate right by 3
  - Wrap around if needed

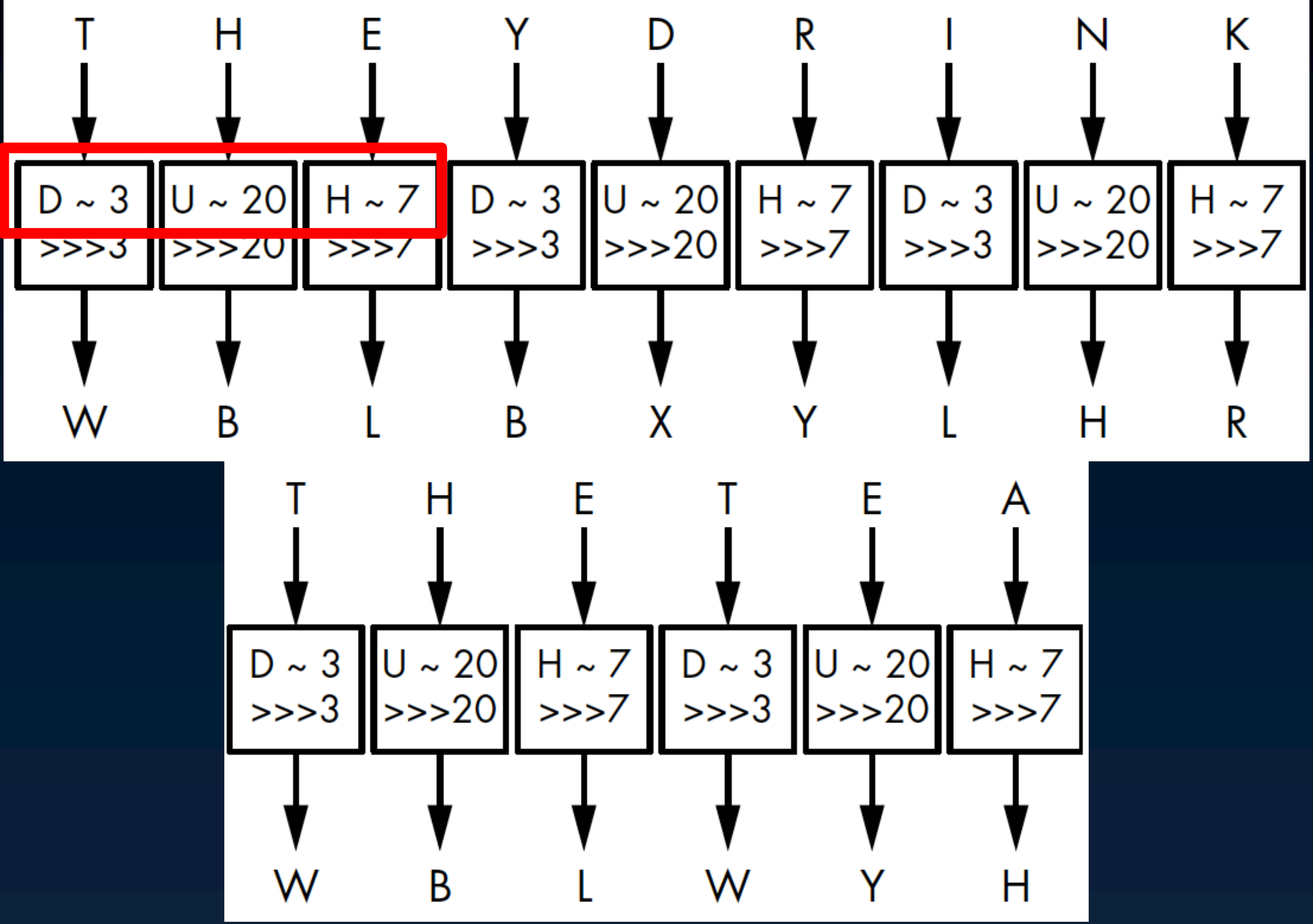
Classical ciphers: Caesar cipher



Classical ciphers: Caesar cipher

- How to break Caesar?
  - ?
- What are the possible keys?
  - ?
- What if the rotation amount is variable?
  - Each index is rotated by a different amount
  - This is defined by a key

Classical ciphers: Vigenere cipher



Classical ciphers: Vigenere cipher

- Is this more secure?
  - WBL appears twice!
  - Interval is 9 letters
- What does this mean?
  - Key length (DUH here) divides 9
- Other attacks?
  - Frequency analysis
  - Uneven distribution of letters in ptxt
- Vigenere better for short/shortlived ptxt

Two components of ciphers

- (1) A permutation
  - Transformation with a unique inverse
- (2) A mode of operation
  - Process ptxt of arbitrary size

Permutations

- Letter substitution in classical ciphers
  - Rotation by some amount
  - Cannot be just any substitution
    - Can I substitute A with D and B with D?
- Desirable properties
  - The permutation of inputs should be determined by a secret key
  - Different permutations for different keys
  - The permutations should look random

Modes of operation

- How to encrypt long messages?
- Ensure that repeating patterns in the plaintext disappear in the ciphertext
  - Should not reveal duplicates (leaks info)
- Concerns:
  - If you find patterns in a ctxt, it is possible to perform frequency analysis
  - Reusing the same key across different messages reveals patterns across ptxts

Vigenere with longer keys?

- Would Vigenere be secure if the key is as long as the message?
  - Key = KYN
  - Ptxt1 = TIE   Ctxt1 = DGR
  - Ptxt2 = PIE   Ctxt2 = ZGR
- Is there a problem here?
- Both end with GR
  - This exposes similarities between the ptxts

The problem with classical ciphers

- The number of possible permutations can be very large
  - What is the number of permutations in the English alphabet?
  - ?
- Classical ciphers use only a fraction of these permutations
  - The cipher description is too simple
- Can we define secure permutations?

Perfect Encryption: OTP

- One time pad
- ctxt = ptxt XOR k
- Requirements for k
  - K should be a long as ptxt
  - K should be random
- Reusing k reveals relationship between plaintexts
  - Two time pad

Reading for next lecture

- Aumasson: Chapter 1