2.2.13 (*) In the derivation of the formula for the minimum number of trials necessary to have probability greater than 1/2 that at least two outcomes are the same, did we give up anything of significance by using the simple estimate

$$\ln(1-x) < -x$$

valid for $0 < x < 1$?

2.2.14 (**) Suppose that two real numbers are chosen "at random" between 0 and 1. What is the probability that their sum is greater than 1? What is the probability that their product is greater than 1/2?

2.3 Statistics of English

What features do English sentences have that random streams of characters do not? How can this be described in purely mechanical terms? Specifically, how can we tell whether an alleged decryption is really a decryption, or is just gibberish?

In some circumstances, when trying to decrypt a message, we are done when what we have looks like English. And directions for further partial decryption are indicated by trying to imagine a sensible message into which known fragments would fit.

In such contexts, there is a strong presumption that the message *will not be gibberish*, so that a correct decryption will be immediately and easily recognizable. Especially in classical pre-computer cryptanalysis, study of the *content* of a 'partial plaintext' was a fundamental device. If the message were pure gibberish anyway, then the contents of a correct decryption would be indistinguishable from the contents of an incorrect decryption, and the process would be both futile and pointless.

Even in the context of a presumably coherent and 'meaningful' plaintext, there is the issue of *automation* of the process of distinguishing a plausible message from an implausible one. This amounts to giving a sufficiently precise description of features that English sentences have that random strings of characters do not. For the moment we will not attempt to describe the *meaning* that English sentences have, but only their *form*. Even that we only address in a very low-level statistical way.

So we will pick out some characteristics of English plaintext for two uses: First, these would be features that we would want to *obscure* in order to frustrate an adversary attempting to break *our* cryptosystem. Second, these would be features to *employ* in attempting to break an adversary's cryptosystem. In this section we will simply delineate some of these features, postponing both exploitation and masking of them to a later section.

The main device we'll use is the heuristic device of **frequency** of letters or words in 'typical samples' of English text. This starts from very crude observations such as that the letter 'e' seems to occur much more often than the letter 'z'. Pursuing this, if we pretend that English texts are sequences of characters selected

FROM "AN INTRODUCTION TO CRYPTOLOGY" BY P. GARRETT

according to some probabilities, then the '*probability*' f_e of a character being an 'e' should be approximately the ratio

$$\frac{\text{number of 'e's in that body of text}}{\text{total number of characters in large body of text}}$$

Similarly, the '*probability*' f_{the} that a *word* in an English text is 'the' is the ratio

$$\frac{\text{number of 'the's in large body of text}}{\text{total number of characters in that body of text}}$$

Thus, to pretend to 'compute' the probability that a character will be 'e', we count the number of 'e's in a large body of text and divide by the total number of characters. To 'compute' the probability that a word will be 'the', we count the number of 'the's in a large body of text and divide by the total number of words.

While this idea is *compatible* with the notion of **limiting frequency** in probability (introduced earlier), it is really just a heuristic, since the 'selection' of characters and words in English is not a 'random' process. Nevertheless, especially in pre-computer cryptanalysis this heuristic proved very useful. Specifically, it *does* become clear that some letters occur much more often than others, as do certain combinations of letters and short words. Thus, even from the viewpoint of this very simple probabilistic heuristic, English is very distinguishable from a 'random' sequence of letters. This fact is essential is ciphertext-only attacks on classical cryptosystems.

Some of the distinguishing features of English are as follows.

First are **small words**. In English, there are few very short words. Thus, if word boundaries are detectable in an encrypted text, quite sharp inferences can be made. The only single-letter words are *a* and *I*, out of 26 possibilities. Not counting abbreviations, in a 500-kilobyte sample of filtered email, only 35 two-letter words occur, out of $26 \times 26 = 676$ possibilities. And, in that sample, only 196 3-letter words appear, out of $26 \times 26 \times 26 = 17,576$ possibilities for combination of 3 letters.

Next are **common words**. In the 500-kilobyte sample used, more than 5000 distinct words appear. The 9 most common words already account for 21% of all words, the 20 most common account for 30%, the 104 most common give 50%, the most common 247 give 60%. A listing of common words and their frequency (as percentage) is given at the end of this section.

Blanks: In ordinary English, counting characters 'A' through 'Z' *and blanks* (but ignoring uppercase and lowercase) shows that blanks are by far the most common characters: about 17–18% of characters are blanks, while the next most numerous characters such as 'e', 't', 'o', 'a', and 'i' each occur less than 9% of the time. Thus, if blanks are treated as characters and encrypted, their high frequency may give away information. On the other hand, if they are *not* encrypted, then *other* information is given away more directly: a cryptanalyst can make use of knowledge about frequencies of small words and statistics about letter frequencies at word boundaries. For these reasons, in 'classical' cryptosystems the blanks often are removed from messages.

Character frequency. Counting characters 'A' through 'Z' but ignoring uppercase/lowercase distinctions in a megabyte of old email (after removing the headers), we find approximate frequencies (as percentages)

e	11.67	e	9.53	e	8.22	i	7.81	a	7.73	n	6.71	s	6.55
r	5.97	h	4.52	l	4.3	d	3.24	u	3.21	c	3.06	m	2.8
p	2.34	y	2.22	f	2.14	g	2.00	w	1.69	b	1.58	v	1.03
k	0.79	x	0.30	j	0.23	q	0.12	z	0.09				

That is, 11.67% of all characters are 'e', 9.53% of all characters are 'o', and so on, down to 0.09% of all characters being 'z'. In particular, the letter 'e' occurs more than 100 times as often as does 'z', and the other letters occur with frequencies in between these. Thus, there is a quite marked *statistical bias*, meaning that the 26 characters do not occur with equal frequency. Far from it.

Presenting this sort of information can be misleading, since even 500,000 characters of filtered email is not a very random sample of English. And, more important, one would need to know something about the likelihood of deviation from these frequencies. Such deviations would be very likely in small samples. And some people have in fact gone to the trouble of writing novels without using the specific letter 'e'.

And, without having done considerable testing, it is not at all clear how well the plausibility that a stream of characters is English is established by knowing that the frequencies of individual letters do or do not resemble these numbers. And, on the other hand, it is not clear at all that a stream of characters whose frequencies are *different* than these *can't* be English. A more serious and legitimate statistical study would take into account **sample size** and other variables.

Digrams are *adjacent pairs* of characters. The $26 \times 26 = 676$ different digrams which do or do not occur in a character stream tell much more about that stream of characters than do the single-character frequencies. Indeed, while every letter of the alphabet really does occur in many English words, the same is not true for digrams. We can also discuss digrams that include *blanks*, digrams that can occur at word boundaries, etc.

In the same sample of 500 kilobytes of email (with headers removed), *with spaces left in*, only 611 of the possible 676 digrams occur at all. (If blanks are removed, then 659 of the possible 676 digrams occur.) The top 44 digrams already give more than 50% of the total, the top 102 give 75%, the top 175 give 90%, and the top 279 give 98%. With blanks eliminated before counting, the frequencies are spread out a bit: the top 54 give 50%, the top 126 give 75%, the top 222 give 90%, and the top 359 give 98%. Tables of the most common digrams are given at the end of this section.

Trigrams are *adjacent triples* of characters. Out of $26 \times 26 \times 26 = 17,576$ possible trigrams, relatively few occur often. From the same megabyte of filtered email, looking only at the trigrams that occur *within English words*, the top 241 already give 50% of all trigrams occurring. This is quite extreme: fewer than 1/70 of all trigrams account for 50% of all occurrences! The top 652 give 75%, the top 1271 give 90%, and the top 2520 account for 98% of all trigrams that occur. If blanks are removed, then the frequencies are spread out, as with digrams: the top 430 give 50%, the top 1162 give 75%, the top 2314 give 90%, and the top 4408 give

98%. Tables of the most common trigrams are given at the end of this section.

There are several points here. First, even just looking at frequencies of occurrence of single characters, a stream of English text is not at all random: some letters persistently occur much more often than others. Further, some *adjacent pairs of characters* (digrams) occur much more often than others. Similarly for *adjacent triples of characters* (trigrams). This low-level statistical bias can be put to use in cryptanalysis, on one hand, and must be masked in order to have a secure cryptosystem, on the other hand.

As a simple special statistical feature, *blanks* occur about twice as frequently as any other character. Since messages are still fairly readable even after blanks have been removed, on many occasions blanks *are* removed before encryption. This removes some information which otherwise could be used by the cryptanalyst. For example, the statistics on digrams and trigrams show that the statistical bias is considerably sharper when word boundaries are clear, by contrast to the situation when word boundaries have been obliterated.

There still remains the issue of making systematic (and efficient) use of this information. Some simple illustrations will be given later.

The 100 most common words in the sample, with percentages of the total:

the 4.65	*to* 3.02	*of* 2.61	*i* 2.2	*a* 1.95
and 1.82	*is* 1.68	*that* 1.62	*in* 1.57	*it* 1.22
for 1.17	*you* 1.06	*be* 0.99	*not* 0.84	*on* 0.76
have 0.71	*this* 0.69	*as* 0.57	*at* 0.56	*would* 0.55
are 0.55	*but* 0.54	*if* 0.53	*my* 0.53	*with* 0.5
your 0.48	*so* 0.48	*or* 0.46	*some* 0.43	*will* 0.41
do 0.39	*about* 0.39	*me* 0.38	*from* 0.35	*by* 0.33
no 0.33	*more* 0.33	*what* 0.32	*an* 0.32	*there* 0.32
one 0.32	*all* 0.32	*was* 0.30	*we* 0.30	*just* 0.27
which 0.27	*can* 0.26	*very* 0.25	*series* 0.25	*am* 0.24
things 0.24	*people* 0.24	*get* 0.23	*hi* 0.23	*time* 0.22
think 0.22	*course* 0.22	*etc* 0.22	*also* 0.21	*any* 0.21
other 0.20	*than* 0.2	*know* 0.19	*could* 0.19	*they* 0.19
too 0.19	*only* 0.18	*up* 0.18	*good* 0.18	*out* 0.18
has 0.17	*such* 0.17	*had* 0.16	*should* 0.16	*now* 0.16
dont 0.16	*like* 0.15	*its* 0.15	*want* 0.15	*well* 0.15
here 0.14	*might* 0.14	*who* 0.14	*may* 0.14	*then* 0.14
make 0.14	*thanks* 0.14	*much* 0.13	*thing* 0.13	*did* 0.13
how 0.12	*really* 0.12	*he* 0.12	*students* 0.12	*maybe* 0.12
yours 0.12	*see* 0.12	*been* 0.12	*were* 0.12	*rather* 0.11
when 0.11	*paper* 0.11	*even* 0.11	*our* 0.11	*still* 0.11
case 0.11	*since* 0.11	*while* 0.11	*use* 0.1	*ill* 0.10
email 0.10	*stuff* 0.10	*seems* 0.10	*them* 0.10	*book* 0.10
work 0.10	*please* 0.10	*online* 0.10	*into* 0.10	*does* 0.10
two 0.10	*university* 0.09	*little* 0.09	*page* 0.09	*number* 0.09

Note that some words that have reached this 'high-frequency' list, such as 'university', 'number', and 'students', would very likely have a different frequency if the

text sample were not taken from the correspondence of a university professor. This bias occurs *after* the more obvious technical words and such things are filtered out. On one hand, we might want to have statistics that are more universal than this. On the other hand, any particular information about an adversary is potentially useful.

The top 77 digrams occurring *within words* in the sample, with percentages:

th 3.18	*in* 2.59	*he* 2.17	*er* 1.95	*re* 1.85	*on* 1.63	*an* 1.59
at 1.54	*ou* 1.43	*or* 1.26	*es* 1.26	*ha* 1.24	*to* 1.22	*te* 1.21
is 1.18	*ti* 1.17	*it* 1.16	*en* 1.13	*nt* 1.09	*ng* 1.08	*al* 1.07
se 1.05	*st* 1.01	*nd* 0.98	*le* 0.91	*ar* 0.90	*me* 0.90	*hi* 0.86
ve 0.85	*of* 0.84	*ed* 0.78	*co* 0.74	*as* 0.73	*ll* 0.72	*ne* 0.70
om 0.70	*ri* 0.68	*ic* 0.67	*ro* 0.67	*ea* 0.66	*et* 0.64	*ur* 0.64
io 0.64	*ra* 0.62	*li* 0.62	*no* 0.62	*so* 0.62	*be* 0.61	*de* 0.59
ma 0.59	*si* 0.58	*ly* 0.54	*ut* 0.53	*ot* 0.53	*pr* 0.53	*fo* 0.53
yo 0.52	*il* 0.50	*ca* 0.50	*pe* 0.50	*ch* 0.49	*ho* 0.49	*ul* 0.47
ce 0.47	*ta* 0.45	*di* 0.45	*rs* 0.45	*el* 0.44	*ge* 0.44	*us* 0.44
ec 0.42	*ss* 0.42	*ac* 0.41	*ct* 0.41	*em* 0.41	*wh* 0.40	*oo* 0.40

The most 77 frequent digrams *including blanks*:

*e_* 3.15	*_t* 2.55	*th* 2.11	*s_* 1.97	*t_* 1.93	*_a* 1.81	*in* 1.72
_i 1.69	*he* 1.44	*er* 1.29	*d_* 1.24	*re* 1.23	*_s* 1.18	*n_* 1.15
on 1.08	*an* 1.05	*_o* 1.04	*y_* 1.03	*at* 1.03	*r_* 0.99	*ou* 0.95
*o_* 0.92	*_w* 0.92	*or* 0.84	*es* 0.83	*ha* 0.83	*to* 0.81	*te* 0.80
is 0.79	*ti* 0.78	*it* 0.77	*en* 0.75	*nt* 0.72	*ng* 0.72	*_c* 0.71
al 0.71	*se* 0.70	*_m* 0.69	*_b* 0.67	*st* 0.67	*_p* 0.65	*nd* 0.65
_f 0.62	*le* 0.60	*ar* 0.60	*me* 0.60	*f_* 0.59	*l_* 0.59	*g_* 0.58
hi 0.57	*ve* 0.57	*_h* 0.56	*of* 0.55	*ed* 0.52	*_d* 0.51	*co* 0.49
as 0.48	*ll* 0.48	*ne* 0.47	*om* 0.46	*i_* 0.45	*ri* 0.45	*a_* 0.45
_n 0.44	*ic* 0.44	*ro* 0.44	*ea* 0.44	*et* 0.42	*ur* 0.42	*io* 0.42
_r 0.42	*_e* 0.41	*ra* 0.41	*li* 0.41	*no* 0.41	*so* 0.41	*be* 0.41

The top 77 digrams occurring *after blanks are removed*:

th 2.63	*in* 2.08	*he* 1.75	*er* 1.67	*re* 1.52	*on* 1.33	*es* 1.32
an 1.29	*at* 1.28	*ti* 1.26	*nt* 1.16	*ou* 1.16	*to* 1.13	*st* 1.12
ha 1.05	*or* 1.05	*et* 1.03	*en* 1.01	*te* 1.01	*is* 0.98	*it* 0.97
ea 0.93	*se* 0.90	*al* 0.89	*ng* 0.89	*nd* 0.81	*ed* 0.76	*hi* 0.75
le 0.75	*ar* 0.74	*si* 0.73	*me* 0.73	*so* 0.71	*of* 0.70	*ve* 0.68
ri 0.64	*as* 0.64	*om* 0.64	*ra* 0.61	*no* 0.61	*ne* 0.60	*co* 0.60
ro 0.59	*ll* 0.59	*ta* 0.58	*ic* 0.57	*ot* 0.57	*tt* 0.57	*li* 0.57
yo 0.52	*ur* 0.51	*ec* 0.51	*io* 0.51	*de* 0.51	*di* 0.51	*ma* 0.51
ei 0.49	*be* 0.49	*sa* 0.47	*ss* 0.47	*el* 0.46	*em* 0.46	*rs* 0.45
fo 0.44	*ut* 0.44	*ly* 0.44	*rt* 0.43	*ca* 0.42	*pr* 0.42	*na* 0.42
ts 0.41	*ho* 0.41	*il* 0.41	*pe* 0.40	*ch* 0.40	*ul* 0.38	*ee* 0.38

The 77 most common trigrams *within English words*, with percentages:

the 2.44	*ing* 1.26	*and* 0.82	*hat* 0.78	*tha* 0.77	*ion* 0.75	*you* 0.67
ent 0.66	*for* 0.63	*tio* 0.63	*thi* 0.60	*her* 0.51	*ati* 0.47	*our* 0.47
ere 0.45	*all* 0.43	*ter* 0.43	*ver* 0.40	*not* 0.40	*hin* 0.40	*ome* 0.36
oul 0.36	*uld* 0.36	*int* 0.34	*rea* 0.34	*pro* 0.34	*res* 0.33	*ate* 0.33
hav 0.30	*ave* 0.30	*ill* 0.30	*his* 0.30	*com* 0.30	*ons* 0.30	*are* 0.28
ple 0.28	*ers* 0.28	*con* 0.27	*ess* 0.27	*out* 0.27	*one* 0.26	*ith* 0.25
som 0.25	*ive* 0.25	*tin* 0.25	*nce* 0.24	*ble* 0.24	*ted* 0.24	*han* 0.23
ine 0.23	*per* 0.23	*ect* 0.23	*nte* 0.23	*wit* 0.22	*men* 0.22	*but* 0.22
wou 0.21	*ica* 0.21	*eve* 0.21	*cal* 0.21	*pre* 0.21	*cou* 0.21	*lin* 0.21
est 0.20	*eri* 0.20	*mor* 0.20	*ser* 0.20	*ore* 0.19	*any* 0.19	*abl* 0.19
tic 0.19	*urs* 0.19	*ant* 0.19	*sti* 0.18	*ear* 0.18	*hou* 0.18	*ies* 0.18

The 77 most common trigrams *including blanks*:

_th 1.67	*the* 1.22	*he_* 0.80	*ing* 0.63	*_to* 0.62	*to_* 0.55	*ng_* 0.52
_an 0.50	*_in* 0.49	*_of* 0.49	*at_* 0.45	*is_* 0.44	*of_* 0.44	*e_t* 0.43
*on_* 0.43	*er_* 0.42	*nd_* 0.42	*and* 0.41	*ed_* 0.40	*es_* 0.39	*hat* 0.39
tha 0.38	*ion* 0.37	*re_* 0.37	*_i_* 0.36	*_co* 0.35	*or_* 0.33	*t_t* 0.33
you 0.33	*e_a* 0.33	*ent* 0.33	*in_* 0.33	*_is* 0.32	*e_i* 0.32	*for* 0.31
_yo 0.31	*tio* 0.31	*_a_* 0.31	*thi* 0.30	*_be* 0.30	*ly_* 0.30	*s_a* 0.29
_re 0.29	*_no* 0.28	*nt_* 0.27	*t_i* 0.27	*_fo* 0.27	*_it* 0.27	*s_t* 0.26
_ha 0.26	*e_s* 0.26	*le_* 0.26	*_on* 0.25	*it_* 0.25	*her* 0.25	*ll_* 0.25
*me_* 0.25	*_so* 0.24	*n_t* 0.24	*_wh* 0.23	*ati* 0.23	*our* 0.23	*ve_* 0.23
_se 0.22	*s_i* 0.22	*ut_* 0.22	*ere* 0.22	*all* 0.21	*al_* 0.21	*ter* 0.21
*st_* 0.21	*d_t* 0.21	*_pr* 0.21	*se_* 0.20	*ver* 0.20	*not* 0.20	*_wi* 0.20

The 77 most common trigrams *after blanks have been removed*:

the 1.49	*ing* 0.77	*tha* 0.52	*and* 0.50	*hat* 0.47	*ion* 0.45	*ent* 0.43
you 0.41	*thi* 0.38	*for* 0.38	*ati* 0.38	*tio* 0.38	*her* 0.35	*ere* 0.35
eth 0.34	*int* 0.32	*our* 0.28	*tth* 0.27	*all* 0.27	*rea* 0.26	*ter* 0.26
nth 0.26	*ome* 0.25	*hin* 0.25	*ver* 0.25	*not* 0.24	*res* 0.23	*est* 0.22
oul 0.22	*ont* 0.22	*ate* 0.21	*uld* 0.21	*ers* 0.21	*tin* 0.21	*oth* 0.20
pro 0.20	*sth* 0.20	*ons* 0.20	*his* 0.19	*ith* 0.19	*ave* 0.19	*eri* 0.19
sin 0.19	*ess* 0.18	*are* 0.18	*hav* 0.18	*ist* 0.18	*ill* 0.18	*out* 0.18
com 0.18	*rth* 0.18	*ese* 0.17	*ore* 0.17	*ple* 0.17	*con* 0.17	*one* 0.16
att 0.16	*iti* 0.16	*ert* 0.16	*ica* 0.16	*ein* 0.16	*eto* 0.16	*som* 0.16
han 0.15	*oft* 0.15	*nte* 0.15	*ine* 0.15	*sto* 0.15	*ted* 0.15	*ive* 0.15
ear 0.15	*fth* 0.15	*nce* 0.15	*ret* 0.14	*ngt* 0.14	*ble* 0.14	*lin* 0.14

Exercises

2.3.01 Suppose that the probability that a given letter in English is 'e' really is 0.1167. What is the probability that in a "random text" of 10 words there will be *no* 'e'? In 100 characters?

2.3.02 Suppose that the probability that a given English word is 'the' really is 0.0465. What is the probability that in a "random text" of 10 words there will be *no* 'the'? In 100 words?

2.3.03 Suppose that there is a language which uses just two characters, '1' and '0'. Suppose that the '1' occurs with probability 2/3 and the '0' occurs with probability 1/3 in that language, in general. What is the probability that a stream of N 1's and 0's in that language could have $N/2$ or fewer 1's? Address this for $N = 3, 6, 9, 12$.

2.4 Attack on the Affine Cipher

With some information about *frequencies* of characters, words, digrams, etc., it is possible to give a more graceful ciphertext-only attack on an affine cipher than the brute-force attack of trying all the possible keys.

For example, suppose we are given the ciphertext

JFFGJFDMGFSJHYQHTAGHQGAFDCCFP

Our goal is to find the key (a, b) so that (with notation as earlier in discussion of the affine cipher)

$$E_{a,b}(\text{plaintext}) = JFFGJFDMGFSJHYQHTAGHQGAFDCCFP$$

Since the spaces have been removed, we cannot make direct use of small-word frequencies. Nevertheless, looking at single-letter percentages, we have

F	20.68
G	13.79
H	10.34
J	10.34
Q	6.89
A	6.89
C	6.89
D	6.89
P	3.44
S	3.44
T	3.44
Y	3.44
M	3.44

This would cause us to think that the encryption of the letter 'e' is 'F', since 'e' is by far the most common letter in English and 'F' is by far the most common letter in the ciphertext. Then, just hoping for luck, the second-most common letter in English is 't', and (by a good margin) the second-most common letter in this ciphertext is 'G', so we might guess that 't' is encrypted as 'G'. To assess the quality of this guess, we must determine the key (a, b) so that

$$E_{a,b}(e) = F \qquad E_{a,b}(t) = G$$

Numerically, since 'e' is encoded as 4, 'F' is 5, 't' is 19, and 'G' is 6. Thus, in terms of the numbers,

$$(a \cdot 4 + b) \ \% \ 26 = 5$$
$$(a \cdot 19 + b) \ \% \ 26 = 6$$

As mentioned in the discussion of the *chosen-plaintext* on the affine cipher, this gives (by subtracting)

$$a \cdot (19 - 4) \ \% \ 26 = 6 - 5 = 1$$

That is,

$$15 \cdot a \ \% \ 26 = 1$$

We find (by brute force or otherwise) that the multiplicative inverse of 15 mod 26 is 7, so $a = 7$ is our guess. Then going back to the equation

$$(a \cdot 4 + b) \ \% \ 26 = 5$$

with the tentative $a = 7$ we get

$$(7 \cdot 4 + b) \ \% \ 26 = 5$$

This gives $2 + b = 5$, so (we guess) $b = 3$.

That is, based on *frequencies*, we guess that the key is $(7, 3)$. If so, by our general formula for the inverse

$$E_{a,b}^{-1} = E_{a^{-1}, -a^{-1}b}$$

where the a^{-1} denotes inverse modulo 26. We have already noted that $15 \cdot 7 \ \% \ 26 = 1$, so 15 is the multiplicative inverse of 7. We compute

$$-7^{-1} \cdot 3 \ \% \ 26 = -15 \cdot 3 \ \% \ 26$$
$$= -45 \ \% \ 26 = 7$$

Thus, the inverse of $E_{7,3}$ is $E_{15,7}$. Applying $E_{15,7}$ to the ciphertext should *decrypt* and recover the plaintext:

$$E_{15,7}(JFFGJFDMGFSJHYQHTAGHQGAFDCCFP)$$
$$= meetmeaftermidnightinthealley$$

which is readily broken up into *meet me after midnight in the alley.* We were lucky that our first guess was correct.

Note that in this ciphertext-only attack we used frequency analysis to bring us closer to what amounted to a known-plaintext attack. If we had been less lucky, we would have had to guess again, and go through the determination of the alleged key, then decrypt to see if what came out looked reasonable.

Exercises

2.4.01 Decrypt the affine cipher with ciphertext 'VCLLCP BKLC LJKX XCHCP'

2.4.02 Decrypt the affine cipher with ciphertext 'LBBKL BJMKB OLTQW TXIKT WKIBJ AABN'

2.4.03 (*) Decrypt 'DBUHU SPANO SMPUS STMIU SBAKN OSMPU SS'