



# Elementi di Crittografia

A.A. 2023-2024



Prof. Paolo D'Arco

# Introduzione

---

## Crittografia / Crittologia

Due parole greche:

**Kryptos** che significa “nascosto”

**Graphìa** che significa “scrittura” / **Logos** che significa “discorso”



# Introduzione

---

Dal vocabolario Treccani:

“Crittografia: l’insieme delle teorie e delle tecniche (manuali, meccaniche o elettroniche) che permettono di **cifrare** un testo in chiaro ...”

Dal Concise Oxford English Dictionary:

“Cryptography: the **art** of **writing or solving codes**”

Definizioni storicamente corrette ma ... **non catturano** l’ampiezza attuale!

---



# Attività umane

---

- Conversare privatamente
- Avere un diario segreto
- Prelevare soldi dal conto
- Votare alle elezioni
- Partecipare ad un'asta
- Giocare a poker
- Effettuare compravendite

...

---



# Attività umane

---

- Conversare privatamente
- Avere un diario segreto
- Prelevare soldi dal conto
- Votare alle elezioni
- Partecipare ad un'asta
- Giocare a poker
- Effettuare compravendite

...

## Dispositivi fisici e parti fidate

- Ambienti riservati/protetti
- Lucchetti e casseforti
- Bancomat, carta, pin
- Cabine e schede elettorali
- Buste oscuranti
- Carte da gioco
- Notai e ufficiali riconosciuti



---

# Fiducia

---



# Mondo digitale: “chiedere la luna ...”

---

*Vogliamo riprodurre le stesse attività, ma gli unici oggetti di cui disponiamo sono sequenze di bit, che possono essere elaborate e trasmesse su reti di computer!*

**La crittografia moderna rende possibile tutto ciò senza ricorso ad alcun dispositivo fisico e riducendo al minimo l'uso di parti fidate**

---



# Crittografia moderna

---

*“Lo studio delle tecniche matematiche utili a proteggere l’informazione digitale, i sistemi di elaborazione e le computazioni distribuite **da attacchi avversari**”*

**Strumento per corroborare la fiducia nei processi che abbiamo riprodotto o creato ex-novo nel mondo digitale**

---



# Scopo del corso

---

Presentare i principi e le tecniche alla base della Crittografia Moderna.

Al termine dovrete essere in grado di:

- **apprezzare** le **garanzie** che le primitive crittografiche offrono in termini di sicurezza
- **conoscere** ed avere una discreta familiarità con le **costruzioni standard**
- **effettuare valutazioni di base** di primitive e protocolli crittografici

Approccio: studio rigoroso e strutturato

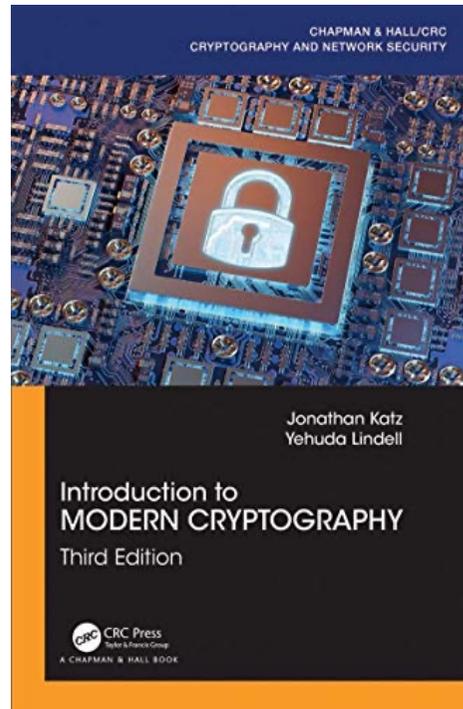
---



# Libri di testo

---

J. Katz and J. Lindell, **Introduction to Modern Cryptography**, 3-rd Edition. CRC Press, 2021



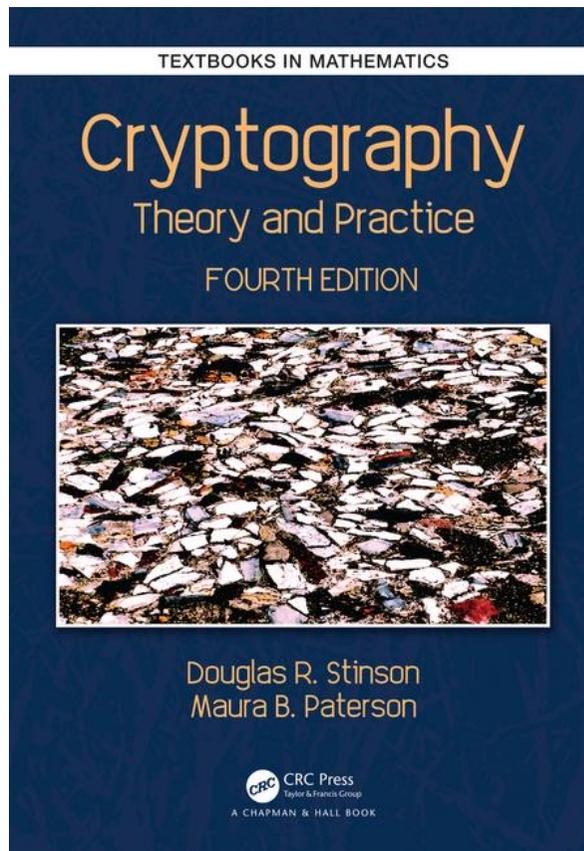
Riferimento principale ... più letture di approfondimento consigliate

---



# Libri di testo

---



D.R. Stinson and M.B. Paterson,  
**Cryptography: Theory and Practice**,  
CRC Press, 2018

Approccio più tradizionale, con capitoli su  
tecniche avanzate (e.g. post-quantum crypto).

Taglio matematico.

---



# Confidenzialità ...

---

Dal vocabolario Treccani:

“Crittografia: l’insieme delle teorie e delle tecniche (manuali, meccaniche o elettroniche) che permettono di **cifrare** un testo in chiaro ...”

Dal Concise Oxford English Dictionary:

“Cryptography: the **art** of **writing or solving codes**”

Definizioni storicamente corrette ma ... **non catturano** l’ampiezza attuale!

---



# Introduzione: oltre la “confidenzialità”

---

- Meccanismi per assicurare l'**integrità** dei dati
  - esser certi che i dati trasmessi o memorizzati non siano stati modificati
- Meccanismi per assicurare l'**autenticità** dei dati e delle parti che entrano in gioco in un protocollo
  - esser certi che **i dati** che abbiamo necessità di elaborare siano prodotti dall'ente/persona che supponiamo sia l'origine dei dati stessi
  - esser certi che **la parte** con cui stiamo comunicando sia quella giusta e non un impostore

e, ancora ...

---



# Introduzione: oltre la “confidenzialità”

---

- Protocolli per lo scambio sicuro di **chiavi crittografiche** tra persone che non si sono mai incontrate in precedenza
- Protocolli per funzionalità generali e funzionalità specifiche:
  - Schemi per la **condivisione di segreti**
  - Sistemi di **prova a conoscenza zero**
  - **Calcolo sicuro di funzioni multi-party** (n variabili – n parti)
  - Elezioni elettroniche
  - Giochi on-line
  - Denaro digitale
  - Database distribuiti di transazioni autenticate ... e molto altro!



# Crittografia moderna

---

- Tra gli anni 80' e 90': transizione da “arte” a “scienza”



# Natura della transizione

---

Insieme di  
strumenti euristici  
che garantiscono  
comunicazioni  
segrete a militari e  
diplomatici



Scienza che aiuta a  
proteggere i  
sistemi digitali usati  
da persone  
ordinarie sparse su  
tutto il globo  
terrestre



# Crittografia Classica

---

- Permette di introdurre facilmente alcuni concetti di base
- Ci serve per motivare l'approccio più rigoroso che applicheremo nel seguito

Obiettivo classico: progetto ed uso di codici (anche detti cifrari)

“due parti comunicano segretamente in presenza di un ascoltatore che può monitorare tutte le comunicazioni tra di loro”

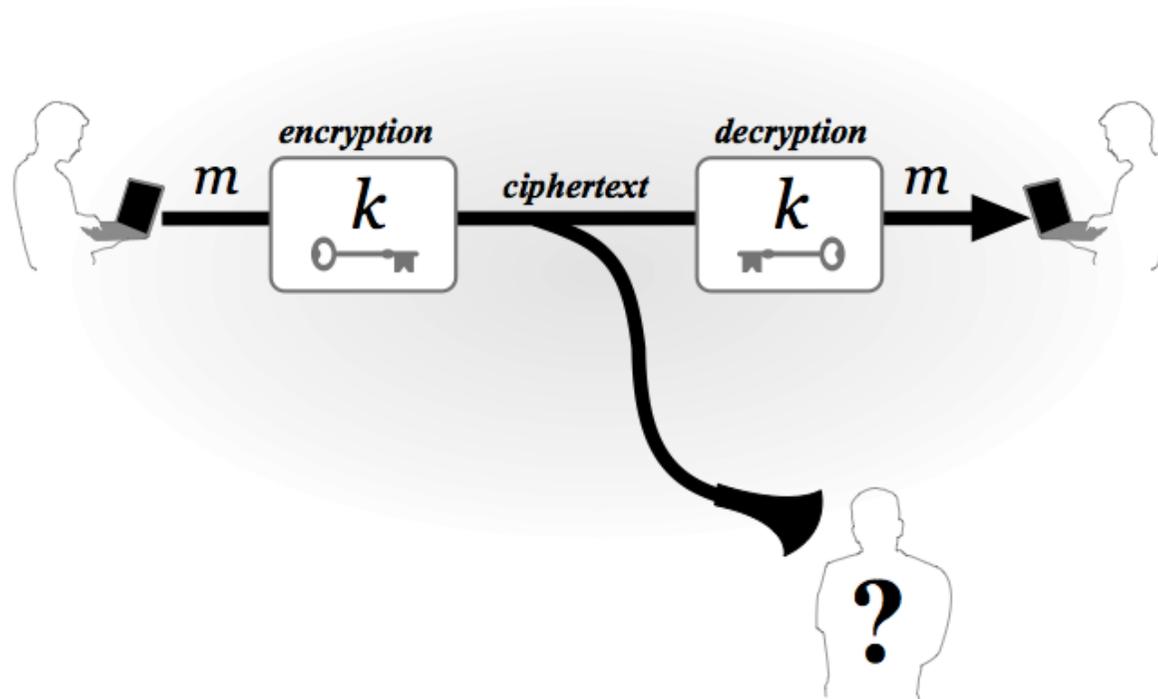
Codici, in termini moderni: schemi di cifratura

---



# Schemi di cifratura

---



Basato sulla segretezza della chiave condivisa dalle due parti comunicanti

---



# Schemi di cifratura

---

A chiave simmetrica (symmetric key setting) = **stessa chiave** per cifrare e decifrare

A chiave asimmetrica (asymmetric key setting / public key setting) = **chiavi diverse**

Due applicazioni canoniche per lo scenario a chiave simmetrica:

- le parti sono separate “**nello spazio**”
- una parte comunica con se stessa “**nel tempo**”

Sintassi che useremo per riferirci agli oggetti di uno schema:

**M** : spazio dei messaggi leciti  
**C** : spazio dei cifrati possibili  
**Gen** : algoritmo per generare le chiavi  
**Enc** : algoritmo di cifratura  
**Dec** : algoritmo di decifratura



# Chiavi e principio di Kerckhoffs

---

Osservazione: un avversario, disponendo della chiave  $K$  e **Dec**, decifra!

Pertanto, la chiave deve essere segreta. E circa l'algoritmo **Dec** di decifratura?

August Kerckhoffs, fine 19esimo secolo

*“Il metodo di cifratura non è richiesto che sia segreto e dovrebbe essere in grado di cadere nelle mani del nemico senza creare problemi.”*



La sicurezza non dovrebbe basarsi sulla segretezza degli algoritmi di cifratura, ma soltanto sulla segretezza della chiave.

---



# Perché?

---

Argomenti a favore del principio di Kerckhoffs:

- è più facile per le parti proteggere solo  $K$
- è più facile cambiare una chiave che un intero schema
- è più facile lo sviluppo su larga scala se gli utenti possono usare gli stessi algoritmi
- rende possibile la standardizzazione, che è utile per garantire compatibilità ed il pubblico scrutinio

Oggi il principio di Kerckhoffs è universalmente accettato e la pratica della cosiddetta “security by obscurity” – che poggia sulla segretezza anche dei metodi - sempre più desueta.

---



# Nota: Kerckhoff e la Cybersecurity

---

Terminologia introdotta dalla letteratura (Science-fiction)

*Cyberspace* – termine introdotto nel romanzo *Burning Chrome* di William Gibson del 1982 per denotare reti interconnesse di computer, dispositivi e persone

*Cyberpunk* – titolo del romanzo di Bruce Bethke del 1983, usato per denotare individui “socialmente inetti” ma con capacità tecnologiche notevoli

*Cyberattack, cybercrime, cyberwar, cyberterrorism ...et cetera*

Il principio di Kerckhoff si applica anche nell’area più ampia della cosiddetta cybersecurity.

---



# Schemi “storici”

---

Consideriamo ora alcuni cifrari al fine di:

- mostrare le debolezze di un approccio “ad hoc”, motivando l’importanza dell’approccio moderno strutturato
- rendere chiaro che approcci semplicistici difficilmente hanno successo



# Cifrario di Cesare

---

“De Vita Caesarum”, 110 a. C.

Cifratura: ogni lettera del messaggio in chiaro viene sostituita dalla lettera che si trova **3** posti in avanti nell'alfabeto

$A \rightarrow D, B \rightarrow E, C \rightarrow F, \dots, Z \rightarrow C$

Decifratura: operazione inversa (lettera che precede di **3** posti)

Osservazioni: lo schema di cifratura è fisso. Non c'è chiave segreta!

- una variante – ROT 13 – è usata ancora oggi in alcuni forum on-line per disturbare chi vuole capire al volo cosa le parti si dicono
- invece di **3**, i posti in avanti sono **13**

Il cifrario di Cesare è un caso particolare dello Shift Cipher

---



# Shift cipher

---

Alfabeto Inglese mappato su  $\{0, 1, 2, \dots, 25\}$

↑ ↑ ↑ ↑  
a, b, c, ..., z

Messaggio  $m = m_1m_2\dots m_l$ , con  $m_i$  in  $\{0, 1, 2, \dots, 25\}$

**Enc<sub>K</sub>**( $m_1m_2\dots m_l$ ) =  $c_1c_2\dots c_l$ , con  $c_i = [(m_i + k) \bmod 26]$  per  $i=1, \dots, l$

dove  $[a \bmod N]$  denota il resto della divisione per  $N$  e risulta  $0 \leq [a \bmod N] < N$

L'associazione di  $[a \bmod N]$  ad  $a$  prende il nome di “riduzione modulo  $N$ ”.

**Dec<sub>K</sub>**( $c_1c_2\dots c_l$ ) =  $m_1m_2\dots m_l$ , con  $m_i = [(c_i - k) \bmod 26]$  per  $i=1, \dots, l$

È sicuro lo Shift cipher?

---



# Ricerca esaustiva

---

Dato un cifrato, ci sono soltanto 26 possibili chiavi. Basta provarle tutte!

Un attacco che richiede di provare ogni possibile chiave viene detto un attacco per “ricerca esaustiva” o anche di “forza bruta”.

Condizione necessaria (ma non sufficiente) affinché un cifrario sia sicuro è che lo spazio delle chiavi sia sufficientemente grande (sufficient key-space principle)

*“Qualsiasi schema di cifratura sicuro deve avere uno spazio delle chiavi sufficientemente grande da rendere un attacco per ricerca esaustiva impraticabile.”*

Tipicamente deve contenere almeno  $2^{80}$  elementi al giorno d’oggi.

---



# Cifrari per sostituzione monoalfabetica

---

“Shift cipher”  $\longrightarrow$  l’associazione carattere in chiaro/carattere cifrato è uno **spostamento fissato**

“Mono-alfabetico”  $\longrightarrow$  associazione **arbitraria**

Spazio delle chiavi = { tutte le possibili permutazioni dell’alfabeto }

alfabeto in chiaro

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

alfabeto cifrante

---



# Cifrari per sostituzione monoalfabetica

---

Messaggio in chiaro = tell him about me      tellhimaboutme  
Messaggio cifrato = GDOO KVC XEFLG CD      GDOOKVCXEFLGCD

Sostituzione monoalfabetica  $\longrightarrow$  La chiave specifica una sostituzione FISSA per ogni carattere del messaggio in chiaro

Lo stesso carattere viene cifrato sempre allo stesso modo

| Spazio delle chiavi | =  $26! = 26 \cdot 25 \cdot 24 \cdot \dots \cdot 2 \cdot 1 \approx 2^{88}$

Un attacco di forza bruta è impraticabile! Sfortunatamente, non è sufficiente ...

---



# Analisi delle frequenze

---

Il cifrario è facile da “rompere”.

Supponiamo che il messaggio in chiaro sia testo inglese corretto. Il cifrario può essere attaccato usando **caratteristiche statistiche** della lingua inglese.

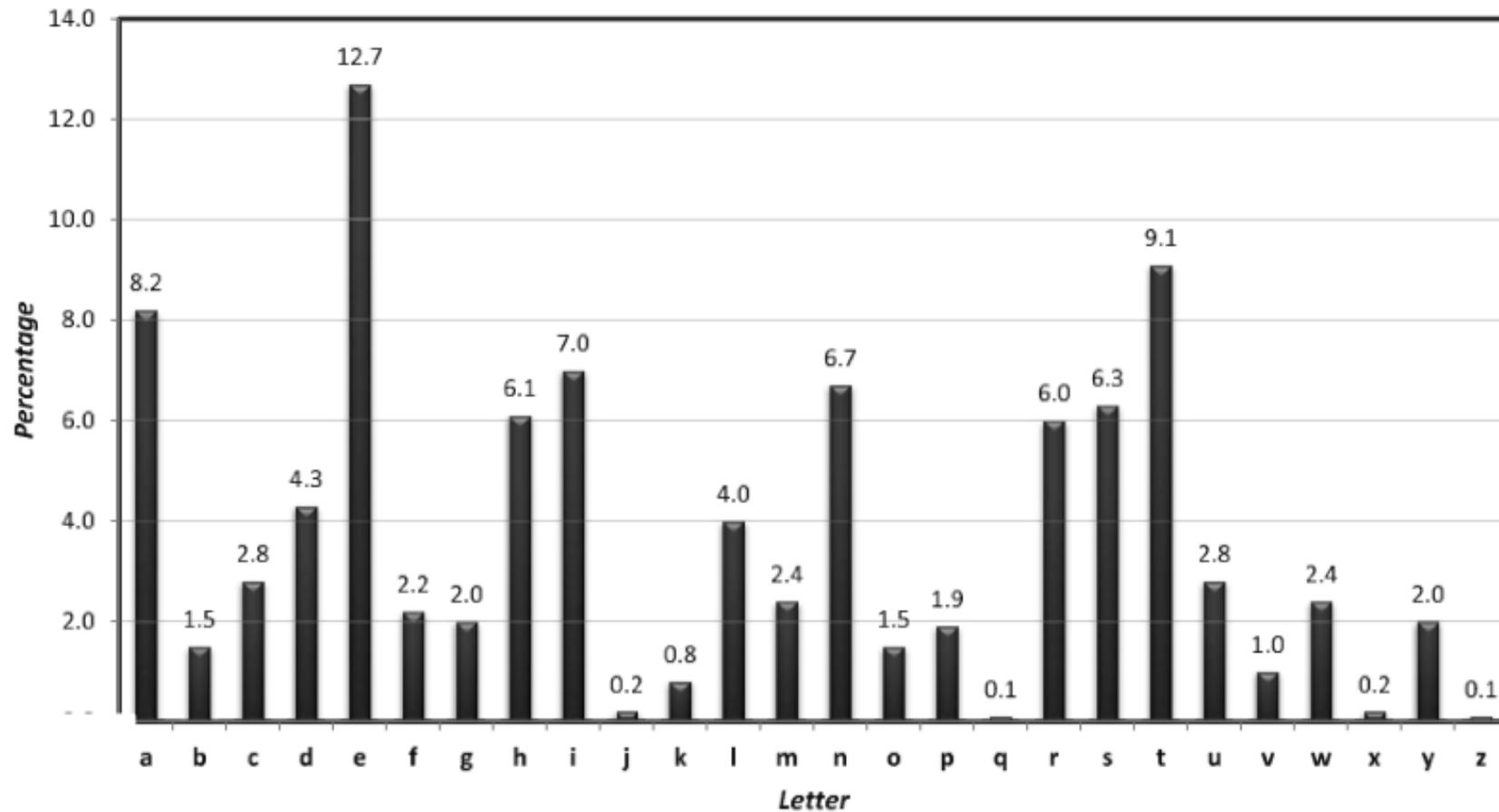
L'attacco si basa su due elementi:

1. Per ogni chiave, l'associazione di ciascuna lettera **è fissa**, per cui se “D” viene associata ad “e”, allora **ogni** occorrenza di “e” nel messaggio in chiaro diventa “D” nel cifrato
2. La **distribuzione delle frequenze** delle lettere dell'alfabeto Inglese è **nota**.



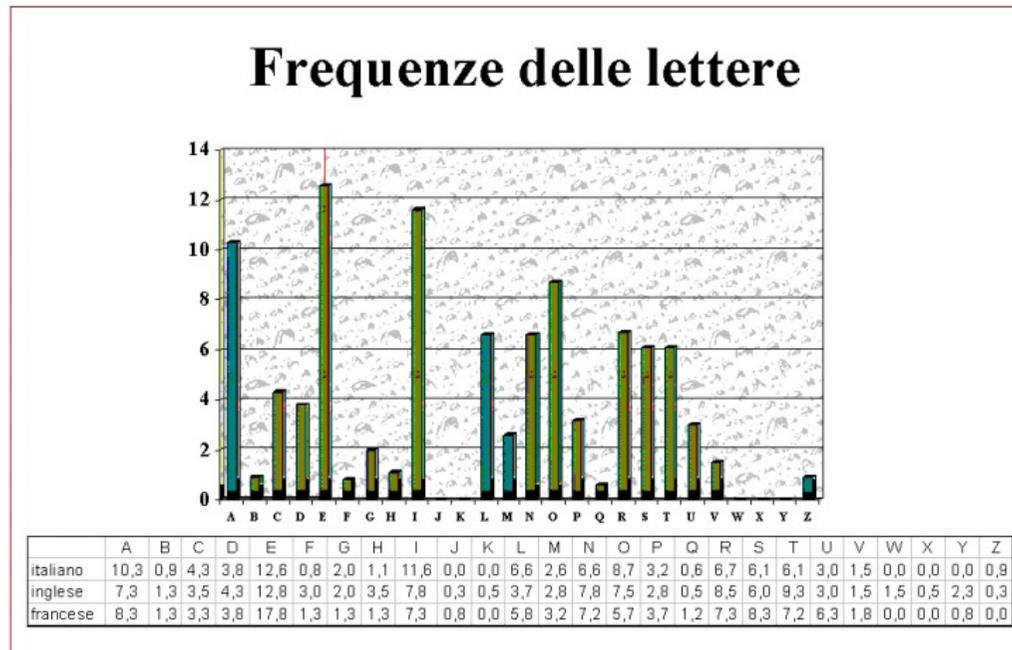
# Frequenze delle lettere dell'Inglese

---



# Frequenze in lingue diverse

---



# Attacco tramite analisi delle frequenze

---

L'attacco opera **tabulando le frequenze dei caratteri presenti nel cifrato.**

Queste frequenze sono poi **comparate** alle frequenze dell'Inglese

In questo modo:

- parte dell'associazione può essere inferita
- ipotesi corrette ed altre conoscenze della lingua Inglese possono aiutare a ricostruirla (e.g., dipendenze tipo “the”, “qu”, etc)

Conclusione: spazio delle chiavi grande ma ... cifrario debole!

---



# Provate!

---

Per rendervi conto dell'applicabilità dell'attacco e del suo funzionamento, provate a decifrare l'esempio che segue (usando la tabella delle frequenze dell'Inglese)

JGRMQOYGHMVBjWRWQFPWHGFFDQGFPFZRKBEEBJIZQQOCIBZKLFAFGQVFZFWWE  
OGWOPFGFHWOLPHLRLOLFDMMFGQWBLWBWQOLKFWBYLBLYLFSFLJGRMQBOLWJVFP  
FWQVHQWFFPQOQVFPQOCFPOGFWFJIGFQVHLHLROQVFGWJVFPFOLFHGQVQVFILE  
OGQILHQFQGIQVVOSFAFGBWQVHQWijVWJVFPFWHGFIWIHZZRQGBABHZQOCGFHX



# Shift cipher: un attacco più efficiente

---

Usiamo la distribuzione delle frequenze per calcolare la chiave segreta.

Sia  $0 \leq p_i \leq 1$  la frequenza della  $i$ -esima lettera dell'alfabeto inglese.

Risulta:

$$\sum_{i=0}^{25} p_i^2 \approx 0.065$$

Si consideri il cifrato dato.

Sia  $0 \leq q_i \leq 1$  la frequenza della  $i$ -esima lettera dell'alfabeto inglese nel cifrato, cioè:

$$0 \leq q_i = (\text{num. di occorrenze } i\text{-esima lettera nel cifrato} / \text{lungh. cifrato}) \leq 1$$

Se la chiave segreta scelta è  $k$ , allora si ha che:

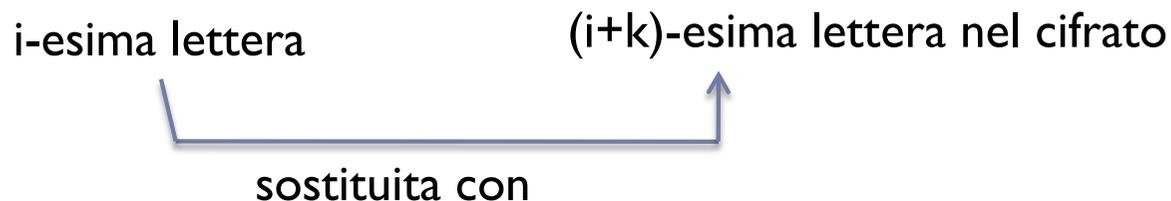
---



# Shift cipher: un attacco più efficiente

---

$p_i$   $\longrightarrow$   $q_{i+k}$  per tutti gli indici  $i=0, \dots, 25$



Per ogni valore di  $j = 0, 1, \dots, 25$ , calcoliamo  $I_j = \sum_{i=0}^{def\ 25} p_i \cdot q_{i+j}$

Ci aspettiamo:  $I_k \approx 0.065$

$I_j, j \neq k$  sostanzialmente diverso da 0.065

---



# Shift cipher: un attacco più efficiente

---

L'attacco restituisce **come ipotesi per**  $k$  il valore di  $j$  per cui  $I_j$  è più vicino a 0.065

Osservazioni sull'attacco:

- è efficiente
- è facile da automatizzare
- non è richiesta alcuna “analisi del significato” (attacco precedente manuale)



# Il cifrario di Vigenère

---

Realizza uno “shift poli-alfabetico”

Osservazioni:

- gli attacchi statistici contro i cifrari mono-alfabetici sono possibili poiché la chiave definisce un’associazione FISSA, lettera per lettera, da applicare al testo in chiaro
- nei cifrari poli-alfabetici la chiave definisce una sostituzione da applicare a blocchi di caratteri del messaggio in chiaro

e.g., ab → DZ, ac → TY



non è associata ad un singolo  
carattere nel cifrato

**In questo modo essenzialmente  
la distribuzione delle frequenze  
viene “appianata”**



# Il cifrario di Vigenère

---

Il cifrario di Vigenere usa semplicemente “più istanze indipendenti dello shift cipher”.

chiave  $\longrightarrow$  stringa di caratteri

Plaintext:	tellhimaboutme
Key (repeated):	cafecafecafeca
Ciphertext:	VEQPJIREDOZXOE

**Ricordo che:**  
caratteri = numeri  
shift = + mod 26

$\downarrow$   
 $\downarrow$   
 $\downarrow$   
 $\downarrow$   
shift cipher con chiave “e”: 4°, 8°, 12°, ...  
shift cipher con chiave “f”: 3°, 7°, 11°, ...  
shift cipher con chiave “a”: 2°, 6°, 10°, ...  
shift cipher con chiave “c”: 1°, 5°, 9°, ...

---



# Il cifrario di Vigenère

---

Se la chiave è sufficientemente lunga, rompere il cifrario sembra impresa ostica.

È stato ritenuto inattaccabile per centinaia di anni.



# Rompere il cifrario di Vigenère

---

Per iniziare, si **assuma** che la lunghezza (detta *periodo*) della chiave  $k$  sia nota.

Sia  $k = k_1 k_2 \dots k_t$ , dove  $k_j$  è un carattere della chiave.

Si divida il cifrato  $c = c_1 c_2 \dots$  in  $t$  parti. Precisamente, per  $j=1, \dots, t$



Si calcoli la chiave  $k_j$ , per  $j=1, \dots, t$ , usando l'attacco efficiente contro lo shift cipher descritto in precedenza.

---

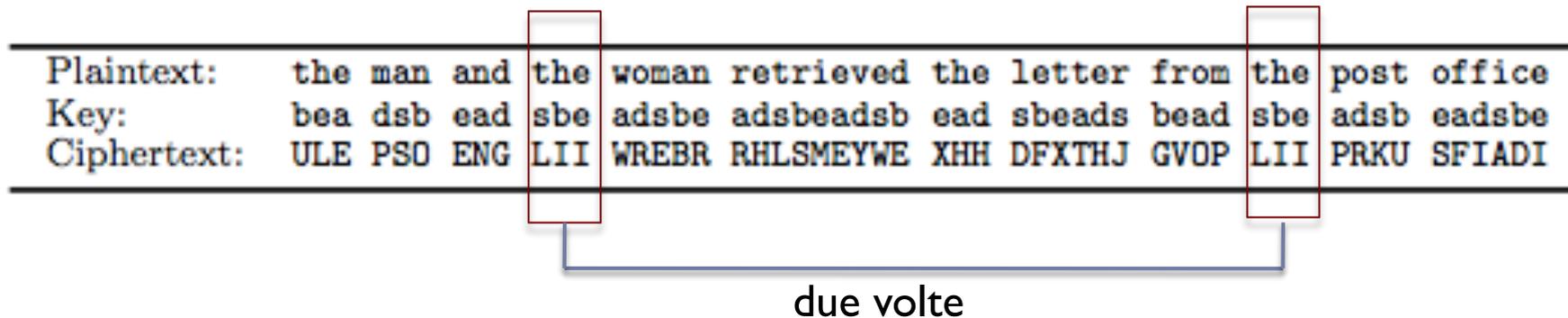


# Test di Kasiski

---

Facile, vero? E se invece la lunghezza della chiave non è nota? Che si fa?

1. Se è noto un massimo per la lunghezza, potremmo tentare tutte le ipotesi, una per una, i.e.,  $t=1$ ,  $t=2$ ,  $t=3$ , ...
2. Usare il metodo di Kasiski
  - cerca di individuare “**pattern ripetuti**”



# Test di Kasiski

---

La distanza tra i pattern ripetuti, assumendo che non sia un evento accidentale, deve essere un **multiplo** del periodo della chiave

Il **massimo comune divisore** delle distanze tra i pattern ripetuti restituirà il periodo (o un multiplo di esso)



limite superiore al periodo

Tuttavia, esiste un altro approccio al calcolo del periodo, maggiormente strutturato e facile da automatizzare.

È il metodo dell'indice di coincidenza.



# Indice di coincidenza

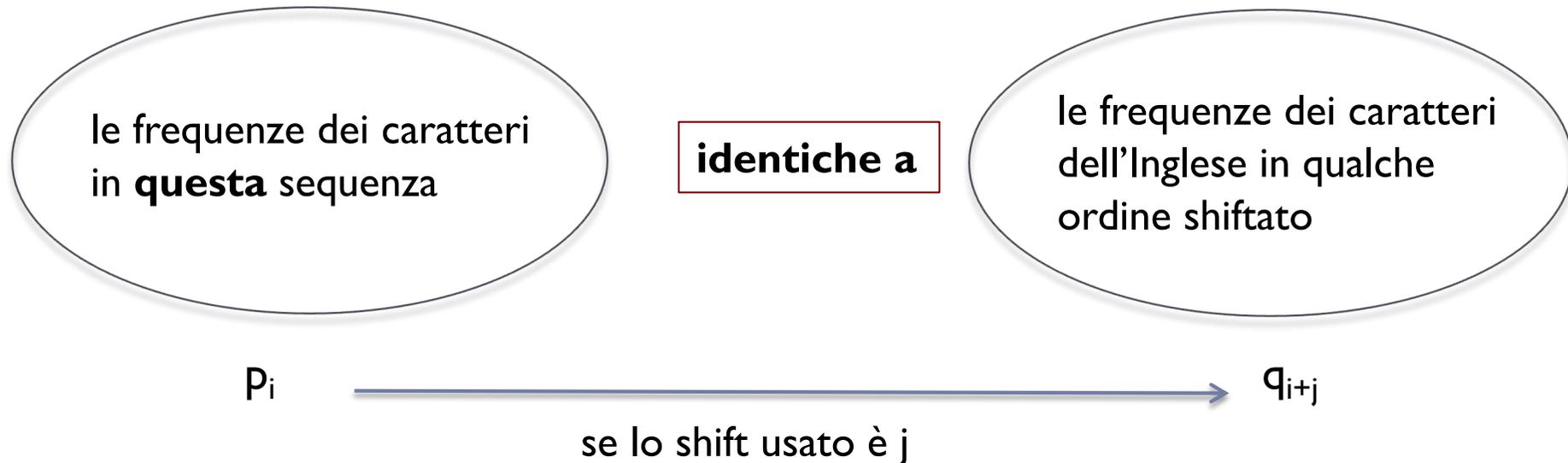
---

Se il periodo del cifrato  $c = c_1c_2\dots$  è  $t$ , allora

$c_1, c_{1+t}, c_{1+2t}, c_{1+3t}, \dots$

è la prima sequenza che risulta dalla cifratura ottenuta usando il “primo shift cipher”.

Come già osservato



# Indice di coincidenza

---

Ciò significa che  $q_0q_1\dots q_{25}$  è semplicemente  $p_0p_1\dots p_{25}$  “shiftata” di  $j$  posizioni.  
Pertanto, risulta

$$\sum_{i=0}^{25} q_i^2 \approx \sum_{i=0}^{25} p_i^2 \approx 0.065$$

Disponiamo, quindi, di un modo elegante per determinare la lunghezza  $t$  della chiave.  
Per  $w=1,2,3, \dots$  consideriamo la sequenza

$$C_1, C_{1+w}, C_{1+2w}, C_{1+3w} \dots$$

e calcoliamo i valori  $q_0q_1\dots q_{25}$ , e il valore

$$S_w = \sum_{i=0}^{25} q_i^2$$

---



# Indice di coincidenza

---

Ci aspettiamo che, quando  $w = t$ , risulta  $S_w \approx 0.065$ .

D'altra parte, se  $w \neq t$ , i caratteri  $c_1, c_{1+w}, c_{1+2w}, c_{1+3w} \dots$  sono presenti approssimativamente con la stessa frequenza, vicina a quella uniforme.

Vale a dire,  $q_i \approx 1/26$ , per  $i = 0, \dots, 25$ . Da cui risulta

$$S_w = \sum_{i=0}^{25} q_i^2 \approx \sum_{i=0}^{25} \left(\frac{1}{26}\right)^2 \approx 0.038$$

Pertanto, il valore più piccolo di  $w$  per cui  $S_w \approx 0.065$  è verosimilmente il periodo (lunghezza della chiave).

Una seconda sequenza  $c_2, c_{2+w}, c_{2+2w}, c_{2+3w} \dots$  può essere usata per confermare l'ipotesi

---



# Conclusioni

---

- l'attacco è elegante ed efficiente
- richiede un cifrato lungo per la stima delle frequenze
- una chiave più lunga implica la necessità di un cifrato più lungo

**Progettare cifrari sicuri non è impresa facile!**



# Team di supporto al corso

---

- ▶ **Link al team**

- ▶ [https://teams.microsoft.com/l/team/19%3a5cMNkd9jS8vaYuKjvTuoetBW3\\_0a6rljGh8K7bswZG01%40thread.tacv2/conversations?groupId=ada271a6-8cdb-4575-8366-2d4d5124eedc&tenantId=c30767db-3dda-4dd4-8a4d-097d22cb99d3](https://teams.microsoft.com/l/team/19%3a5cMNkd9jS8vaYuKjvTuoetBW3_0a6rljGh8K7bswZG01%40thread.tacv2/conversations?groupId=ada271a6-8cdb-4575-8366-2d4d5124eedc&tenantId=c30767db-3dda-4dd4-8a4d-097d22cb99d3)

- ▶ **Codice di accesso: naeyznb**

