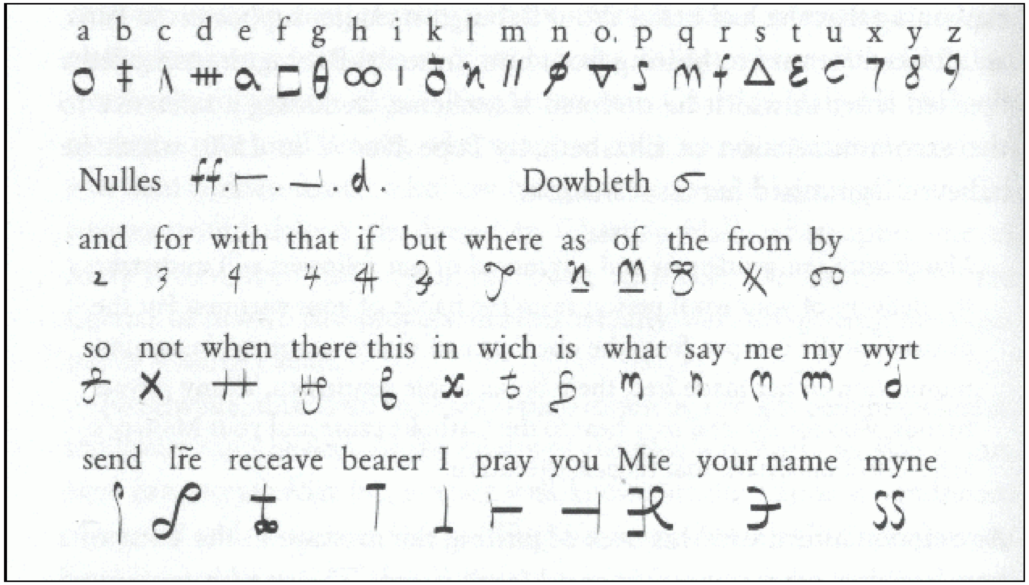
Historical cryptography



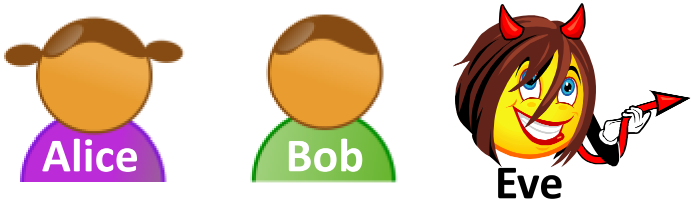
a b c d e f g h i k l m n o p q r s t u x y z

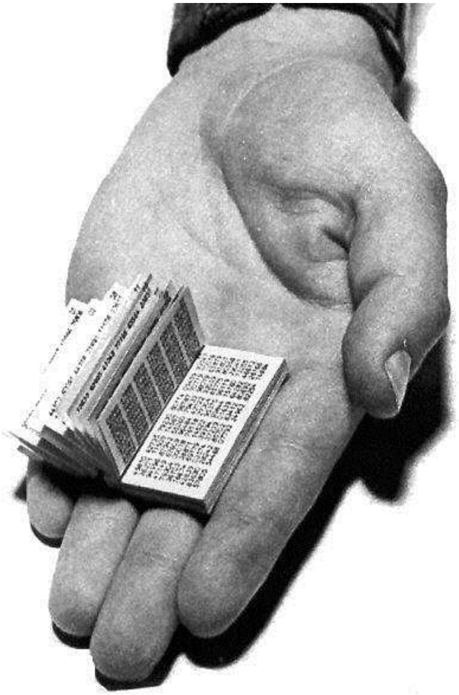
Nulles ff. ⊢ . ⊣ . d .   Dowbleth σ

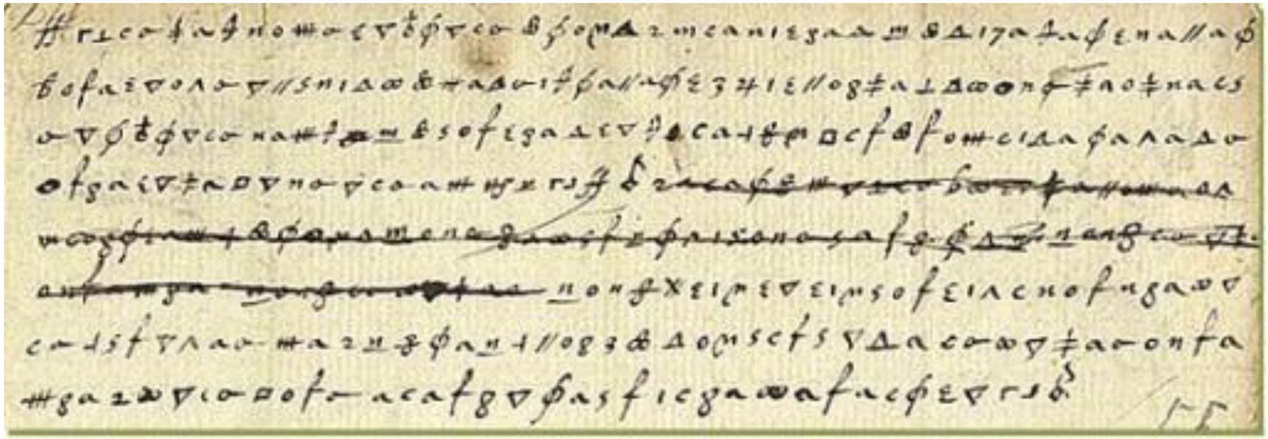
and for with that if but where as of the from by

so not when there this in wich is what say me my wyrt

send lr̃e receave bearer I pray you Mte your name myne

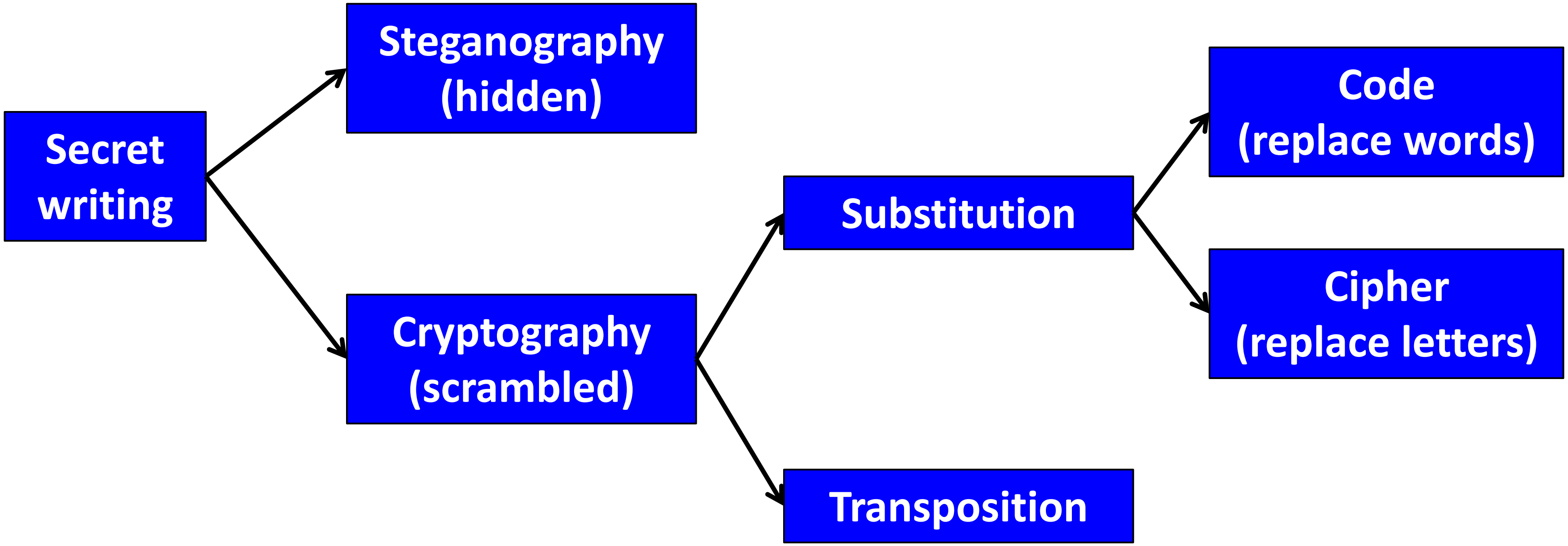






Overview

- Historical cryptography
  - Monoalphabetic substitution ciphers
    - Breaking them
    - Some improvements
    - The cipher of Mary Queen of Scots
  - Polyalphabetic substitution ciphers
  - Unbreakable encryption



Secret writing

Steganography (hidden)

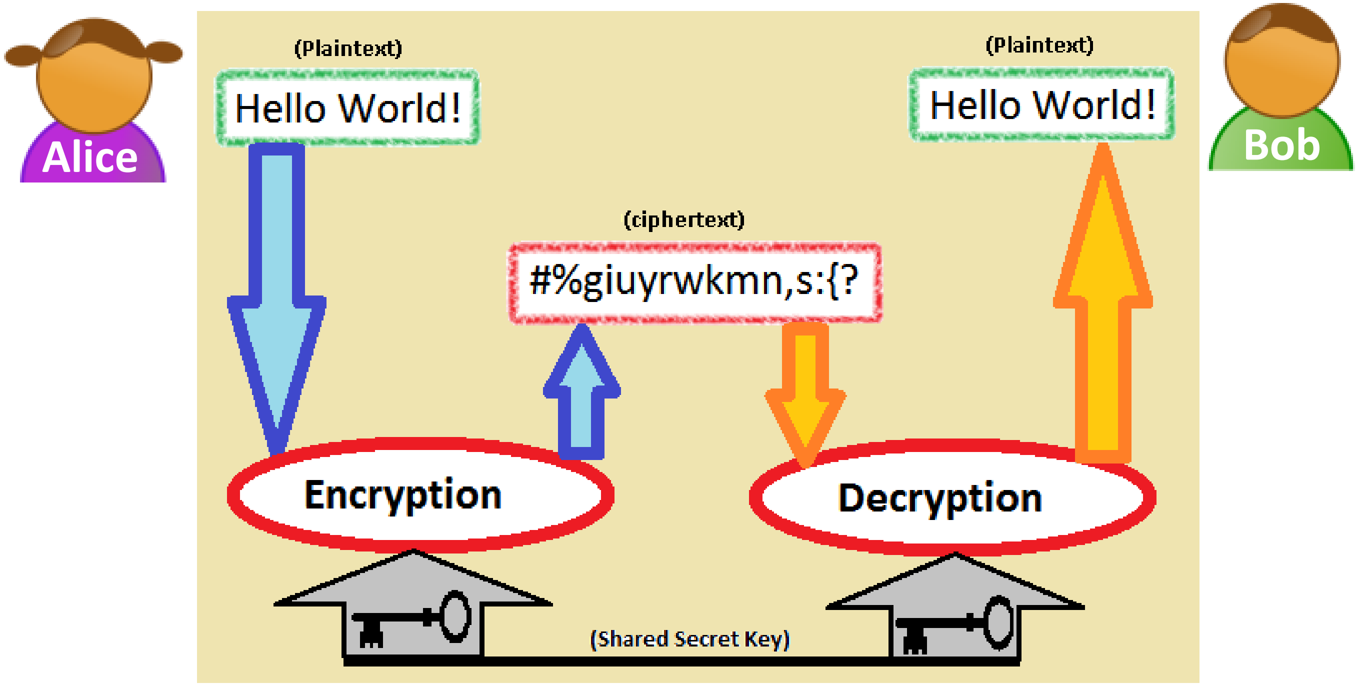
Cryptography (scrambled)

Substitution

Transposition

Code (replace words)

Cipher (replace letters)



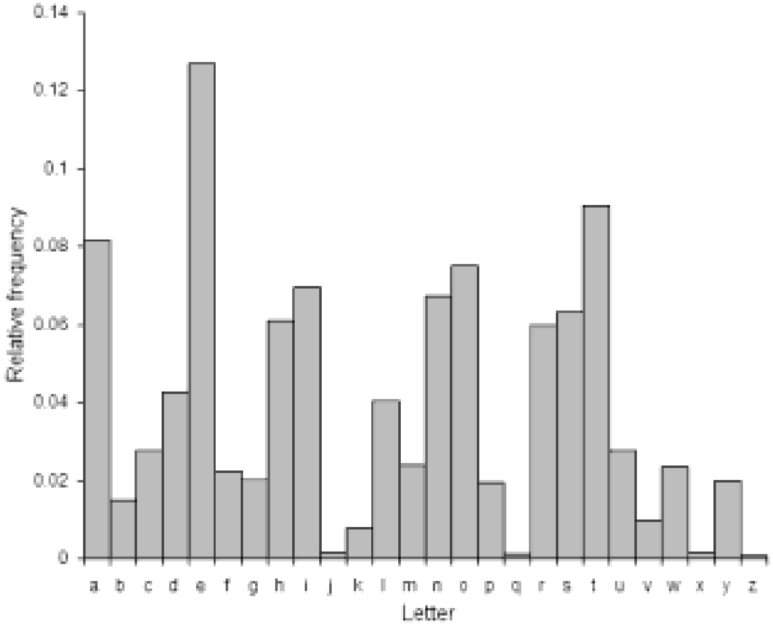
Monoalphabetic ciphers

- Monoalphabetic cipher
  - Use a fixed substitution over entire message
- Assigning substitutions
  - Option 1: Caesar shift cipher
  - Option 2: Completely random
    - 26! ways to assign ≈ 400,000,000,000,000,000,000,000,000
    - But hard to remember a completely random assignment
  - Option 3: Based on key phrase
    - Shared secret: "ugly black swan"

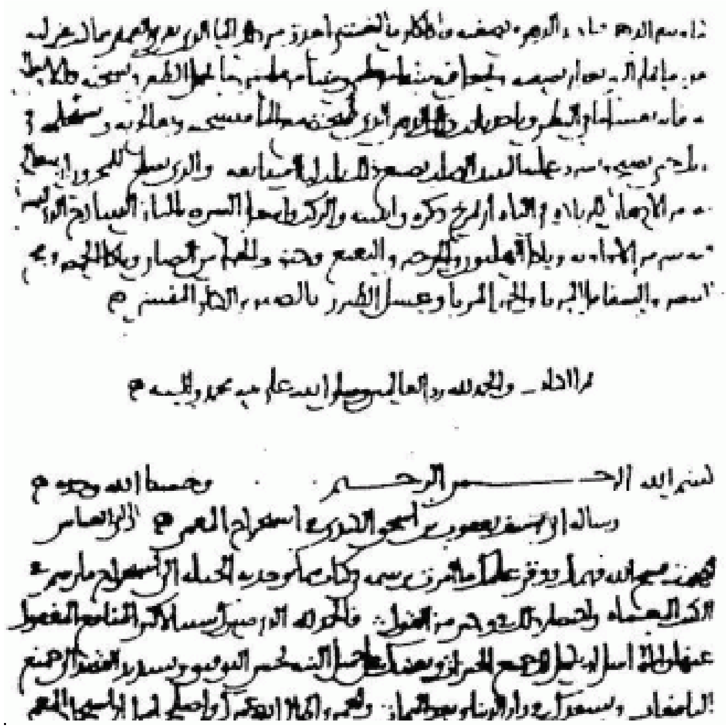
a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
U	G	L	Y	B	A	C	K	S	W	N	D	E	F	H	I	J	M	O	P	Q	R	T	V	X	Z

Monoalphabetic ciphers

- Dominated secret writing
  - Codemakers had a seemingly unbreakable code
    - No need for further innovation
  - At least for most of the first millennium AD
- Breaking monoalphabetic ciphers
  - Key idea: frequency analysis
    - Arabs ~800 AD
  - Easiest on long texts



Breaking a monoalphabetic cipher

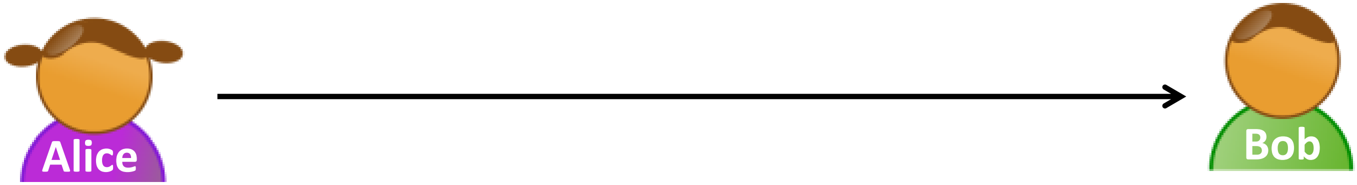


<span style="color:red">*On Deciphering Cryptographic Messages*</span>
<span style="color:red">*by Abu Yusuf Ya‘qūb ibn Isḥāq al-Sabbah al-Kindī*</span>

"One way to solve an encrypted message, if we know its language, is to find a different plaintext of the same language long enough to fill one sheet or so, and then we count the occurrences of each letter. We call the most frequently occurring letter the 'first', the next most occurring letter the 'second' the following most occurring letter the 'third', and so on, until we account for all the different letters in the plaintext sample.

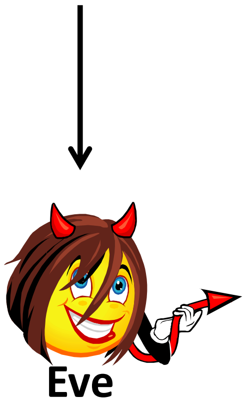
Then we look at the cipher text we want to solve and we also classify its symbols. We find the most occurring symbol and change it to the form of the 'first' letter of the plaintext sample, the next most common symbol is changed to the form of the 'second' letter, and the following most common symbol is changed to the form of the 'third' letter, and so on, until we account for all symbols of the cryptogram we want to solve."

Breaking a monoalphabetic cipher



```
LIVITCSWPIYVEWHEVSRIQMXLEYVEOIEWHRXEXIPFEMVEWHKVSTYLXZIXLIKIIX
PIJVSZEYPERRGERIMWQLMGLMXQERIWGPSRIHMXQEREKIETXMJTPRGEVEKEITRE
WHEXXLEXXMZITWAWSQWXSWEXTVEPMRXRSJGSTVRIEYVIEXCVMUIMWERGMIWXMJ
MGCSMWXSJOMIQXLIVIQIVIXQSVSTWHKPEGARCSXRWIEVSWIIBXVIZMXFSJXLIK
EGAEWHEPSWYSWIWIEVXLISXLIVXLIRGEPIRQIVIIBGIIHMWYPFLEVHEWHYPSRR
FQMXLEPPXLIECCIEVEWGISJKTVWMRLIHYSPHXLIQIMYLXSJXLIMWRIGXQEROIV
FVIZEVAEKPIEWHXEAMWYEPPXLMWYRMWXSGSWRMHIVEXMSWMGSTPHLEVHPFKPEZ
INTCMXIVJSVLMRSCMWMSWVIRCIGXMWYMX
```

Ciphertext (spaces removed)



Eve counts up frequency of:

single letters

letter pairs (bigrams)

letter triples (trigrams)

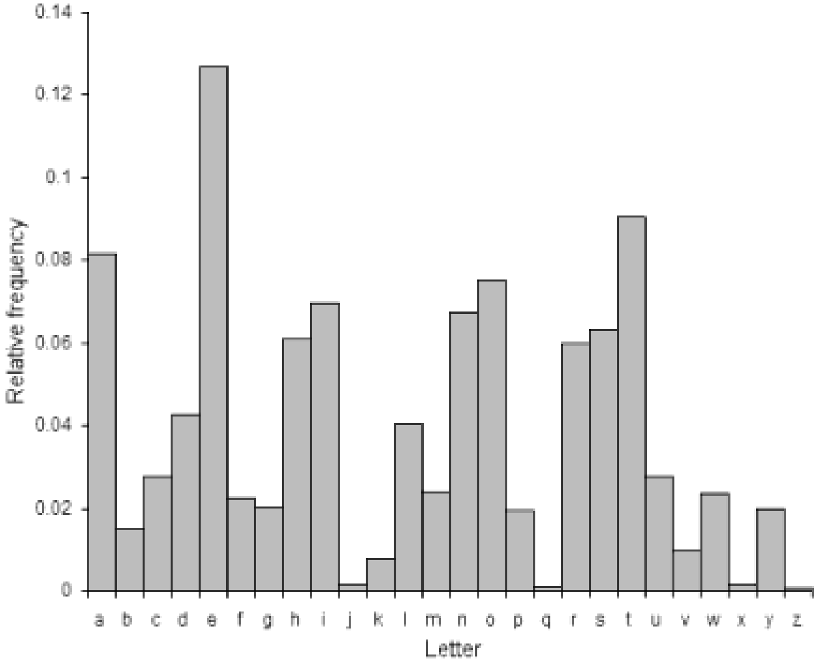
…

Breaking a monoalphabetic cipher: step 1

```
LIVITCSWPIYVEWHEVSRIQMXLEYVEOIEWHRXEXIPFEMVEWHKVSTYLXZIXLIKIIX
PIJVSZEYPERRGERIMWQLMGLMXQERIWGPSRIHMXQEREKIETXMJTPRGEVEKEITRE
WHEXXLEXXMZITWAWSQWXSWEXTVEPMRXRSJGSTVRIEYVIEXCVMUIMWERGMIWXMJ
MGCSMWXSJOMIQXLIVIQIVIXQSVSTWHKPEGARCSXRWIEVSWIIBXVIZMXFSJXLIK
EGAEWHEPSWYSWIWIEVXLISXLIVXLIRGEPIRQIVIIBGIIHMWYPFLEVHEWHYPSRR
FQMXLEPPXLIECCIEVEWGISJKTVWMRLIHYSPHXLIQIMYLXSJXLIMWRIGXQEROIV
FVIZEVAEKPIEWHXEAMWYEPPXLMWYRMWXSGSWRMHIVEXMSWMGSTPHLEVHPFKPEZ
INTCMXIVJSVLMRSCMWMSWVIRCIGXMWYMX
```

ciphertext	plaintext	
I	e	most common letter
XL	th	most common bigram
XLI	the	most common trigram
E	a	second most common letter

Eve



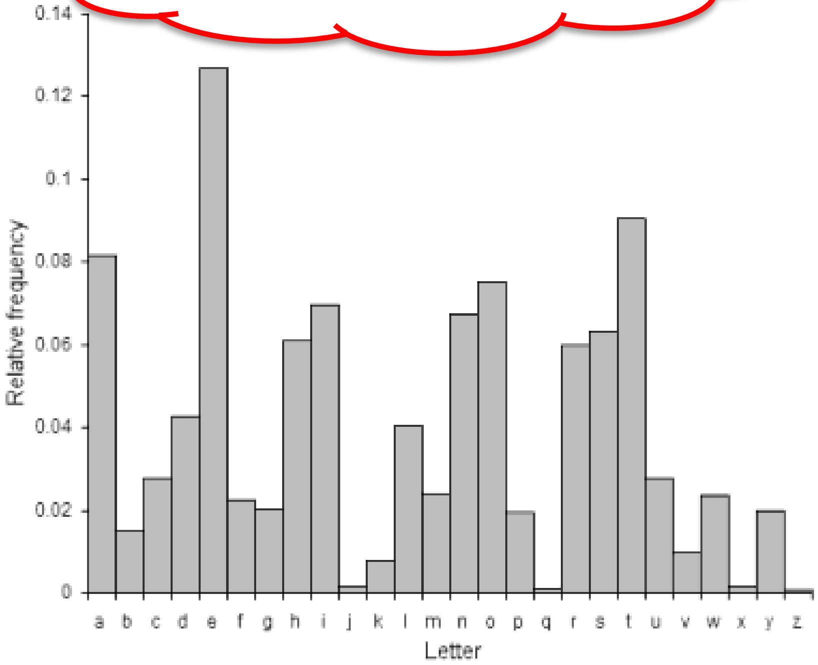
Letter distribution in English.

Breaking a monoalphabetic cipher: step 1

```
heVeTCSWPeYVaWHaVSReQMthaYVaOeaWHRtatePFaMVaWHKVSTYhtZetheKeet
PeJVSZaYPaRRGaReMWQhMGhMtQaReWGPSReHMtQaRaKeaTtMJTPRGaVaKaeTRa
WHatthattMZeTWAWSQWtSWatTVaPMRtRSJGSTVReaYVeatCVMUeMWaRGMeWtMJ
MGCSMWtSJOMeQtheVeQeVetQSVSTWHKPaGARCStRWeaVSWeeBtVeZMtFSJtheK
aGAaWHaPSWYSWeWeaVtheStheVtheRGaPeRQeVeeBGeeHMWYPFhaVHaWHYPSRR
FQMthaPPtheaCCeaVaWGeSJKTVWMRheHYSPHtheQeMYhtSJtheMWReGtQaROeV
FVeZaVAaKPeaWHtaAMWYaPPthMWYRMWtSGSWRMHeVatMSWMGSTPHhaVHPFKPaZ
eNTCMteVJSVhMRSCMWMSWVeRCeGtMWYMt
```

Eve now has a partially decoded message.

ciphertext	plaintext	
I	e	most common letter
XL	th	most common bigram
XLI	the	most common trigram
E	a	second most common letter



Letter distribution in English.

Breaking a monoalphabetic cipher: step 2

```
heVeTCSWPeYVaWHaVSReQMthaYVaOeaWHRtatePFaMVaWHKVSTYhtZetheKeet
PeJVSZaYPaRRGaReMWQhMGhMtQaReWGPSReHMtQaRaKeaTtMJTPRGaVaKaeTRa
WHatthattMZeTWAWSQWtSWatTVaPMRtRSJGSTVReaYVeatCVMUeMWaRGMeWtMJ
MGCSMWtSJOMeQtheVeQeVetQSVSTWHKPaGARCStRWeaVSWeeBtVeZMtFSJtheK
aGAaWHaPSWYSWeWeaVtheStheVtheRGaPeRQeVeeBGeeHMWYPFhaVHaWHYPSRR
FQMthaPPtheaCCeaVaWGeSJKTVWMRheHYSPHtheQeMYhtSJtheMWReGtQaROeV
FVeZaVAaKPeaWHtaAMWYaPPthMWYRMWtSGSWRMHeVatMSWMGSTPHhaVHPFKPaZ
eNTCMteVJSVhMRSCMWMSWVeRCeGtMWYMt
```

ciphertext	plaintext	cipher fragment	plaintext guess
V	r	heVe	here
R	s	Rtate	state
M	i	atthattMZe	atthattime
Z	m	atthattMZe	atthattime

Eve can now use her knowledge of language to make further guesses…

Breaking a monoalphabetic cipher

```
hereTCSWPeYraWHarSseQithaYraOeaWHstatePFairaWHKrSTYhtmetheKeet
PeJrSmaYPassGaseiWQhiGhitQaseWGPSseHitQasaKeaTtiJTPsGaraKaeTsa
WHatthattimeTWAWSQWtSWatTraPistsSJGSTrseaYreatCriUeiWasGieWtiJ
iGCSiWtSJOieQthereQeretQSrSTWHKPaGAsCStsWearSWeeBtremitFSJtheK
aGAaWHaPSWYSWeWeartheStherthesGaPesQereeBGeeHiWYPFharHaWHYPSss
FQithaPPtheaCCearaWGeSJKTrWisheHYSPHtheQeiYhtSJtheiWseGtQasOer
FremarAaKPeaWHtaAiWYaPPthiWYsiWtSGSWsiHeratiSWiGSTPHharHPFKPam
eNTCiterJSrhisSCiWiSWresCeGtiWYit
```

and so on…

Eve

Decoded monoalphabetic cipher

```
hereuponlegrandarosewithagraveandstatelyairandbroughtmethebeet
lefromaglasscaseinwhichitwasenclosedditwasabeautifulscarabaeusa
ndatthattimeunknowntonaturalistsofcourseagreatprizeinascientif
icpointofviewthereweretworoundblackspotsnearoneextremityoftheb
ackandalongoneneartheotherthescaleswereexceedinglyhardandgloss
ywithalltheappearanceofburnishedgoldtheweightoftheinsectwasver
yremarkableandtakingallthingsintoconsiderationicouldhardlyblam
ejupiterforhisopinionrespectingit
```

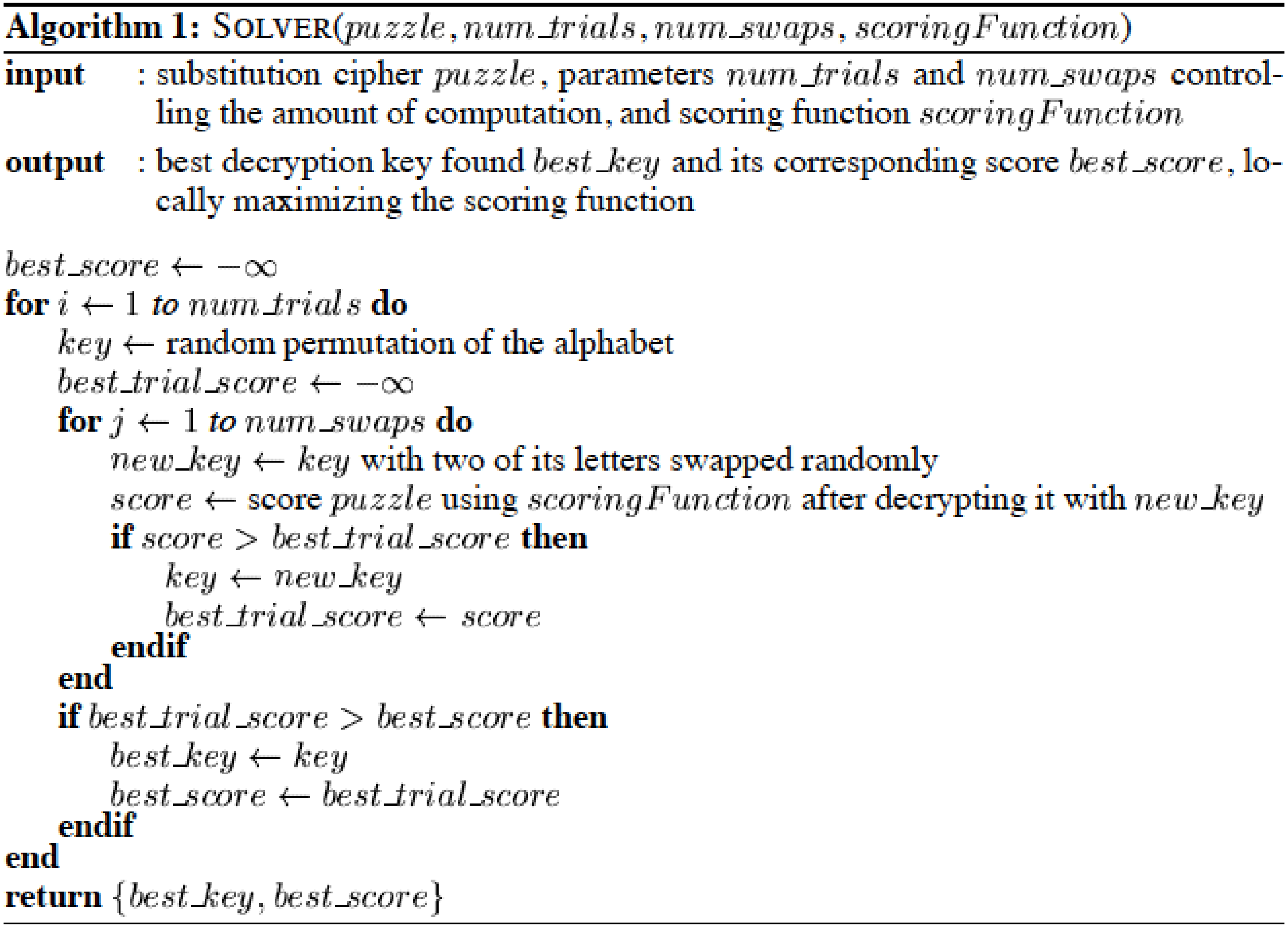
```
Hereupon Legrand arose, with a grave and stately air, and
brought me the beetle from a glass case in which it was
enclosed. It was a beautiful scarabaeus, and, at that time,
unknown to naturalists—of course a great prize in a scientific
point of view. There were two round black spots near one
extremity of the back, and a long one near the other. The
scales were exceedingly hard and glossy, with all the
appearance of burnished gold. The weight of the insect was very
remarkable, and, taking all things into consideration, I could
hardly blame Jupiter for his opinion respecting it.
```

The Gold Bug by Edgar Allan Poe.

Or use some code from the Internet…

```
c:\Dropbox\mtech\websci\resources>simpsub2.exe

Name of sample ("learning") file: moby.txt
Name of cipher file: mono2.txt
Is the cipher formatted with spaces? (y/n): n
Reading sample file...
Analyzing sample file...
Reading cipher file...
Analyzing cipher file...

Initial closeness is 1.487429, PLEASE WAIT...
DONE! Func value=0.866612

Key is: abcdefghijklmnopqrstuvwxyz
        ekghijylmdapzwscnvrxtoqbfu

hereuponlegrandarosewithagraveandstatelyairandbroughtmethebeetle
fromaglasscaseinwhichitwasencloseditwasabeautifulscarabaeusandat
thattimeunknowntonaturalistsofcourseagreatprizeinascientificpoin
tofviewtherewereworoundblackspotsnearoneextremityofthebackandal
ongonenearttheotherthescaleswereexceedinglyhardandglossywithallth
eappearanceofburnishedgoldtheweightoftheinsectwasveryremarkablea
ndtakingallthingsintoconsiderationicouldhardlyblamequpiterforhis
opinionrespectingit
```

Or develop your own program...



Algorithm 1: SOLVER($puzzle, num\_trials, num\_swaps, scoringFunction$)

input : substitution cipher $puzzle$, parameters num_trials and num_swaps controlling the amount of computation, and scoring function $scoringFunction$

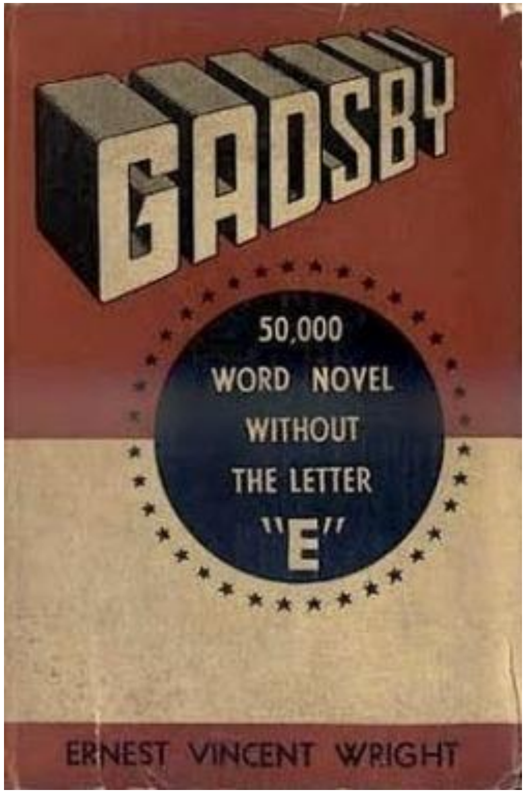
output : best decryption key found $best\_key$ and its corresponding score $best\_score$, locally maximizing the scoring function

```
$best\_score \leftarrow -\infty$
for $i \leftarrow 1$ to $num\_trials$ do
    $key \leftarrow$ random permutation of the alphabet
    $best\_trial\_score \leftarrow -\infty$
    for $j \leftarrow 1$ to $num\_swaps$ do
        $new\_key \leftarrow key$ with two of its letters swapped randomly
        $score \leftarrow$ score $puzzle$ using $scoringFunction$ after decrypting it with $new\_key$
        if $score > best\_trial\_score$ then
            $key \leftarrow new\_key$
            $best\_trial\_score \leftarrow score$
        endif
    end
    if $best\_trial\_score > best\_score$ then
        $best\_key \leftarrow key$
        $best\_score \leftarrow best\_trial\_score$
    endif
end
return $\{best\_key, best\_score\}$
```

Algorithm from “Solving Substitution Ciphers” by Sam Hasinoff

Shoring up monoalphabetic ciphers

- Improved resistance to frequency analysis:
  - Insert nulls, symbols that represent nothing
    - e.g. cipher alphabet 1-99, 73 numbers represent nulls
  - Mespall thangs on pirpus
    - Screws up frequency, humans can correct
  - Use code words
    - Need to exchange large dictionary of codes
    - Capture of codebook destroys security
  - Nomenclature
    - Small list of words or syllables
    - Cipher alphabet with homophones
  - Homophonic substitution
    - Multiple cipher symbols per plaintext symbol



Homophonic substitution

- Improved resistance to frequency analysis:
  - Homophonic substitution
    - For each plaintext symbol, set of cipher symbols
    - Set size proportional to frequency in the language

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
09	48	13	01	14	10	06	23	32	15	04	26	22	18	00	38	94	29	11	17	08	34	60	28	21	02
12	81	41	03	16	31	25	39	70			37	27	58	05	95		35	19	20	61		89		52	
33		62	45	24			50	73			51		59	07			40	36	30	63					
47			79	44			56	83			84		66	54			42	76	43						
53				46			65	88					71	72			77	86	49						
67				55			68	93					91	90			80	96	69						
78				57										99					75						
92				64															85						
				74															97						
				82																					
				87																					
				98																					

Mary Queen of Scots

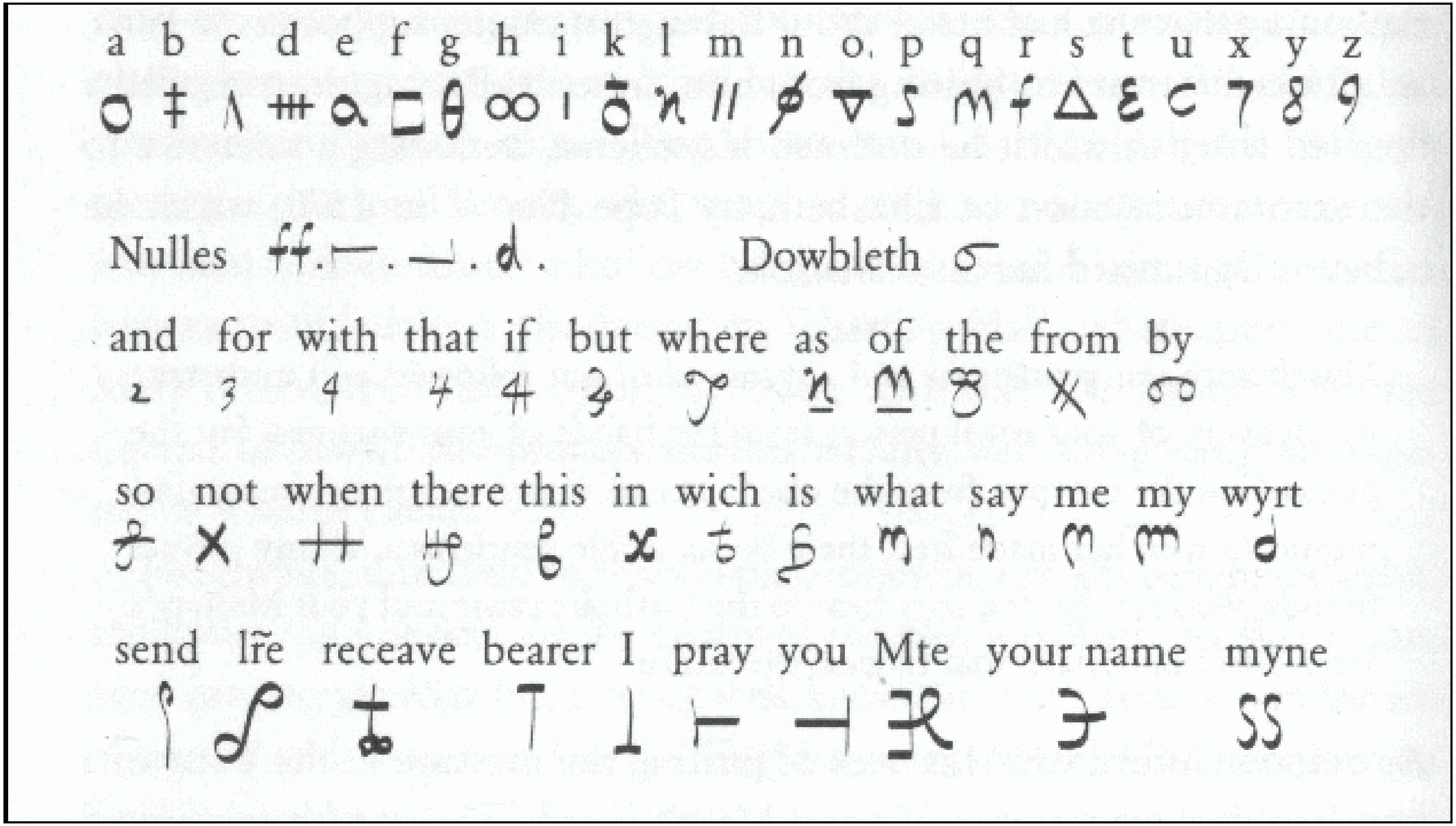
- Babington Plot
  - Mary imprisoned for 18 years
  - Gilbert Gifford: double agent
    - "recruited" to communicate with Mary
  - Detoured letters via Walsingham
  - Anthony Babington and company
    - Rescue Mary
    - Assassinate Elizabeth
    - Wanted blessing of Mary

Mary Queen of Scots

Elizabeth I

Francis Walsingham

Mary's nomenclature



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

Nulles

Dowbleth

and for with that if but where as of the from by

so not when there this in wich is what say me my wyrt

send lr̃e receave bearer I pray you Mte your name myne

http://simonsingh.net/The_Black_Chamber/maryqueenofscots.html

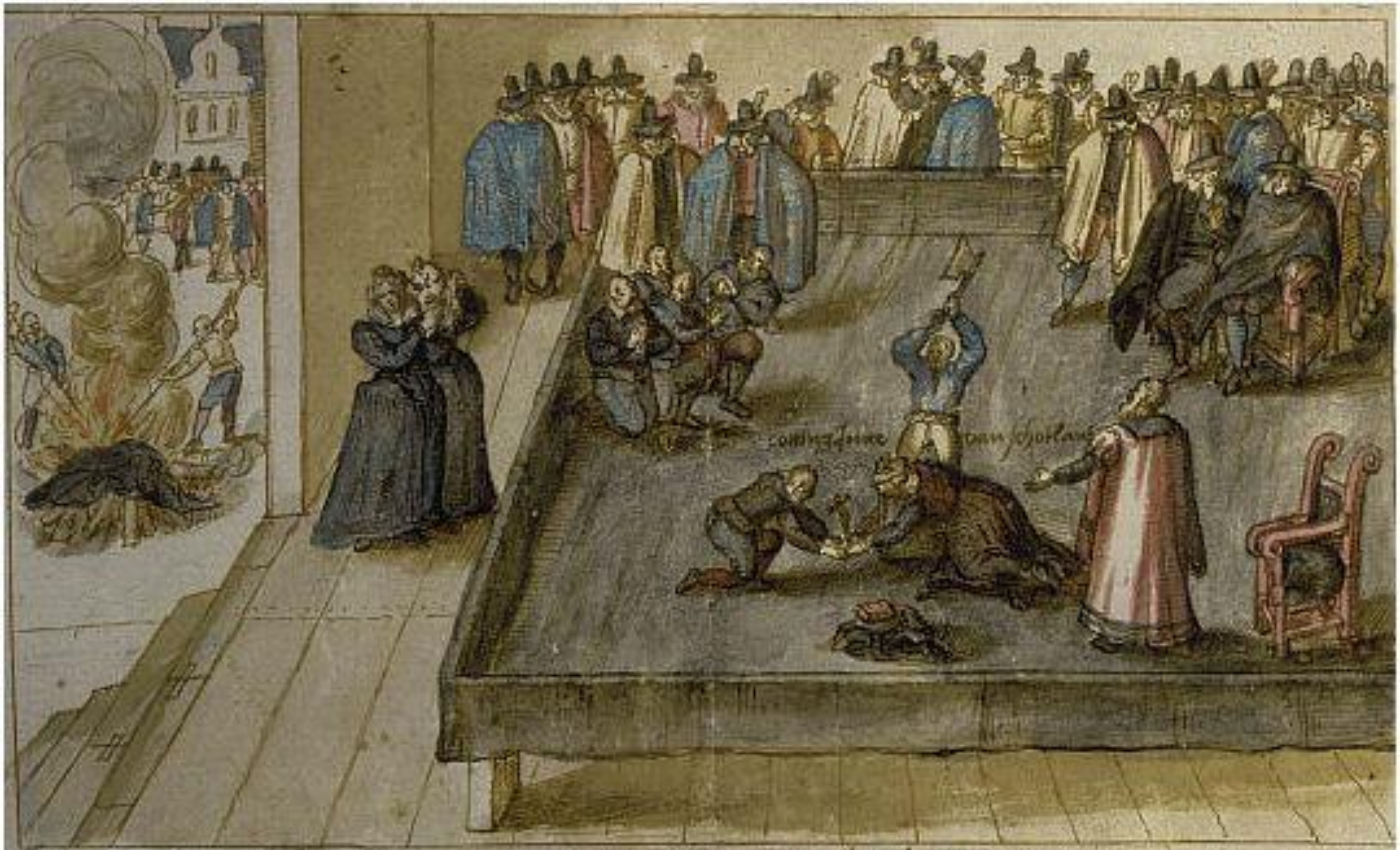
The plot

- Babington plot
  - Gifford delivers message from Mary to Babington
  - Babington replies with outline of plot:

> “Myself with ten gentlemen and a hundred of our followers will undertake the delivery of your royal person from the hands of your enemies. For the dispatch of the usurper, from the obedience of whom we are by the excommunication of her made free, there be six noble gentlemen, all my private friends, who for the zeal they bear to the Catholic cause and your Majesty's service will undertake that tragical execution”

  - Mary replies endorsing plan
    - Walsingham forges postscript, asking to name names:

> “I would be glad to know the names and qualities of the six gentlemen which are to accomplish the designment; for it may be that I shall be able, upon knowledge of the parties, to give you some further advice necessary to be followed therein, as also from time to time particularly how you proceed: and as soon as you may, the for the sample purpose, who be already, and how far everyone is privy hereunto.”



Den viii february werde onthalst Maria
Stuart Schots Conninginne sterwende Roomsch Catho-
lyck hebbende gesocht veel onrust ten eyn te crijghen haer schoon
meester te maecken van Engelant t'doodelijck haer vanden Raet
ofte parlement solemnelyck werde vertoont, Anno 1587.
Meteren xiii fol xiii en xiiii b

Polyalphabetic cipher

- Monoalphabetic cipher
  - – Single set of substitutions for all letters

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
U	G	L	Y	B	A	C	K	S	W	N	D	E	F	H	I	J	M	O	P	Q	R	T	V	X	Z

- Polyalphabetic cipher
  - – Multiple sets of substitutions
  - – Switch between them during encryption
  - – 1460s, Leon Alberti hits on idea of using 2+ sets

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
U	G	L	Y	B	A	C	K	S	W	N	D	E	F	H	I	J	M	O	P	Q	R	T	V	X	Z
T	H	E	Q	U	I	C	K	B	R	O	W	N	F	X	J	M	P	S	V	L	A	Z	Y	D	G

Polyalphabetic cipher

- 1586, Vigenère cipher, "Le Chiffre Indéchiffrable"
  - Letters Caesar shifted, change based on keyword

	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
A	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
B	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	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	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

Blaise de Vigenère

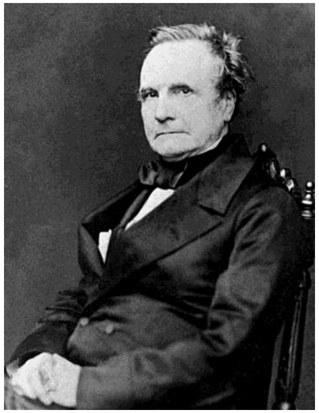
Plaintext	attackatdawn
Key	LEMONLEMONLE
Ciphertext	LXFOPVEFRNHR

Breaking the Vigenère Cipher

- Vigenère cipher
  - Much better at hiding letter frequency info
  - But key repeats:
    - If you know length, an interwoven set of Caeser ciphers

Key:	ABCDABCDABCDABCDABCDABCDABCD
Plaintext:	**crypto**isshortfor**crypto**graphy
Ciphertext:	**CSASTP**KVSIQUTGQU**CSASTP**IUAQJB

- Distance between repeats = 16
- Suggests key length if 16, 8, 4, 2, or 1
- Find additional repeats to narrow lengths
- Frequency analyze each interwoven set



Charles Babbage

Long keys

- Polyalphabetic with |key| = |message|
  - Babbage's method won't work

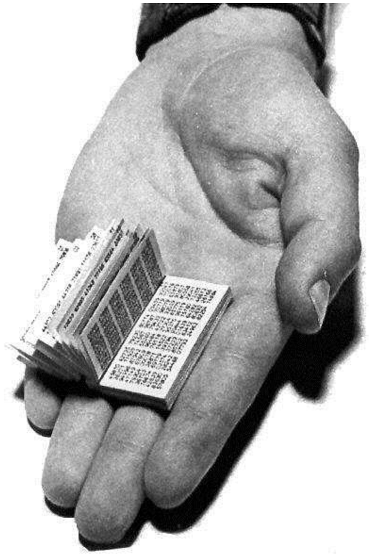
Key:	CAN???BSJ?????YPT????
Plaintext:	the???the?????the????
Ciphertext:	VHRMHEUZNFQDEZRWXFIDK

Key:	CAN?????????CRYPT????
Plaintext:	the?????????cithe????
Ciphertext:	VHRMHEUZNFQDEZRWXFIDK

Key:	CAN?????????EGYPT????
Plaintext:	the?????????atthe????
Ciphertext:	VHRMHEUZNFQDEZRWXFIDK

Key:	CANADABRAZILEGYPTCUBA
Plaintext:	themeetingisatthedock
Ciphertext:	VHRMHEUZNFQDEZRWXFIDK

Unbreakable encryption



- One-time pad, 1882
  - – Use a key as long as the message
  - – Choose key (truly) randomly
  - – Use key once and only once
  - – Provably secure

```
        h        e        l        l        o    message
     7 (H)    4 (E)   11 (L)   11 (L)   14 (O)   message
  + 23 (X)   12 (M)    2 (C)   10 (K)   11 (L)   key
  = 30       16       13       21       25       message + key
  =  4 (E)   16 (Q)   13 (N)   21 (V)   25 (Z)   message + key (mod 26)
        E        Q        N        V        Z    ciphertext
```

```
        E        Q        N        V        Z    ciphertext
     4 (E)   16 (Q)   13 (N)   21 (V)   25 (Z)   ciphertext
 -  23 (X)   12 (M)    2 (C)   10 (K)   11 (L)   key
 = -19        4       11       11       14       ciphertext - key
 =   7 (H)    4 (E)   11 (L)   11 (L)   14 (O)   ciphertext - key (mod 26)
        h        e        l        l        o    message
```

Breaking one-time pads?

- Try all possible keys
  - 26^{length} = big
  - Also: generates all possible text sequences

Correct key

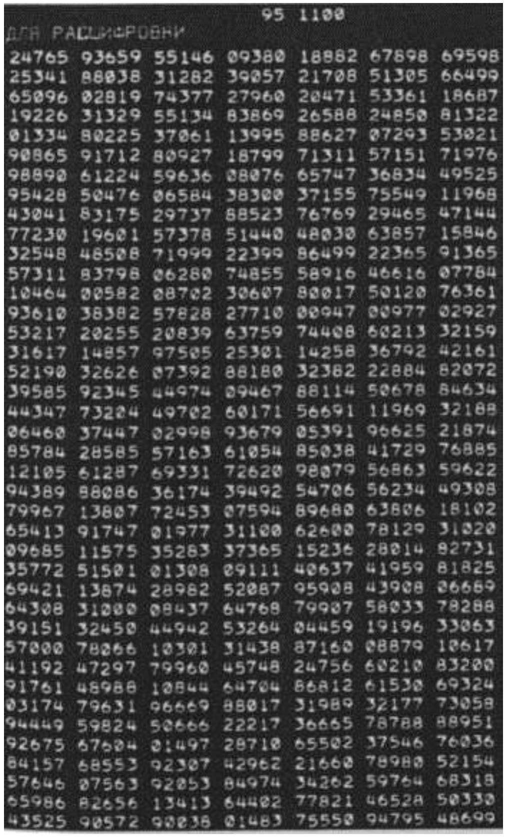
```
      E         Q         N         V         Z    ciphertext
    4 (E)    16 (Q)    13 (N)    21 (V)    25 (Z)  ciphertext
-  23 (X)    12 (M)     2 (C)    10 (K)    11 (L)  key
= -19         4        11        11        14      ciphertext - key
=   7 (H)     4 (E)    11 (L)    11 (L)    14 (O)  ciphertext - key (mod 26)
      h         e         l         l         o    message
```

Some other key

```
      E         Q         N         V         Z    ciphertext
    4 (E)    16 (Q)    13 (N)    21 (V)    25 (Z)  ciphertext
-  19 (T)    16 (Q)    20 (U)    17 (R)     8 (I)  possible key
= -15         0        -7         4        17      ciphertext-key
=  11 (L)     0 (A)    19 (T)     4 (E)    17 (R)  ciphertext-key (mod 26)
      l         a         t         e         r    possible message
```

Unbreakable encryption

- Problems with one-time pads:
  - Must distribute pads securely
    - If captured, code is useless
  - Must use truly random numbers
    - Not pseudo-random
    - Not random typing on a keyboard
  - Must never reuse the same key



```
                          95 1100
ДЛЯ РАСШИФРОВКИ
24765 93659 55146 09380 18882 67898 69598
25341 88038 31282 39057 21708 51305 66499
65096 02819 74377 27960 20471 53361 18687
19226 31329 55134 83869 26588 24850 81322
01334 80225 37061 13995 88627 07293 53021
90865 91712 80927 18799 71311 57151 71976
98890 61224 59636 08076 65747 36834 49525
95428 50476 06584 38300 37155 75549 11968
43041 83175 29737 88523 76769 29465 47144
77230 19601 57378 51440 48030 63857 15846
32548 48508 71999 22399 86499 22365 91365
57311 83798 06280 74855 58916 46616 07784
10464 00582 08702 30607 80017 50120 76361
93610 38382 57828 27710 00947 00977 02927
53217 20255 20839 63759 74408 60213 32159
31617 14857 97505 25301 14258 36792 42161
52190 32626 07392 88180 32382 22884 82072
39585 92345 44974 09467 88114 50678 84634
44347 73204 49702 60171 56691 11969 32188
06460 37447 02998 93679 05391 96625 21874
85784 28585 57163 61054 85038 41729 76885
12105 61287 69331 72620 98079 56863 59622
94389 88086 36174 39492 54706 56234 49308
79967 13807 72453 07594 89680 63806 18102
65413 91747 01977 31100 62600 78129 31020
09685 11575 35283 37365 15236 28014 82731
35772 51501 01308 09111 40637 41959 81825
69421 13874 28982 52087 95908 43908 06689
64308 31000 08437 64768 79907 58033 78288
39151 32450 44942 53264 04459 19196 33063
57000 78066 10301 31438 87160 08879 10617
41192 47297 79960 45748 24756 60210 83200
91761 48988 10844 64704 86812 61530 69324
03174 79631 96669 88017 31989 32177 73058
94449 59824 50666 22217 36665 78788 88951
92675 67604 01497 28710 65502 37546 76036
84157 68553 92307 42962 21660 78980 52154
57646 07563 92053 84974 34262 59764 68318
65986 82656 13413 64402 77821 46528 50330
43525 90572 90038 01483 75550 94795 48699
```

> "As a practical person, I've observed that one-time pads are theoretically unbreakable, but practically very weak. By contrast, conventional ciphers are theoretically breakable, but practically strong."
>
> -Steve Bellovin

Summary

- History of cryptography
  - Substitution ciphers
    - Monoalphabetic
    - Polyalphabetic
  - One-time pads
    - Provably unbreakable
      - (if used carefully)