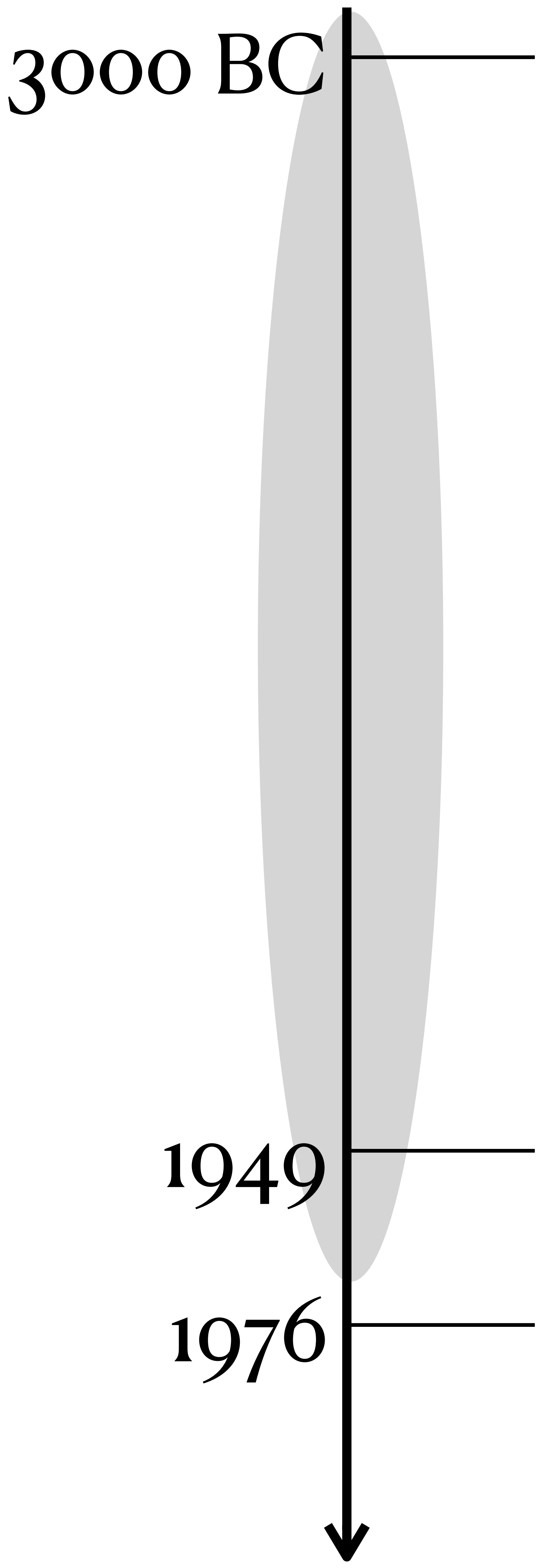
Portland State University

W'23 CS 485/585

Intro to Cryptography

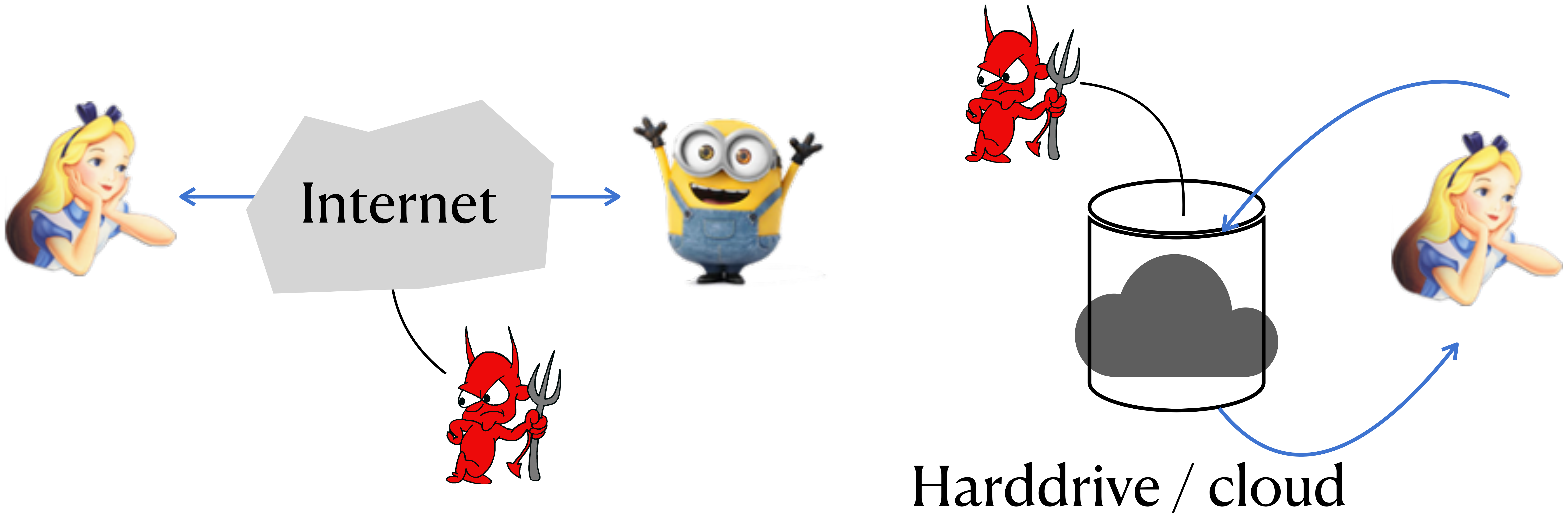


Concise Oxford English Dictionary

Cryptography is the **art** of writing or solving codes (ciphers)

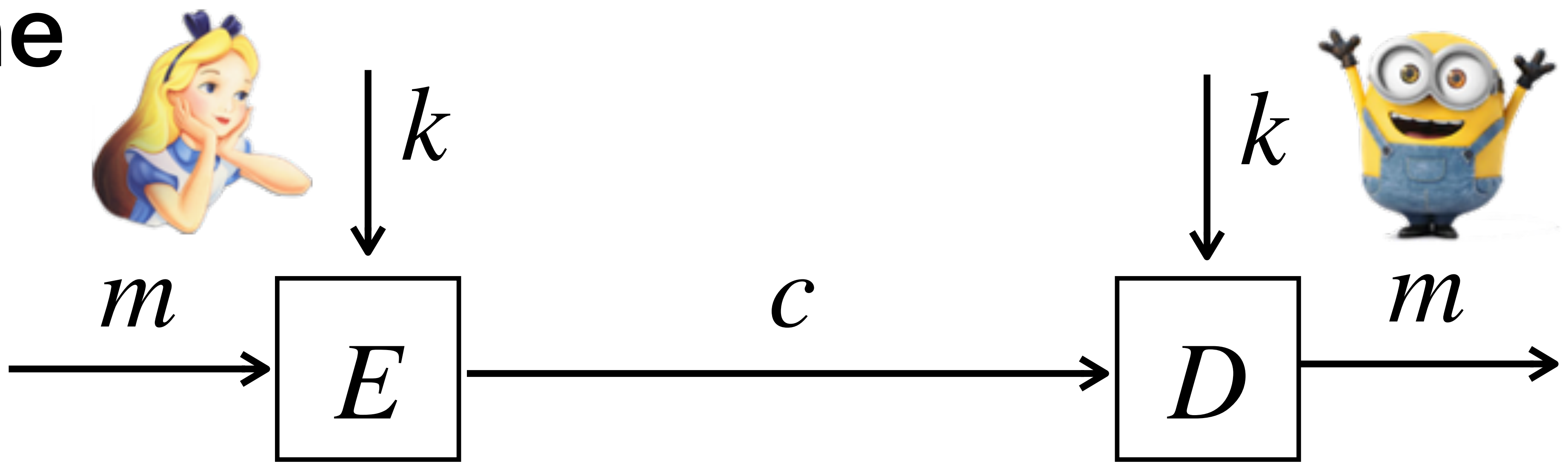
... for military activity and gossip

- 2 typical scenarios of "secret writing"



Private-key (symmetric) encryption

- Call a cipher an encryption scheme
- Syntax of a private-key encryption scheme



  - k: private key (secret key), shared between sender/receiver
  - m: plaintext (message)
  - c: ciphertext
  - E : encryption (encode) algorithm, $(k, m) \mapsto c$
  - D : decryption (decode/decipher) algorithm, $(c, k) \mapsto m$

Ceasar's cipher

- **Example**

$$m = \text{cryptoisfun}$$
$$c = \text{fubswlvixq}$$

- **Rule**

$$\text{a b c d} \ldots \text{x y z} \;\downarrow\; \text{d e f g} \ldots \text{a b c}$$

$$\{a, \ldots, z\} = \{0, \ldots, 25\}$$

- $k = 3$ fixed
- $E(m_i) = (m_i + 3) \bmod 26$

★ **Easy to break if we know it's encoded by Ceasar's cipher.**

Kirchhoff's principle

The cipher method must **NOT** be required to be secret, and it must be able to fall into the hands of the **enemy** without inconvenience.

- **Security should rely solely on the secrecy of the key**

  1. Much easier to secure & update a short key than complex enc/dec algorithms.
  2. Public scrutiny makes a cipher more trustworthy.
  3. Easier to maintain at large-scale.

★ **Only use standardized cryptosystems whenever possible!**

Ceasar ++: shift & substitution cipher

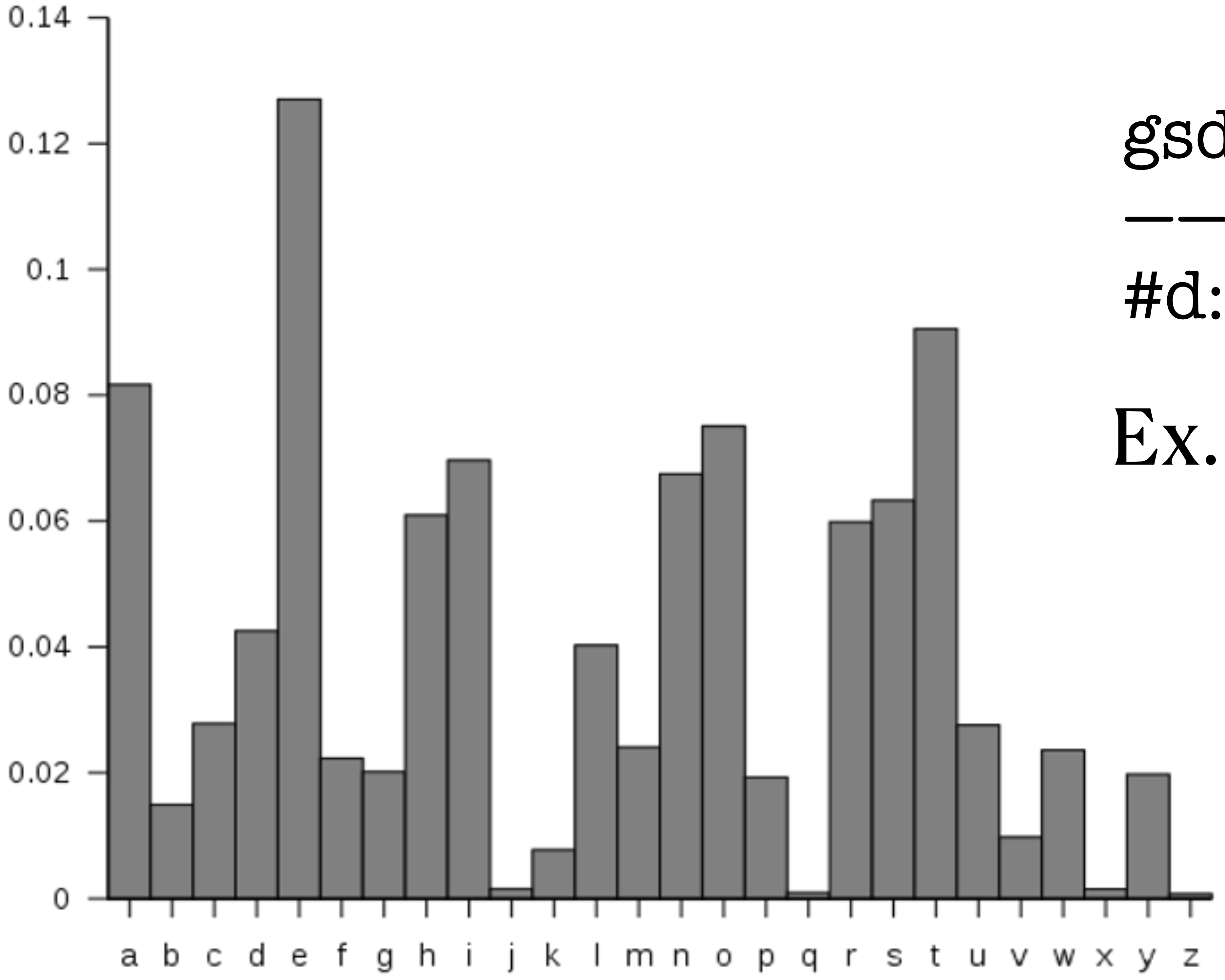
- **Shift cipher** $\{a, \ldots, z\} = \{0,\ldots,25\}$ $\{a, \ldots, z\} = \{0,\ldots,25\}$
  - Pick $k \in \{0,\ldots,25\}$ and keep it secret
  - $E(m_i) = (m_i + k) \bmod 26$
  - $k = 3$ fixed
  - $E(m_i) = (m_i + 3) \bmod 26$
  - Only 26 possibiities, brute-force search a key! Ex. Decipher "dszqupjtgvo".
- **Substitution cipher**
  - k defines a permutation on the alphabet.
  - Ex. encrypt "cryptoisfun" under this key.
  - Subsumes Shift Cipher as a special case.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
x	e	u	a	d	n	b	k	v	m	r	o	c	q	f	s	y	h	w	g	l	z	i	j	p	t

- **How many possible keys?** $26! \approx 2^{88}$

Breaking sub cipher by frequency analysis

- **Sub cipher preserves frequency:** one-to-one correspondence.
- **Frequency distribution in English language is publicly known.**
- **Typical sentences close to average frequency distribution.**



gsd uvpsdh cdgsfa clwg qfg ed hdylvhda gf ed wduhdg

———————————————————————————

#d: 18, #g:14, #q: 9

Ex. Decipher it by hand or online solver.

Poly-alphabetic shift cipher

- A.k.a. Vigenère cipher
  - Key k: a string of letters
  - Encrypt E:

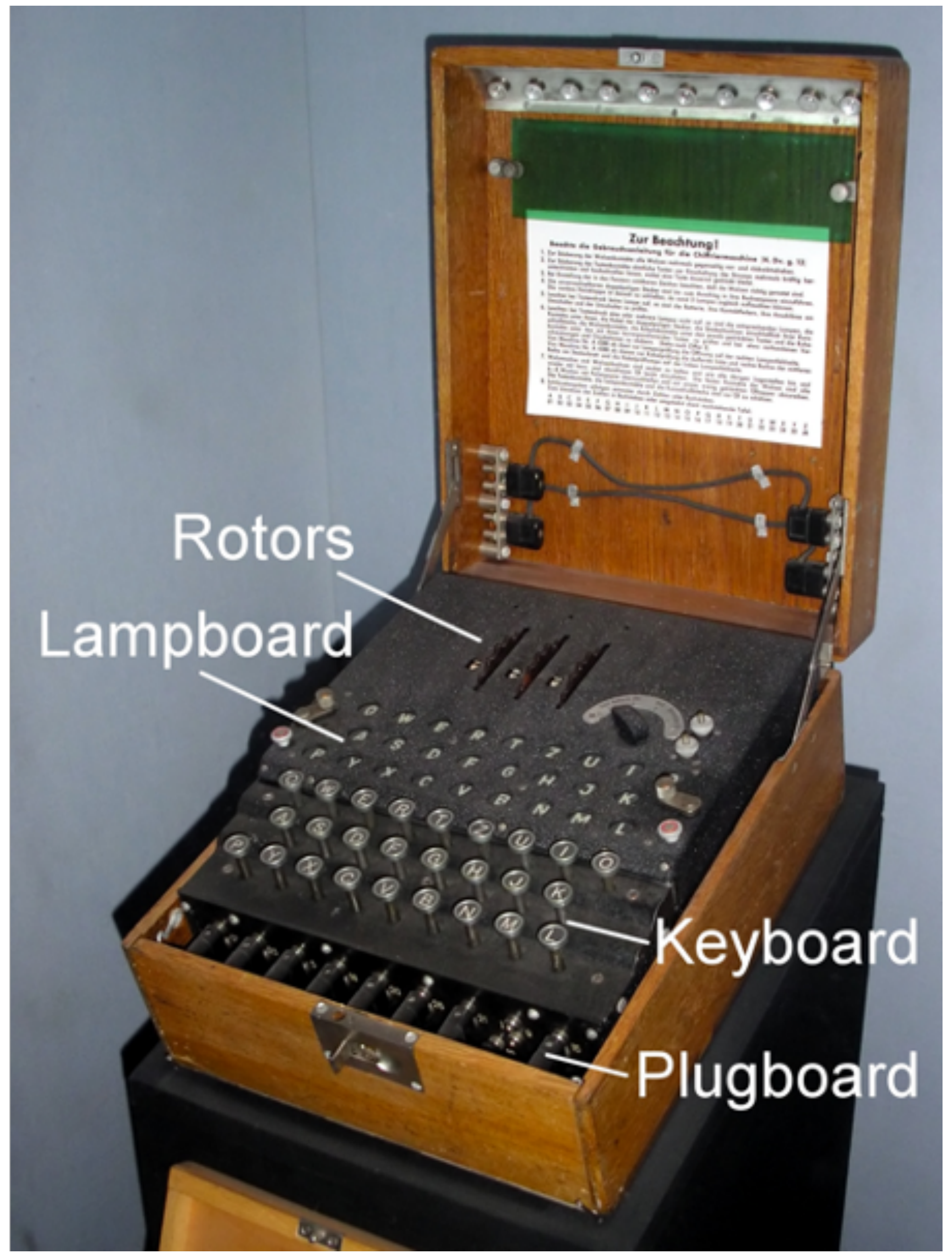
Key	psu psu psu psu psu psu
Plaintext	cry pto isf una ndc ool
Ciphertext	rjs eli xkz jfu cvw dgf

  - Considered “unbreakable“ for > 300 years.

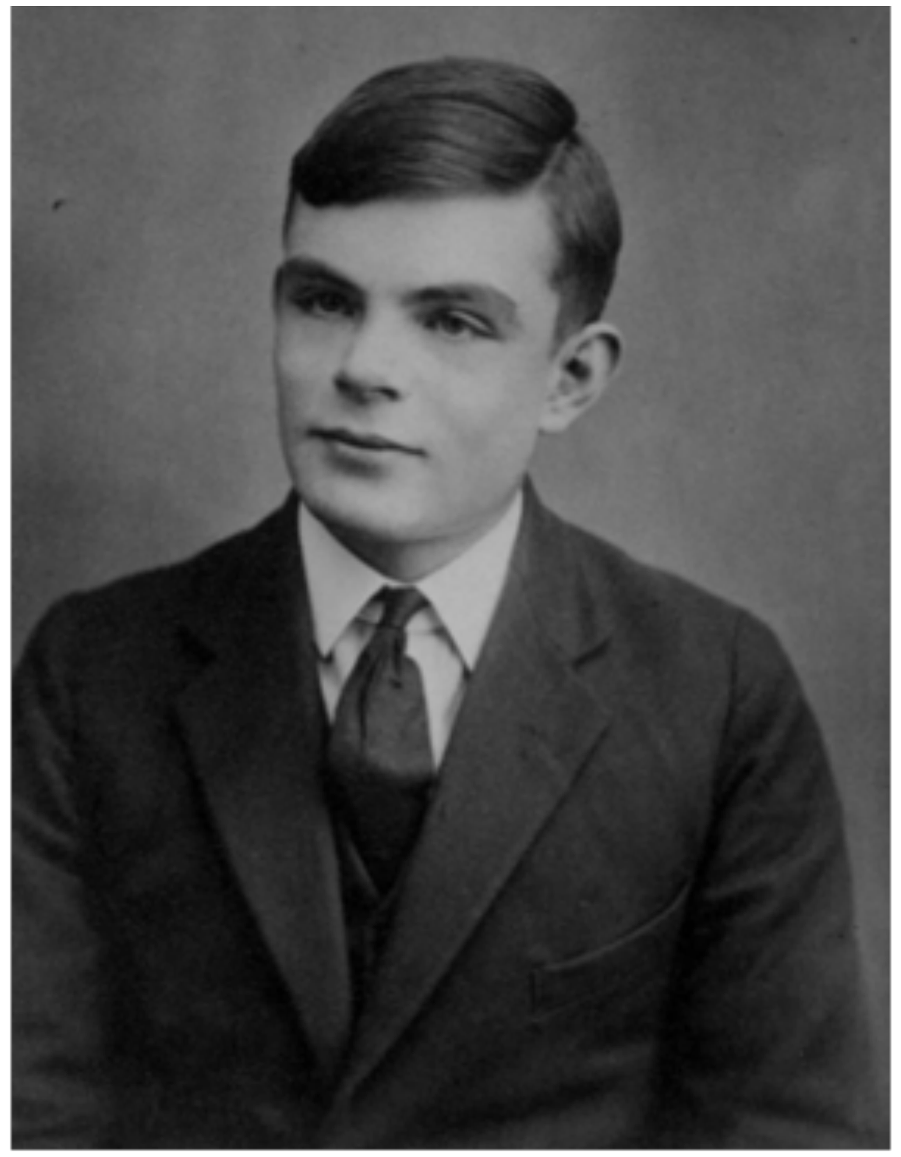
- Breaking Vigenère
  - Key length known: frequency analysis on each substring (under the same shift).
  - How to determine the key length? Read **KL.**

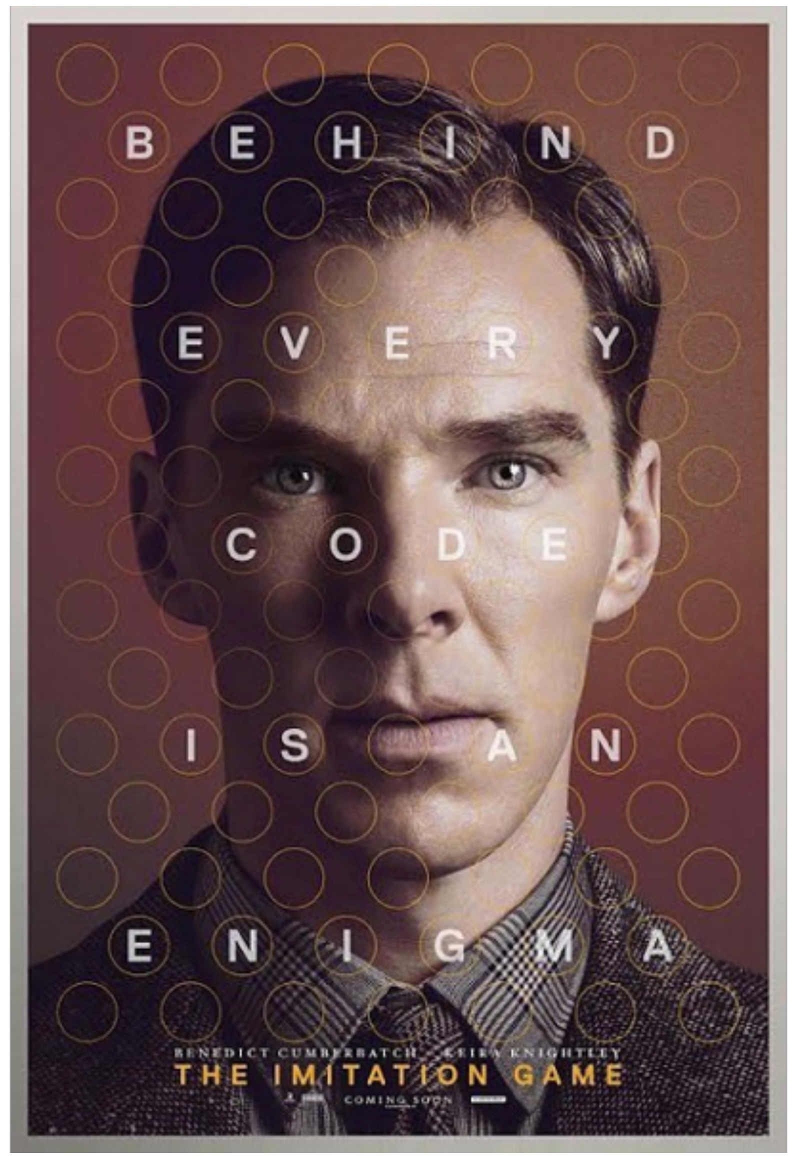
Poly-alphabetic substitution cipher

- Example: Enigma machine in WWII



Alan Turing

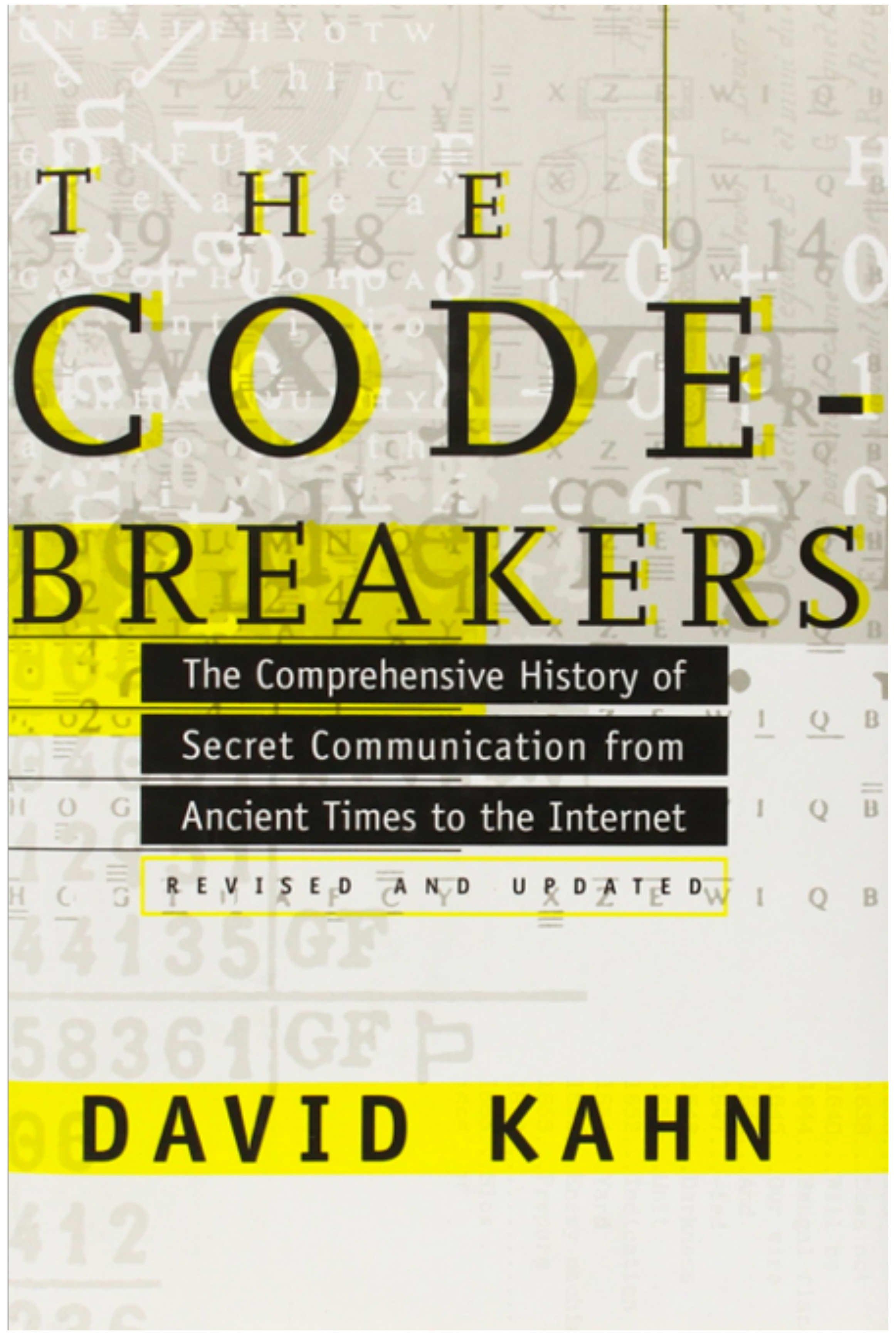


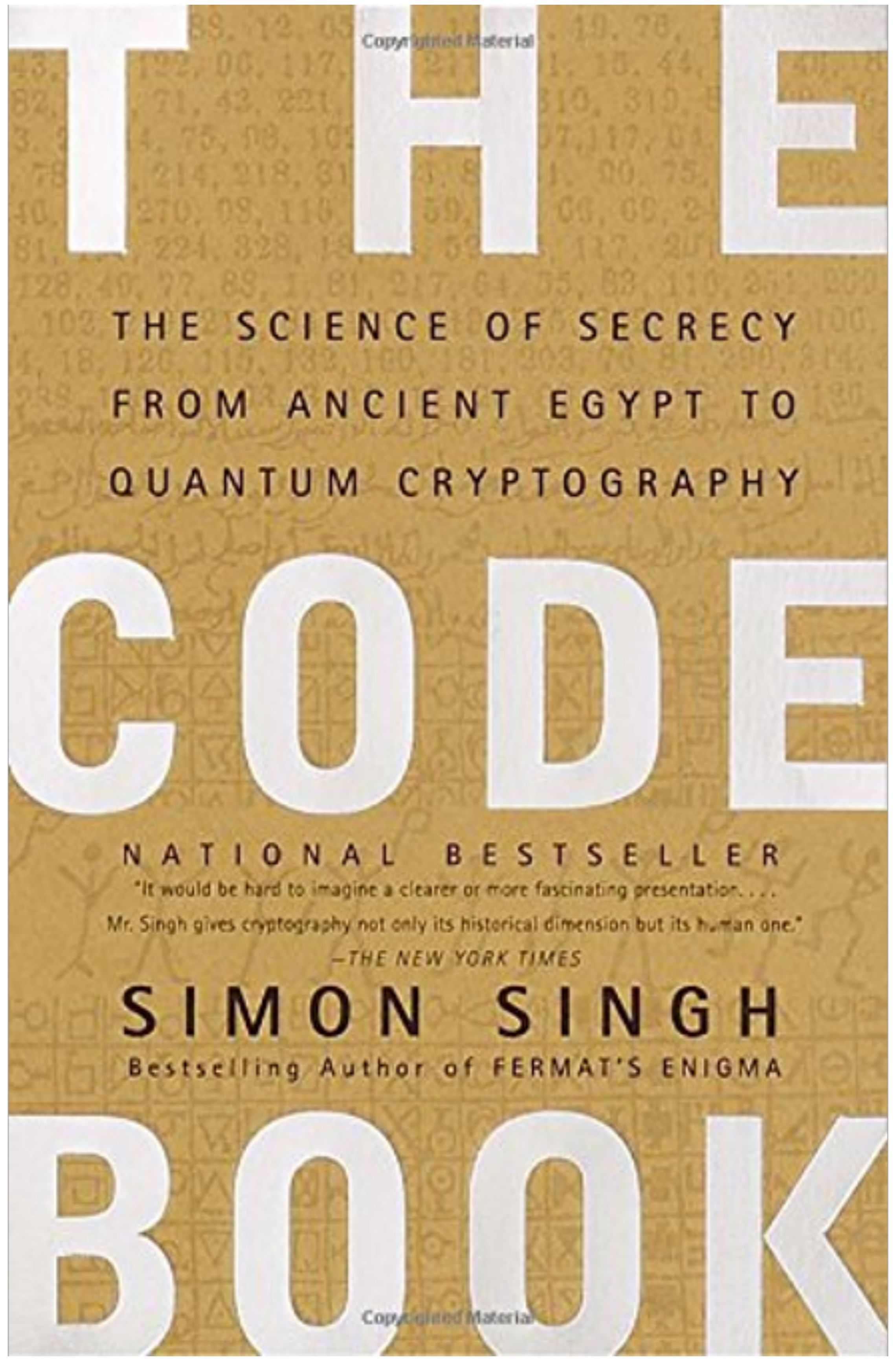


Source: imdb

- Attack: same principle as before.

Good reads on crypto history





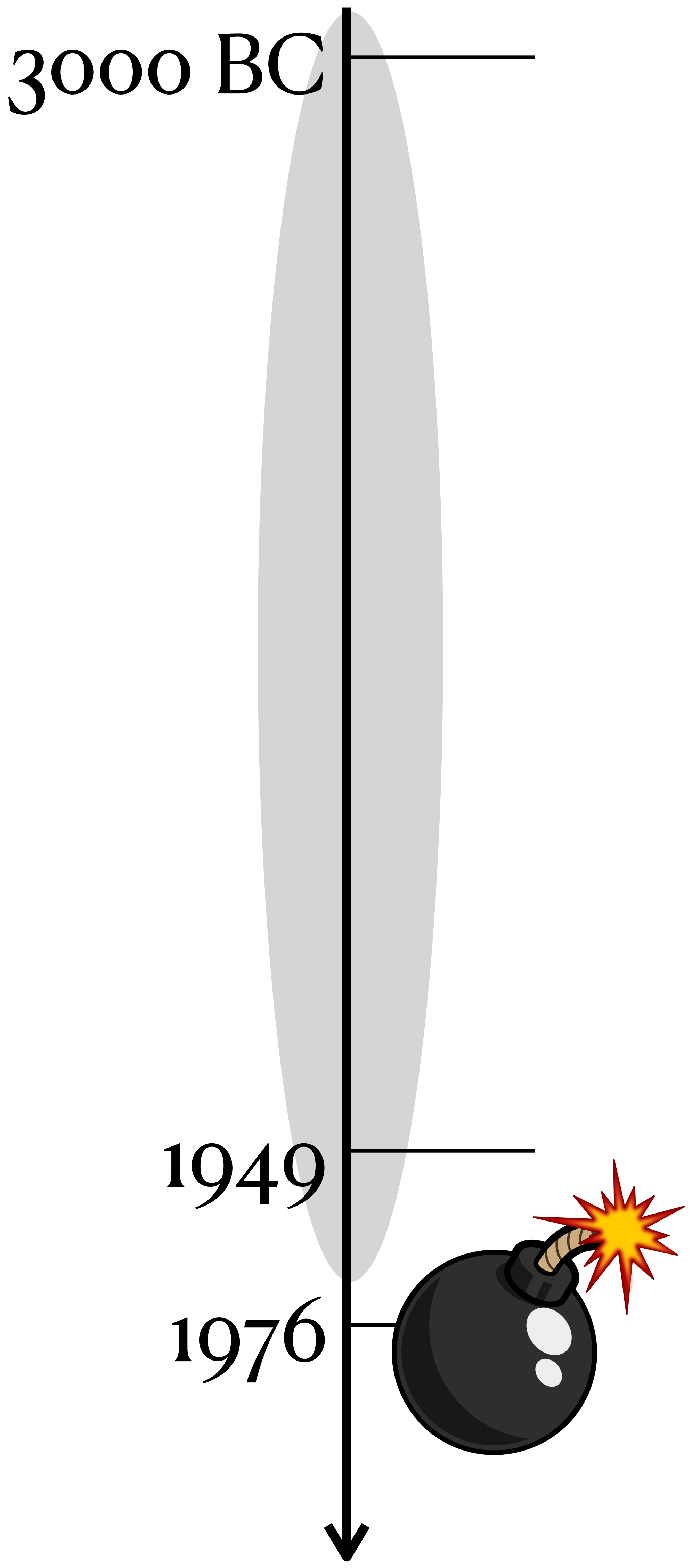
Source: amazon.com

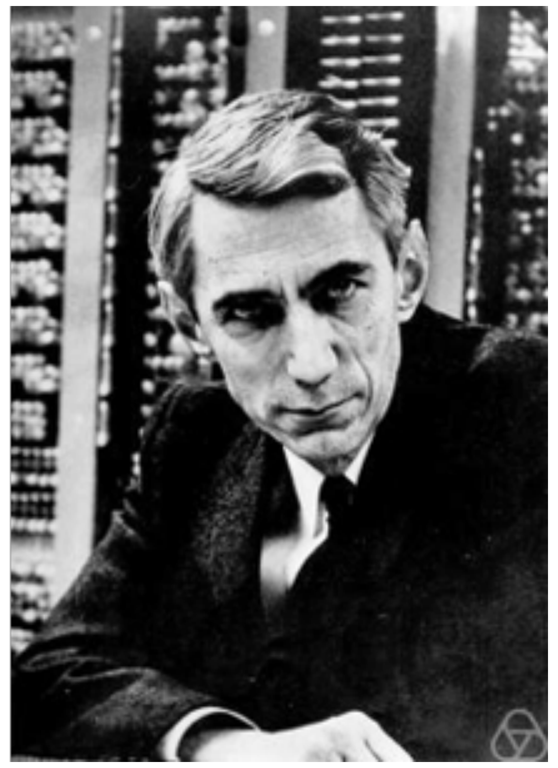
Lessons from historical ciphers

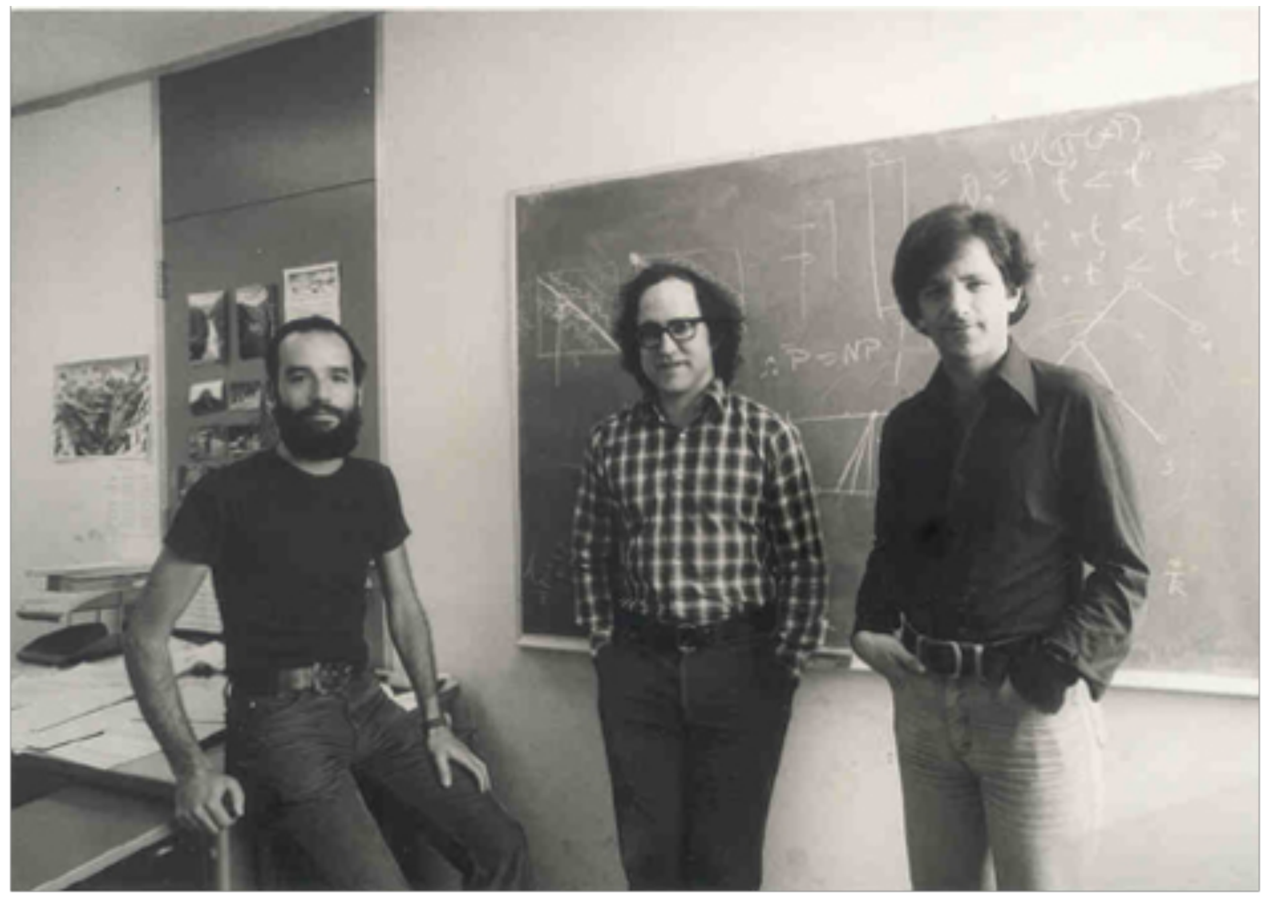
- Designing good ciphers is hard
- Looks unbreakable $\neq$ is unbreakable
- Intelligent but mostly an art

Not clear about

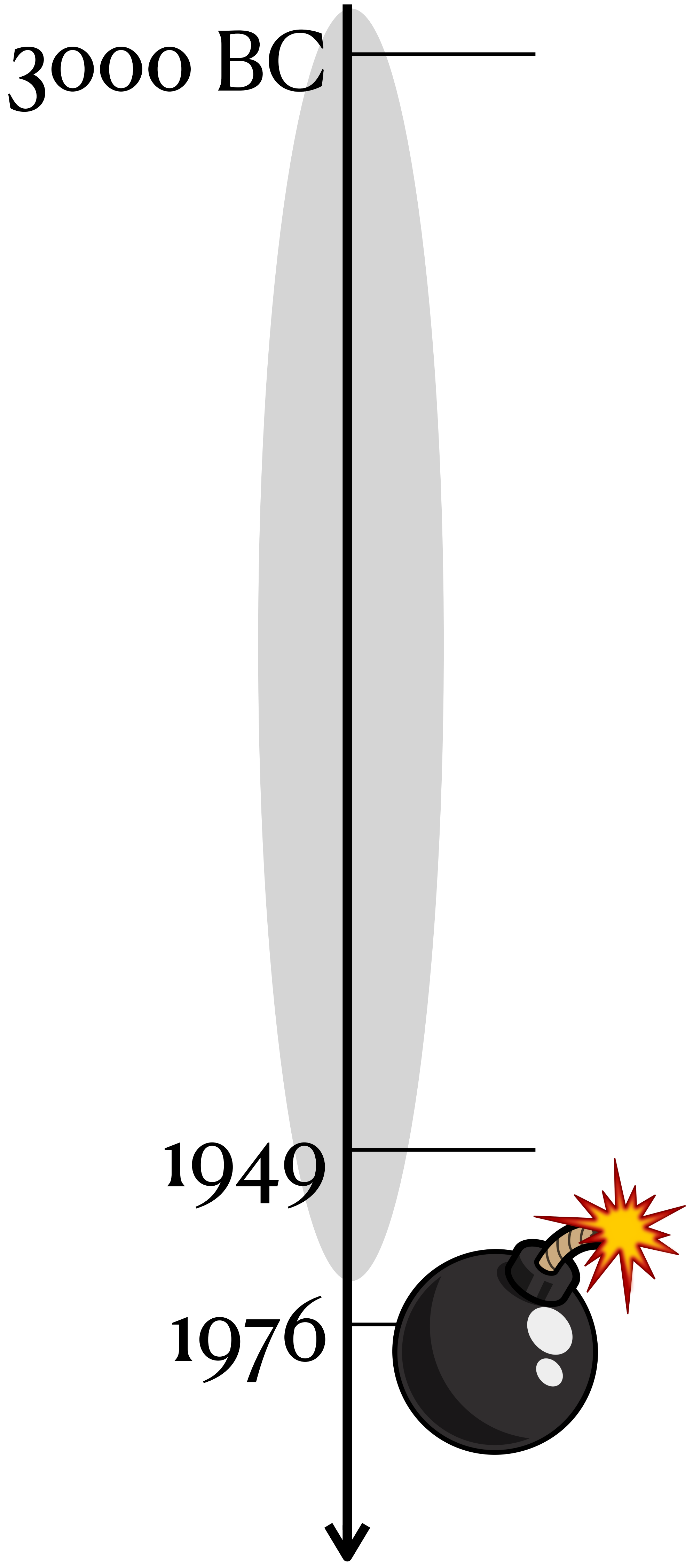
- Is a cipher secure?
- … under what circumstances?
- … and wait, what does “secure” mean precisely?







Revolution of Modern Cryptography



Concise Oxford English Dictionary

(Historical) Cryptography is the art of writing or solving codes (ciphers)

1. Much more rigorous: security via mathematics
2. Much more than "secret writing": public-key crypto, ...

Modern Cryptography involves the study of mathematical techniques for securing {digital information, systems and computations} against adversarial attacks – KL

Revolution of Modern Cryptography

What this course is about

A conceptual and theoretical tour
to modern cryptography

Yes

- Ideas
- Formal approach to security: define, construct, prove.

No

- Implementations
- Engineering skills

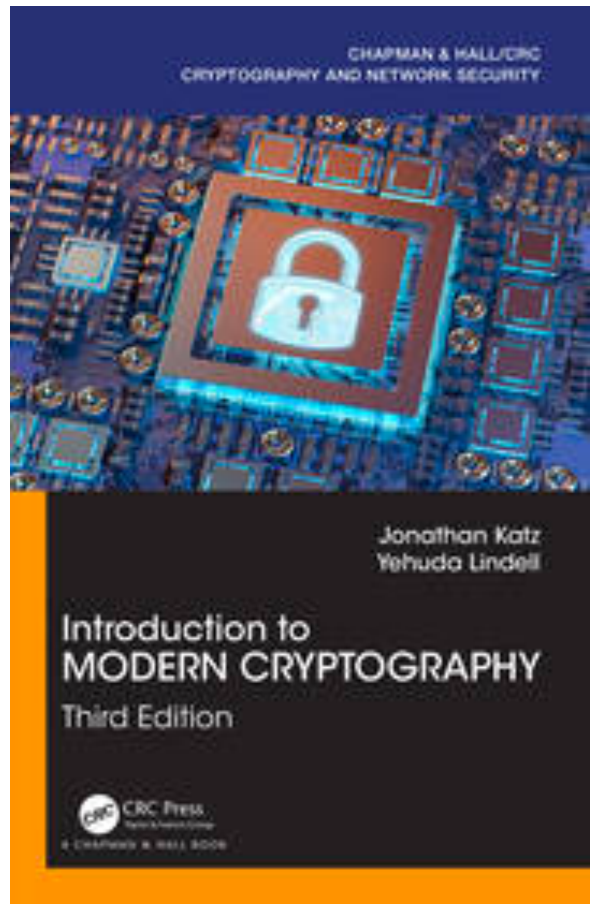
(Important but not our focus)

- **Goal:** a cryptographer's mind
  - A solid foundation for real-world security.
  - Appreciate the intellectual beauty.
  - Beneficial far beyond: differential privacy, ML, algorithms, …

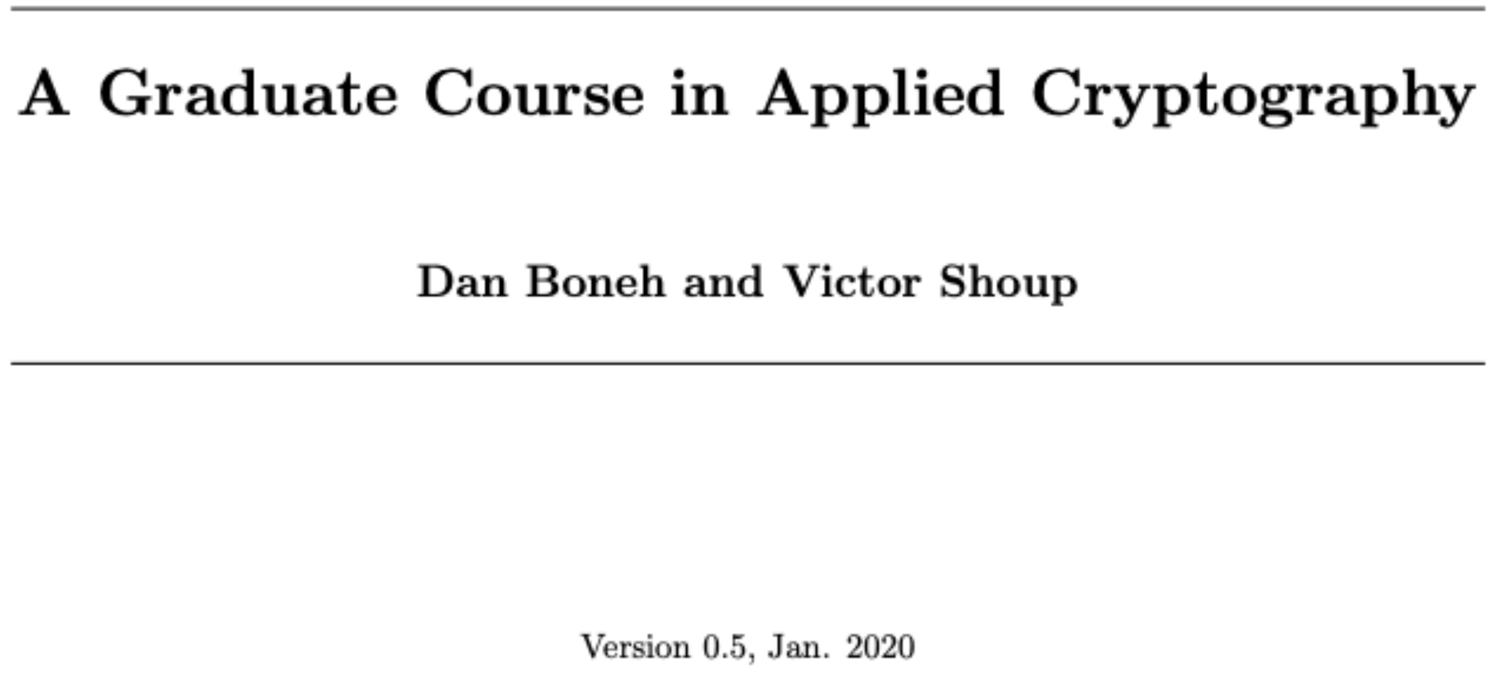
Logistics

- **Meetings:** M/W 2 - 3:50 pm @ CH 382 (Zoom participation available)
- **Instructor**: Prof. Fang Song (fang.song@pdx.edu).

- **Texts**
  - **Required**: KL
  - Supplement: BS + More on Resource page



KL


A Graduate Course in Applied Cryptography

Dan Boneh and Victor Shoup

Version 0.5, Jan. 2020

BS

Prerequisite

Comfortable with READING & WRITING mathematical Proofs

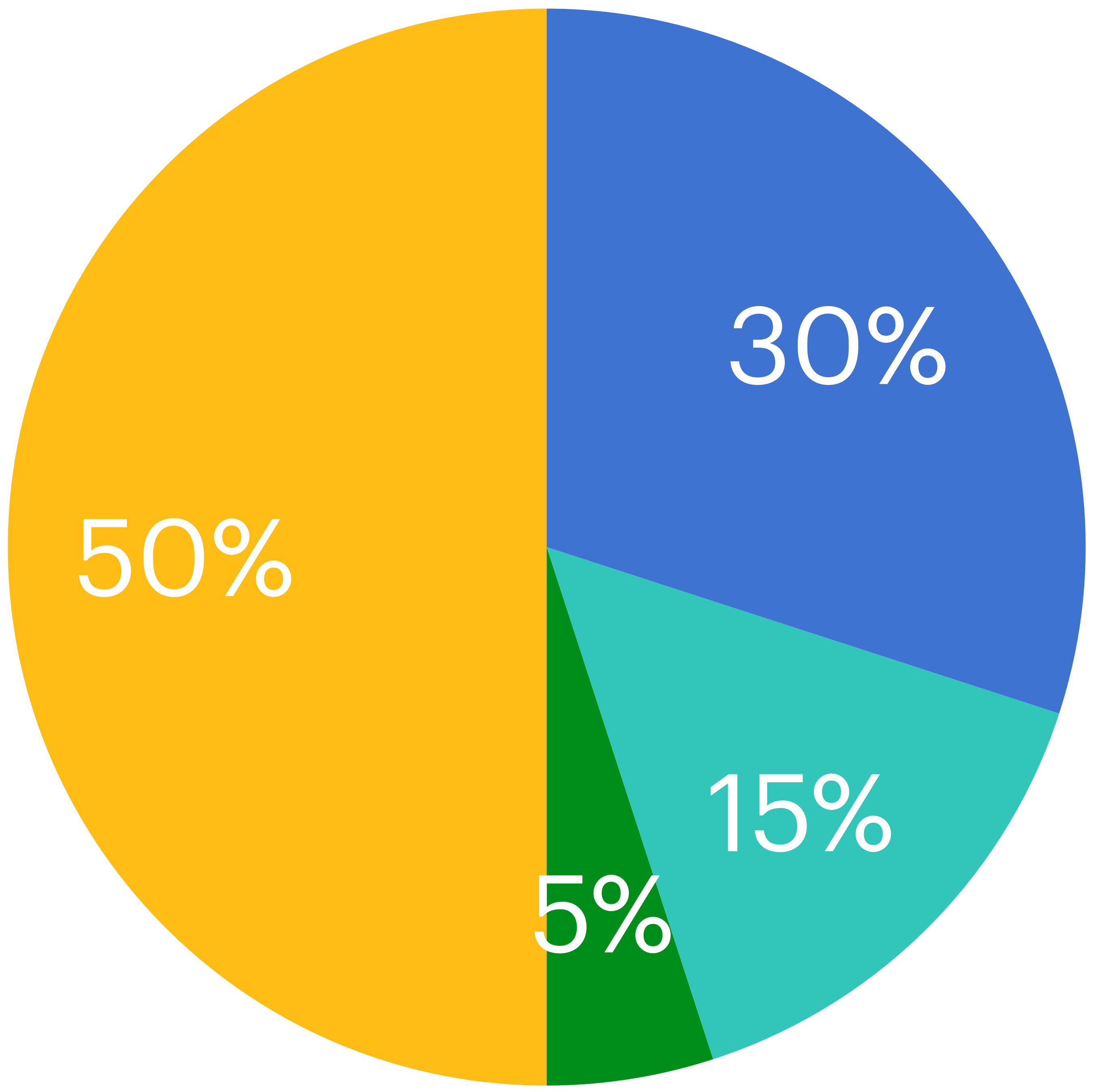
- CS 350 or equivalent
- Some math helpful
  - Combinatorics, **probability**, linear algebra, number theory …
  - "Big-Oh notation, random variable, independence, matrices, eigenvalue, congruence…"
- Programming not required

Main topics

1. **Overview. (1 week)**
    - History, principles of modern crypto, perfect secrecy
2. **Private-key *(symmetric)* crypto (4 weeks)**
    - Encryption, message authentication, hash functions
3. **Public-key *(asymmetric)* crypto (3 weeks)**
    - Encryption, digital signature
4. **Selected topics (2 weeks)**
    - Ethics, Bitcoin, quantum-safe crypto, …

Policy: grading

- Homework (biweekly): 50%.
- Project: 30%.
- Quiz (biweekly): 15%.
- Participation: 5%.



Policy: homework

- **Late submission**
  - 5 late days in total at your dispense.
- **Collaboration is encouraged.**
  - Form study groups of ≤ 3 people, brainstorm etc.
  - Write up your solutions independently.
  - Mark the names of collaborators on each problem.
  - External resources NOT permitted.
- **Your solutions must be intelligible:**
  - Be ready to explain your soln's, and convince others & **yourself.**

Policy cont'd

- **Academic Integrity**
  - PSU Student Code of Conduct

- **Academic accommodation**
  - Contact DRC (503-725-4150, drc@pdx.edu) and notify me.
- **Covid**
- **Lecture recordings**
  - Comply with FERPA and PSU's Student Code of Conduct.
  - Sharing outside this class not permitted.

How to succeed?

- Study the reading materials in advance.

- Ask a lot of questions.

- Form study groups.

- Start on assignments EARLY!
  - Make baby steps every day >> leave everything till last minute.
  - Review lecture notes & reading materials **multiple** iterations!

To-do

1. **Course webpage** https://fangsong.info/teaching/w23_4585_icrypto/
    - "Schedule" page: reading materials.
    - "Resource" page: additional materials.
    - Check regularly!

2. **Google Classroom:** lecture notes, homework, quizzes
    - Join with code: *biqddg3* (https://classroom.google.com/c/NTgwMTAwMDU4MjEw?cjc=biqddg3)
    - A calendar "`W23-CS-4585-iCrypto`" will appear in your PSU Google Calendar.

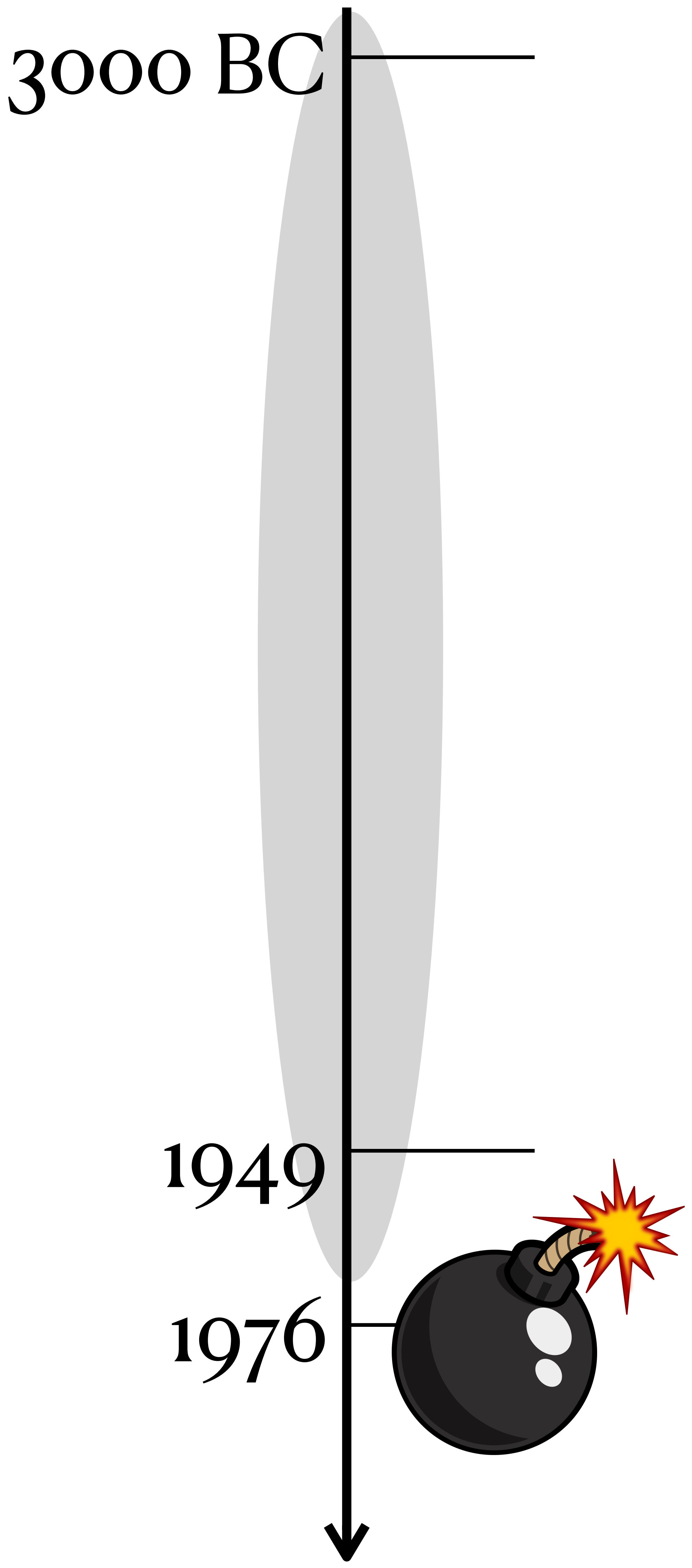
To-do, cont'd

3. **Slack(w23-4585-icrypto):** announcements, discussions, Q&A
   - Invitations sent. Important information in Pinned msg.
   - Post questions **publicly**, except for private concerns (DM me).
   - (Less efficient): email and start your subject line with "w23-4585-icrypto"
4. **Getting to know each other:**
   - Mingle in Slack. Post a short self intro. Form study groups.
5. **HW 1 will be out soon**
   - Short practice on some math/algorithms.
   - Due in one week (others will be biweekly).

Today

1. ~~History & course info.~~

2. Principles of modern Cryptography



Concise Oxford English Dictionary

(Historical) Cryptography is the art of writing or solving codes (ciphers)

1. Much more rigorous: security via mathematics
2. Much more than "secret writing": public-key crypto, ...

Modern Cryptography involves the study of mathematical techniques for securing {digital information, systems and computations} against adversarial attacks – KL

Revolution of Modern Cryptography

Principles of modern crypto

1. **Formal definitions of security**
   - What "security" do you want to achieve exactly?
   - Guide the design and assess of a construction.
   - Know better what you need.

Principles of modern crypto, cont'd

2. **Rigorous proofs of security**
   - The only known method to reason against (infinitely) many possible attacking strategies.
   - Never rely on your pure impression.

Principles of modern crypto, cont'd

3. Precise assumptions

- Unconditional security is often **impossible** to attain.
- Be precise, for validating and comparing schemes.

Assume "my construction satisfies the definition.

Vs.

Assume "factoring 1000-bit integer cannot be done in less than 1000 steps".

- Well-studied >> ad hoc: test-of-time.
- Neat >> vague: easy to assess/falsify.

★ Modularity: replace a building block when needed.

Recap: principles of modern crypto

1. Formal definitions of security
2. Rigorous proofs of security
3. Precise assumptions

In contrast, historical crypto is not careful about

- is a cipher secure?
- ... under what circumstances?
- ... and wait, what does "secure" mean precisely?

Provable security & real-world

A scheme has been proven secure $\not\Rightarrow$ security in the real world

◉ **Are the definitions / assumptions right?**

- Not match what is needed.
- Not capture attackers' true abilities.

★**You (the designer/defender) have more in charge, instead of attackers.**

- Seek improvements proactively: refine defs, test assumptions, …

Scratch

Supplement

1. ~~History & course info.~~

2. ~~Principles of modern Cryptography~~

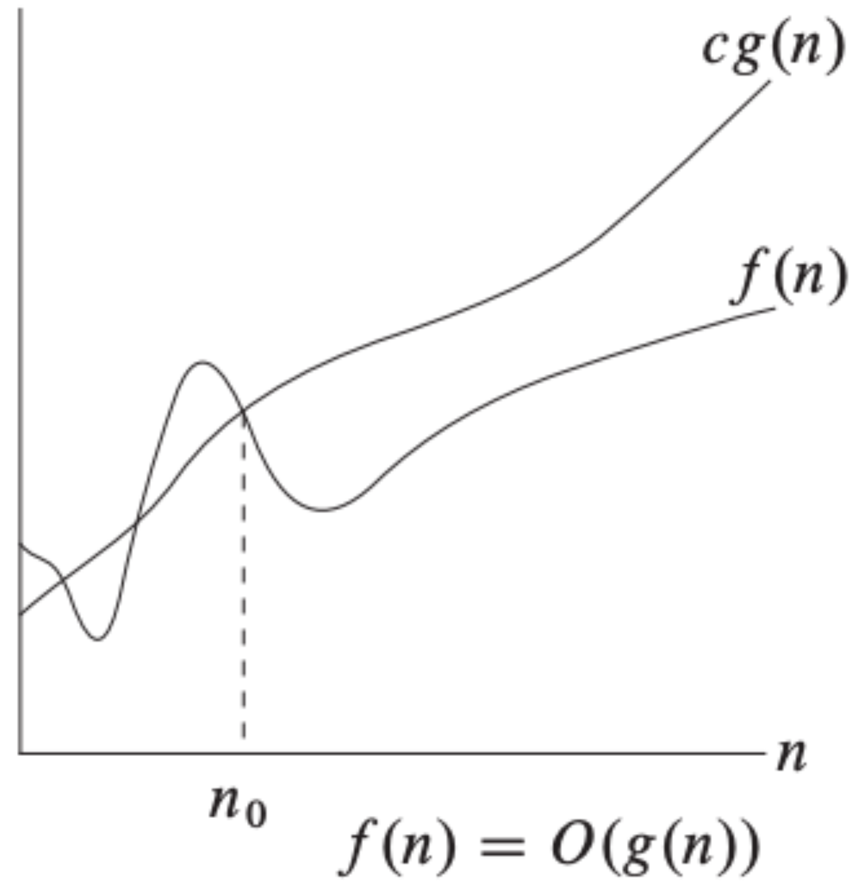
3. Review: mathematical background (on separate note)
    - Sets
    - Asymptotic notations
    - Probability 101

Asymptotic notations

- $O(\cdot), \Omega(\cdot), \Theta(\cdot), o(\cdot), \omega(\cdot)$
  - Measure algorithm behaviors (by functions on integers) as problem size grows.

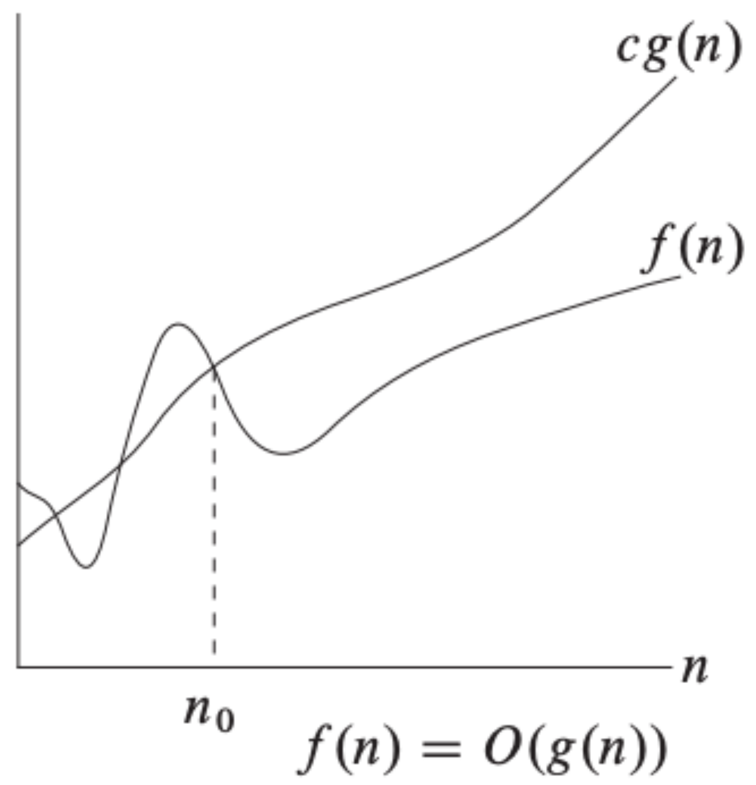
- **Defining $O(\cdot)$: asymptotic upper bound**

We write $f(n) = O(g(n))$ if there exist constants $c > 0, n_0 > 0$, such that $0 \leq f(n) \leq cg(n)$ for all $n \geq n_0$.



- **$O(g(n))$ as a set**

$$O(g(n)) := \{f(n) : \exists c > 0, n_0 > 0, \text{ such that } 0 \geq f(n) \leq c \cdot g(n) \text{ for all } n \geq n_0\}$$



Examples

$f(n) = O(g(n))$ if there exist constants $c > 0, n_0 > 0$, such that $0 \le f(n) \le cg(n)$ for all $n \ge n_0$.

- $2n^2 = O(n^3)$
  - $c = 1, n_0 = 2$.
  - I.e., $2n^2 \in O(n^3)$
- $f(n) = n^3 + O(n^2)$
  - Meaning $f(n) = n^3 + h(n)$ for some $h(n) \in O(n^2)$

Exercise: sort by asymptotic order of growth

1. $n \log n$
2. $\sqrt{n}$
3. $\log n$
4. n^2
5. 2^n
6. n
7. $n!$
8. $n^{1,000,000}$
9. $n^{1/\log n}$
10. $\log(n!)$

List them in ascending order: if f appears before g, then $f = O(g)$

9, 3, 2, 6, 1=10, 4, 8, 5, 7

Summary

Notation	… means …	Think…	E.g.	Lim $f(n)/g(n)$
$f(n)=O(n)$	$\exists\ c>0, n_0>0, \forall\ n > n_0$: $0 \leq f(n) < cg(n)$	Upper bound	$100n^2 = O(n^3)$	If it exists, it is $< \infty$
$f(n)=\Omega(g(n))$	$\exists\ c>0, n_0>0, \forall\ n > n_0$: $0 \leq cg(n) < f(n)$	Lower bound	$n^{100} = \Omega(2^n)$	If it exists, it is > 0
$f(n)=\Theta(g(n))$	both of the above: $f=\Omega(g)$ **and** $f = O(g)$	Tight bound	$\log(n!) = \Theta(n \log n)$	If it exists, it is > 0 and $< \infty$
$f(n)=o(g(n))$	$\forall\ c>0, n_0>0, \forall\ n > n_0$: $0 \leq f(n) < cg(n)$	Strict upper bound	$n^2 = o(2^n)$	Limit exists, $=0$
$f(n)=\omega(g(n))$	$\forall\ c>0, n_0>0, \forall\ n > n_0$: $0 \leq cg(n) < f(n)$	Strict lower bound	$n^2 = \omega(\log n)$	Limit exists, $=\infty$

Review: Chapter 3 Introduction to Algorithms, By Cormen, Leiserson, Rivest and Stein.