Begleitmaterial zu “Cryptography”

Leon Aaron Kaplan aaron@lo-res.org, Donau Uni Krems, Dez. 2004

Übungsaufgabe / Gruppenarbeit

Aufgabe 1

Gegeben sei folgende mit dem (Caesar System) verschlüsselte englische Nachricht:

FRPGVBA 1. BS GUR ANGHER BS SYNGYNAQ

V PNYY BHE JBEYQ SYNGYNAQ, ABG ORPNHFR JR PNYY VG FB,
OHG GB ZNXR VGF ANGHER PYRNERE GB LBH, ZL UNCCL ERNQREF,
JUB NER CEVIVYRTRQ GB YVIR VA FCNPR.

VZNTVAR N INFG FURRG BS CNCRE BA JUVPU FGENVTUG YVARF, GEVNATYRF,
FDHNERF, CRAGNTBAF, URKNTBAF, NAQ BGURE SVTHERF, VAFGRNQ BS ERZNVAVAT
SVKRQ VA GURVE CYNPRF, ZBIR SERRYL NOBHG, BA BE VA GUR FHESNPR,
OHG JVGUBHG GUR CBJRE BS EVFVAT NOBIR BE FVAXVAT ORYBJ VG, IREL ZHPU
YVXR FUNQBJF -- BAYL UNEQ NAQ JVGU YHZVABHF RQTRF -- NAQ LBH JVYY GURA
UNIR N CERGGL PBEERPG ABGVBA BS ZL PBHAGEL NAQ PBHAGELZRA. NYNF,
N SRJ LRNEF NTB, V FUBHYQ UNIR FNVQ "ZL HAVIREFR": OHG ABJ ZL ZVAQ
UNF ORRA BCRARQ GB UVTURE IVRJF BS GUVATF.

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

Knacken Sie den Code!

Aufgabe 2: Vigenère

Lesen Sie den folgenden Text. Erklären Sie, wie man am Ende auf die Substitution kam. Wie viele Möglichkeiten blieben nach dem Kasiski Verfahren für die Substitution? War es einfach die Substitution zu finden? Wie verhält sich das mit kürzeren / längeren Texten?

```
this is a secret message. It is only for your eyesX
abab ab a bababa bababab  ab ab abab aba baba babab
--------------------------------------------------
tiit it a tedrft netsbgf. Iu it oolz fpr zovr fyfsy
1234 56 7 890123 4567890  12 34 5678 901 2345 67890
0           1          2              3           4
```

Patterns:
“It” (Positionen 3,5 und 23), -> abstand: 5-3 = 2, abstand 23-5 = 8 = 2*2*2 => Faktor ist 2. Schlüssellänge könnte 2 sein.

Aufteilen des Textes in 2er Gruppen

```
T i
I t
I t
A t
E d
R f
T n
E t
S b
G f
I u
I t
O o
L z
F p
R z
O v
R f
Y f
S y
```

```
Erste Spalte:
Tiiaertesgiiolfrorys
2te:
itttdfntbfutozpzvffy
```

Häufigkeitsanalyse, Spalte 1

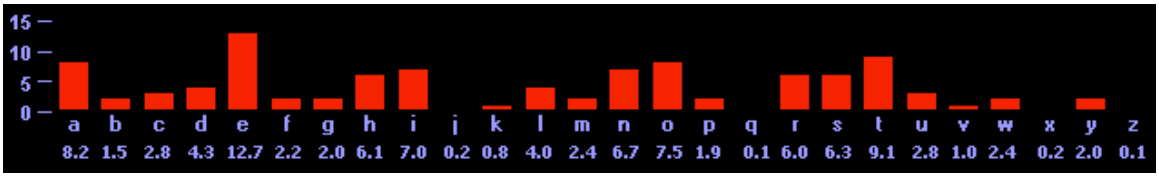
Buchstabe	#	rel Häufigk	Prozent
I	4	0.2	20%
R	3	.15	15%
T	2	.1	10%
E	2	.1	10%
O	2	.1	10%
S	2	.1	10%
A	1	.05	5%
G	1	.05	5%
L	1	.05	5%
F	1	.05	5%
Y	1	.05	5%
Summe:	20 ok		

Häufigkeitsanalyse, Spalte 2 ("itttdfntbfutozpzvffy")

T	5	0.25	25%
F	4	0.20	20%
Z	2	0.1	10%
I	1	0.05	5%
D	1	0.05	5%
N	1	0.05	5%
B	1	0.05	5%
U	1	0.05	5%
O	1	0.05	5%
P	1	0.05	5%
V	1	0.05	5%
Y	1	0.05	5%
Summe:	20	OK	

Versuch und Vergleich mit (engl. Häufigkeitstabelle):

e	t	a	o	i	n	s	h	r	d	l	u	c
12.7	9.1	8.2	7.5	7.0	6.7	6.3	6.1	6.0	4.3	4.0	2.8	2.8
m	w	f	y	g	p	b	v	k	x	j	q	z
2.4	2.4	2.2	2.0	2.0	1.9	1.5	1.0	0.8	0.2	0.2	0.1	0.1



Wir versuchen die substitution I<- i, r<-r, e<-e, a<-a, etc und erhalten für die 1te Spalte:

Tiiaertesgiiolfrorys
Tiiaertesgiiolfrorys

Für die 2te Spalte ersetzen wir: e<-f, s<-t, etc… und erhalten:

`hssscemsaetsnyoyueex`

Wir wandeln um in Spaltenschreibweise

```
T h
I s
I s
A s
E c
R e
T m
E s
S a
G e
I t
I s
O n
L y
F o
R y
O u
R e
Y e
S x
```

Abgelesen ergibt das den klartext

Aufgabe 3: XOR

Lösen Sie in der Gruppe!
Eigenschaften von XOR:

$x \oplus 1 = \neg x$
$x \oplus 0 = x$

⊕	**0**	**1**
0	0	1
1	1	0

```
00000000  44 61 73 20 69 73 74 20  65 69 6e 20 67 65 68 65  |Das ist ein gehe|
00000010  69 6d 65 72 20 54 65 78  74 0a                    |imer Text.|
0000001a
```

der Text wurde mittels ⊕ "123" nach Vigenère verschlüsselt:

```
instabil:~/Desktop/work/uni_krems/sw/src aaron$ hexdump -C simple.txt.xor
00000000  75 53 40 11 5b 40 45 12  56 58 5c 13 56 57 5b 54  |uS@.[@E.VX\.VW[T|
00000010  5b 5e 54 40 13 65 57 4b  45 38                    |[^T@.eWKE8|
0000001a
```

Angenommen, Sie wissen nur, dass die Schlüssellänge 3 ist. Wie können Sie ohne den Schlüssel zu kennen, auf den Klartext schließen?

Arbeiten Sie in der Gruppe eine Idee aus und beweisen Sie auch warum der Trick funktioniert. Sie können, wenn Sie wollen, obigen Bsptext zur Überprüfung Ihrer Vermutung heranziehen.
Wie ist das, wenn der Schlüssel genauso lang ist wie der Klartext?

Mathematik Teil

Aufgabe 4: ggT(a,b)

Sagen Sie Ihrem Gegenüber 2 mindestens 3-stellige Zahlen (keine trivialen wie 2^8 und 2^9) und umgekehrt. Berechnen Sie jeweils nach dem Euklid. Algorithmus den ggT. Anschließend versuchen Sie bitte gemeinsam herauszufinden, was die Komplexität (worst-case complexity) des euklid. Algorithmus ist(*). Ist sie gut oder schlecht? Beginnen Sie mit der Annahme, daß jede Grundrechnungsart die Komplexität O(1) hat.

Kann der euklid. Algorithmus für die Berechnung des ggT von sehr großen Zahlen (200-1000 stellig) verwendet werden? Wie ist hier der Aufwand, wenn die Grundrechnungsarten nicht mehr O(1) sind?

(*) Hint: man überlege sich, was die längste Kette von Operationen sein kann und konstruiere geignete Beispiele wie zB: ggT(2*3*5*7*...*37, 2*3*5*...*37*41).

Aufgabe 5: Sieb

Warum funktioniert das Sieb des Erastosthenes gut bis (mindestens!) 10^9?
Hint: Vgl. Sie mit den Cache-sizes von gängigen CPUs mit der Anzahl der notwendigen Operationen und der Speicherkomplexität
Wie weit kann man mit dem Sieb auf 64 Bit Prozessoren kommen?

Public Key Crypto Teil

Aufgabe: RSA und DH

Warum kann RSA für Signaturen verwendet werden und Diffie-Hellman nicht? Erklären Sie Ihrem Gegenüber! Versuchen Sie eine Erweiterung für DH zu finden, sodas Signaturen möglich sind.

Spielen Sie weiters den RSA Algorithmus Schritt für Schritt für folgende Zahlen durch: Alice möchte an Bob die Nachricht 688 schicken. Bob hat p = 47 und q = 71 gewählt. Also ist n = 47*71 = 3337. Weil e teilerfremd zu (p-1)*(q-1) = 46*70 = 3220 sein muss können wir irgendein e wählen (mit dem Euklid. Algorithmus bestimmen!), sodas dies der Fall ist. Bob hat e = 79 gewählt. Berechnen Sie d (mit dem Wissen, das sie schon haben). Es sollte 1019 rauskommen.
Verschlüsseln Sie anschließend die Nachricht an Bob. Ihr Kollege nimmt dann die Rolle von Bob ein und entschlüsselt die Nachricht wieder.

Falls die Zahlen zu groß sind, dann probieren Sie es vorher mit n = 13*31, e = 17, d = 233, m = “?” (in ASCII) = 63 (in dezimal).

Aufgabe: Digitale Signatur

Das vorgestellte digitale Signaturverfahren hat einen Nachteil: man kann nicht wissen, wann die Nachricht geschickt wurde. Erweitern Sie das Verfahren, sodass wir sicher gehen können, dass die Nachricht 1) von Alice kam und 2) wirklich um diese Zeit weggeschickt wurde.

Originalliteratur zu Vigenère Ciphers

Exzerpt aus: Codes, Ciphers, & Codebreaking, Greg Goebel,
http://www.vectorsite.net/ttcode.html

[3.2] CRACKING THE VIGENERE CIPHER

* The Vigenere cipher was one of the first field ciphers to be widely adopted, with overhead generally reduced through use of a "cipher disk", a device described in a later chapter. The Vigenere cipher was regarded as unbreakable and perfectly secure for such communications, at least if certain precautions were taken. It was known as the "le chiffre indechiffrable (the indecipherable cipher)". In reality, it was not indecipherable, since a solution to the Vigenere cipher was first published by a retired Prussian Army officer named Friedrich Wilhelm Kasiski (1805:1881) in 1863, and his solution is appropriately known as the "Kasiski test".

A Vigenere cipher cannot be cracked by simple frequency analysis, but Kasiski found an indirect method. The Vigenere cipher, as discussed previously, allows the same letter to be enciphered in a number of different ways. Given the cipher key word "WARTHOG", for example, the letter "e" would be enciphered seven different ways:

```
W: e -> A
A: e -> E
R: e -> V
T: e -> X
H: e -> L
O: e -> S
G: e -> K
```

Similarly, the same string of plaintext could be enciphered in seven different ways, depending on which letter of the cipher key matches the beginning of the string, or in other terms what step of the "cycle" of different substitution alphabets was in effect. This means that common strings, such as "the" or "-ing", might be enciphered in exactly the same way in any fairly large message or set of messages.

Given the WARTHOG key, for example, if the plaintext word "the" is enciphered as "AVK", then if it occurs exactly 7, 14, 21, ... letters later in the plaintext message, it will also be enciphered as "AVK". Such repetitive patterns can be used to get a fingerhold into a cipher.

Kasiski's attack on the Vigenere cipher involved two insights. The first was that, as just shown, repetitive patterns in messages encrypted by a Vigenere cipher gave a hint as to the length of the key. It wasn't of immediate importance what the repetitive pattern actually was; only that it was repeated on a certain interval.

Suppose Alice encrypts a message using the cipher key word "WARTHOG". WARTHOG has seven letters. For a plaintext string occurring several times to be encrypted into the same ciphertext patterns several times, the plaintext string has to begin at exactly the same step in the cycle of substitution alphabets. Given the WARTHOG key, this means that these patterns will repeat in the ciphertext with a spacing of an exact multiple of seven letters apart. Of course, this pattern can be confounded by the generation of the same ciphertext pattern from an entirely different plaintext string at a different step in the cycle, but this becomes less likely with longer strings, and such coincidences are more in the nature of noise than insurmountable obstacles.

Suppose Holmes, our codebreaker, finds a number of repetitive ciphertext patterns in a set of messages encrypted with a Vigenere cipher using a common cipher key, and the patterns repeat at the given intervals:

pattern	interval
GIKK	133
YHDX	140
VXMA	1,190
POLQK	3,341
MOGTZL	4,550

We assume here for simplicity that Holmes finds only one repetition of each pattern, though of course in reality there may be several repetitions of the same pattern, and each is likely to be at a different interval. Holmes then factors the intervals into primes:

pattern	interval	factoring
GIKK	133	7 * 19
YHDX	140	2 * 2 * 5 * 7
VXMA	1,190	2 * 5 * 7 * 17
POLQK	3,341	3 * 7 * 23
MOGTZL	4,550	2 * 5 * 5 * 7 * 13

He searches for primes because they represent the smallest possible length of the key could be. For example, since the ciphertext pattern GIKK repeats on an interval of 133, that means the key could be 133, or 7, or 19 letters long. Since the actual (if still unknown) key length must be common to all five of the repetitive patterns Holmes finds in this message, he can narrow down the possible lengths to those factors that are common to all the patterns. The lists

of factors at right show that all include the value 7, which is a clue, a really big one, that the length of the cipher key is seven.

There is, of course, no reason to assume that the key length will actually be a prime number, but that doesn't matter. Whatever the actual length is, all of its factors will appear in all the factorings of the interval lengths. For example, if all of the pattern intervals had the common prime factors of 3 and 7, that could mean that the key is 3, 7, or (most likely) 3 * 7 = 21 letters long.

There is, again, also no saying that one text pattern, say VXMA, might not be produced by coincidence from two entirely unrelated four-letter plaintext strings, but this becomes less likely as the pattern grows longer. If Holmes finds a set of repeating patterns and most point to a particular key length, he would sensibly judge that the few patterns that point to completely different key lengths are just "noise" caused by such coincidences and should be ignored.

* Kasiski's second insight was that the length of the cipher key provided a direct lever for cracking a Vigenere cipher. Suppose, as just explained, Holmes determines that the cipher key is seven letters long. That means that every letter separated by seven letters is enciphered with the same cipher alphabet, and that the letters can be indexed into seven distinct sets:

SET 1: letters with index 1, 8, 15, 22, 29, ...
SET 2: letters with index 2, 9, 16, 23, 30, ...
SET 3: letters with index 3, 10, 17, 24, 31, ...
SET 4: letters with index 4, 11, 18, 25, 32, ...
SET 5 letters with index 5, 12, 19, 26, 33, ...
SET 6: letters with index 6, 13, 20, 27, 34, ...
SET 7: letters with index 7, 14, 21, 28, 35, ...

Holmes could slice the ciphertext into seven such sets of letters, and then crack each set using frequency analysis. Given that the standard Vigenere cipher was based on a Ceasar shift, Holmes could easily figure out the shift by observing a Pareto chart of the frequency distribution of the set and comparing it to a Pareto chart of the frequency distribution of average English text.

Another way of looking at this is to imagine that Holmes writes the ciphertext down in rows, with each row seven letters long. Once he does this, then each column of the message is enciphered using the same cipher alphabet, and can be cracked on a column-by-column basis with frequency analysis. Of course, he only gets a seventh of the plaintext message from each column, and so he won't be able to understand what the message actually says until he cracks all seven columns.

For example, suppose Alice decides to use a Vigenere cipher to encrypt, say, Lincoln's Gettysburg address, a relatively long document which begins as:

Fourscore and seven years ago our fathers brought forth on this continent ...

She uses the keyword WARTHOG to perform the encryption:

```
  WARTHOGWARTHOGWARTHOGWARTHOGWARTHOGWARTHOGWARTHOGWARTHO
...
    fourscoreandsevenyearsagoourfathersbroughtforthonthiscontinent
...
    BOLKZQUNERGKGKREEQLOXOAXHVIXBAKALFYXRFNNVZBOIMOCTPHZLJCTPIEXUH
...
```

Alice encrypts the rest of the document in the same way and sends it to Bob. Holmes intercepts the message. At first the ciphertext:

```
  BOLKZQUNERGKGKREEQLOXOAXHVIXBAKALFYXRFNNVZBOIMOCTPHZLJCTPIEXUH
...
```

-- looks like nothing but gibberish, but he pokes through the full ciphertext, finds a few repeating patterns, and factors out their intervals to determine that the cipher key used by Alice was seven letters long. Holmes then writes the ciphertext in rows, with seven letters per row, as follows:

```
   BOLKZQU
   NERGKGK
   REEQLOX
   OAXHVIX
   BAKALFY
   XRFNNVZ
   BOIMOCT
   PHZLJCT
   PIEXUH ...
```

He now breaks the message into seven separate columns:

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

The first column reads off as:

```
   COLUMN 1: BNROBXBPP ...
```

Holmes performs a frequency analysis on this string of text to get the matching plaintext letters:

```
   COLUMN 1: BNROBXBPP ...  ->  frvsfbftt ...
```

This seems clearly unhelpful in itself, but Holmes then repeats the exercise for the other six columns:

```
   COLUMN 2: OEEAAROHI ...  ->  oeeaarohi ...
   COLUMN 3: LREXKFIZE ...  ->  uangtorin ...
   COLUMN 4: KGQHANMLX ...  ->  rnyohutse ...
   COLUMN 5: ZKLVLNOJU ...  ->  sdeoeghcn ...
   COLUMN 6: QGOIFVCCH ...  ->  csaurhoot ...
   COLUMN 7: UKXXYZTT  ...  ->  oerrstnn  ...
```

Holmes takes these seven sets of plaintext letter matches and arranges them as columns, matching the ciphertext from which they were derived:

```
   1 2 3 4 5 6 7

   f o u r s c o
   r e a n d s e
   v e n y e a r
   s a g o o u r
   f a t h e r s
   b r o u g h t
   f o r t h o n
   t h i s c o n
   t i n e n t  ...
```

Now the solution just jumps out:

```
   foursco
   reandse
   venyear
   sagoour
   fathers
   brought
   forthon
   thiscon
   tinent ... ->

   fourscoreandsevenyearsagoourfathersbroughtforthonthiscontinent
...  ->

   Fourscore and seven years ago our fathers brought forth on
this
   continent ...
```

Of course, it is unrealistic to think that Holmes could perform all the frequency analyses perfectly and go directly to the solution, but even if he doesn't figure out exactly what the proper plaintext letter matches are on his first attempt, he can substitute his initial sets of matches back into the columns, perform some anagramming by inspecting the result to see where the text makes some sense and where it doesn't, and then start adjusting his matches until he converges on a solution.

In addition, Alice did some things that made life easier for Holmes. Her Vigenere cipher square used shift ciphers, which meant that if Holmes could just figure out a few letters in each set, he had the entire set. Furthermore, she used the keyword WARTHOG, and once Holmes had figured out about four of the cipher sets and plugged in their associated key letters, for example:

W???HOG

-- he might be able to guess that the keyword was WARTHOG and save himself some effort. Alice's message would have been more secure if her Vigenere square used mixed cipher alphabets, and if she had used a less obvious keyword. However, this approach would have still cracked her message; it just would have forced Holmes to do more work.

The Kasiski test has obvious limitations. It requires ciphertexts of reasonable length, and as the key grows longer, trying to find valid repetitive patterns in the text grows more difficult, and the size of the letter sets that can be sorted out of the ciphertext grows smaller and less penetrable by frequency analysis. If the key is the same length as the text, the Kasiski test is completely defeated.

* The Kasiski test had been independently discovered almost a decade earlier, in 1854, by the English inventor Charles Babbage (1792:1871), famed among the modern computer community as a foresighted genius who developed mechanical computers and invented some of the basic principles of computing machines. Babbage had begun cracking ciphers when he was a boy and eventually acquired a reputation for expertise in the field. In 1854, a Bristol dentist named John Hall Brock Thwaites proudly announced in scholarly journal that he had developed a new cipher that he intended to patent. It turned out to be just a restating of the Vigenere cipher, and Babbage replied to the journal that Thwaites was walking on well-traveled ground.

Thwaites, with skewed logic reminiscent of those found in Internet message-board quarrels, then challenged Babbage to crack his cipher. This had no bearing on whether the cipher was original or not, but Babbage was intrigued anyway, and came up with the solution. However, he never published it, and so cannot be properly credited as its inventor. Babbage was notorious for not completing or publishing his work, though it is possible that he kept the solution secret at the request of British government cryptanalysts, who wanted to make use of it while other governments still thought the Vigenere cipher was secure.

* There is another approach to cracking a Vigenere cipher known as "superimposition" that was devised by Auguste Kerckhoffs in his great text on cryptology, LA CRYPTOGRAPHIE MILITAIRE, published in 1883. Superimposition works very well no matter how long the key is, as long as the codebreaker has a substantial number of messages enciphered with the same key.

Suppose Holmes takes the first letter of all these messages. All of them have been enciphered with the same cipher alphabet, and so this collection of letters effectively defines a ciphertext encrypted by a monoalphabetic substitution

cipher. Given a large enough set of messages, the cipher alphabet for the first letter could be determined by simple frequency analysis.

The same procedure could be used on the second letter of all the messages, the third letter, and so on, working through the set of messages in "columns". If Holmes can determine that two different columns have been enciphered with the same letter, possibly by identifying a few high-frequency letters, he can then combine the two columns to help pick out lower-frequency letters.

For example, suppose Holmes gets a pile of twenty messages, all enciphered with the same key. The messages start out as follows:

```
MESSAGE 1:   DHKSJPO ...
MESSAGE 2:   RGMNKLJ ...
MESSAGE 3:   LKLNMDG ...
MESSAGE 4:   DHKBNED ...
...

MESSAGE 20:  JVDGYSV ...
```

He arranges the separate messages as if they were rows of a single message, and then splits the result into columns:

```
D H K S J P O ...
R G M N K L J ...
L K L N M D G ...
D H K B N E D ...
...
J V D G Y S V ...
```

Each of the columns amounts to a simple monoalphabetic substitution cipher:

```
 COLUMN 1:  DRLD ...     J
 COLUMN 2:  HGKH ...     V
 COLUMN 3:  KMLK ...     D
 COLUMN 4:  SNNB ...     G
 COLUMN 5:  JKMN ...     Y
 COLUMN 6:  PLDE ...     S
 COLUMN 7:  OJGD ...     V
...
```

-- which can be solved by frequency analysis, just as with the Kasiski test. Of course, also just as with the Kasiski test, each resulting set of plaintext letter matches is gibberish by itself, and only makes sense once they are all substituted back in by columns.

Now in this example Holmes has only 20 messages and so only 20 letters in each column, which is not really enough to do a detailed frequency analysis, but if Holmes is lucky, superimposition can also give hints on the length of the keyword. Suppose Holmes sorts out a set of columns from a set of messages as above, and performs frequency analysis on the columns. If he notices that

results from column 1 are similar to results from column 8, say having the same top four letters; the results from column 2 are similar to those from column 9; the results from column 3 are similar to those from column 10; and so on, then he has a very strong clue that the keyword is seven letters long.

Knowing this, he can merge the sets by columns -- merging column 1 with column 8 and 15 and 22 and so on; column 2 with column 9 and 16 and 23 and so on; continuing in this way until he merges column 7 with column 14 and 21 and 28 and so on -- in effect combining the Kasiski test and superimposition to multiply the number of letters for his frequency analyses. Of course, if the keyword is short then Holmes would have likely used the Kasiski test first and not bothered with superimposition, but this approach would be able to penetrate a long keyword whose length might not be evident using the Kasiski test.