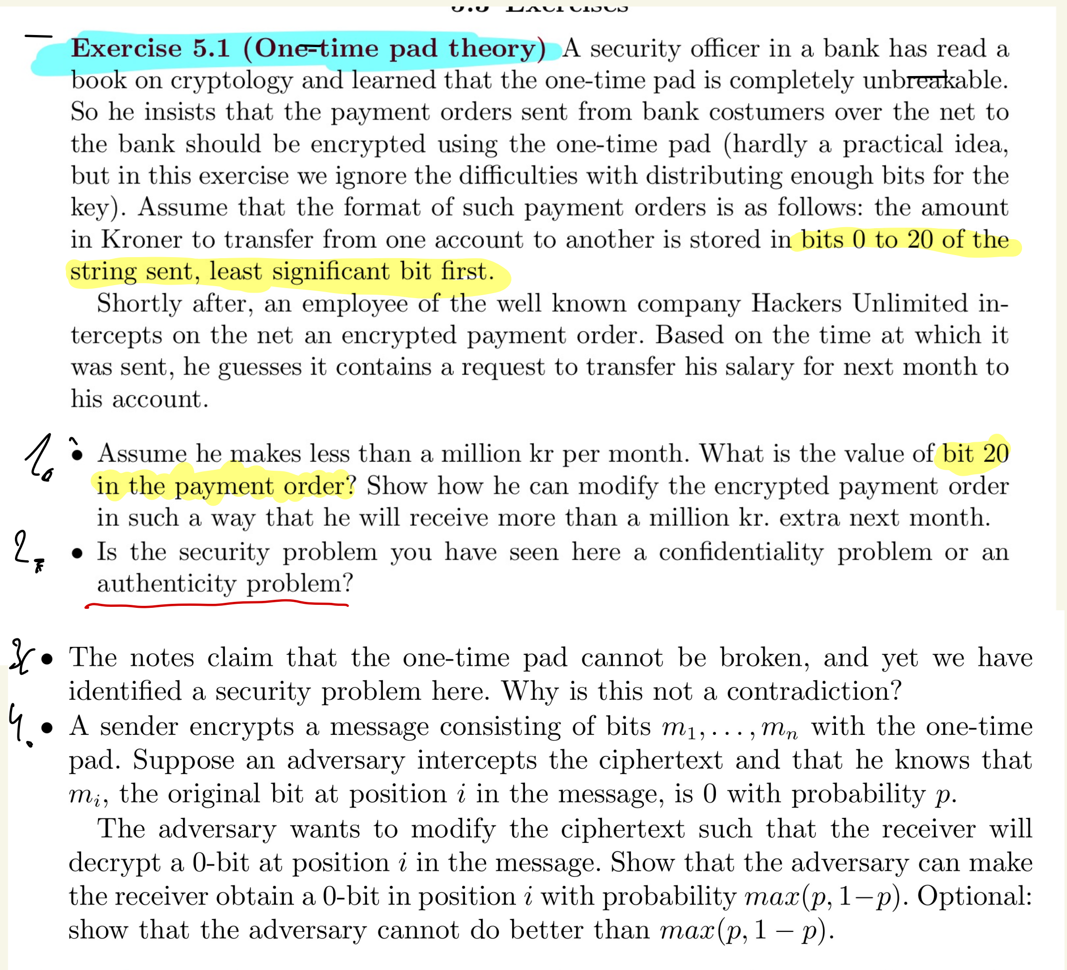


Exercise 5.1 (One-time pad theory) A security officer in a bank has read a book on cryptology and learned that the one-time pad is completely unbreakable. So he insists that the payment orders sent from bank costumers over the net to the bank should be encrypted using the one-time pad (hardly a practical idea, but in this exercise we ignore the difficulties with distributing good bits for the key). Assume that the format of such payment orders is as follows: the amount in Kroner to transfer from one account to another is stored in bits 0 to 20 of the string sent, least significant bit first.

Shortly after, an employee of the well known company Hackers Unlimited intercepts on the net an encrypted payment order. Based on the time at which it was sent, he guesses it contains a request to transfer his salary for next month to his account.

1. - Assume he makes less than a million kr per month. What is the value of bit 20 in the payment order? Show how he can modify the encrypted payment order in such a way that he will receive more than a million kr. extra next month.
2. - Is the security problem you have seen here a confidentiality problem or an authenticity problem?
3. - The notes claim that the one-time pad cannot be broken, and yet we have identified a security problem here. Why is this not a contradiction?
4. - A sender encrypts a message consisting of bits $m_1, \ldots, m_n$ with the one-time pad. Suppose an adversary intercepts the ciphertext and that he knows that m_i, the original bit at position i in the message, is 0 with probability p.

The adversary wants to modify the ciphertext such that the receiver will decrypt a 0-bit at position i in the message. Show that the adversary can make the receiver obtain a 0-bit in position i with probability $max(p, 1-p)$. Optional: show that the adversary cannot do better than $max(p, 1-p)$.

Handin 3

The most ↓ | the least

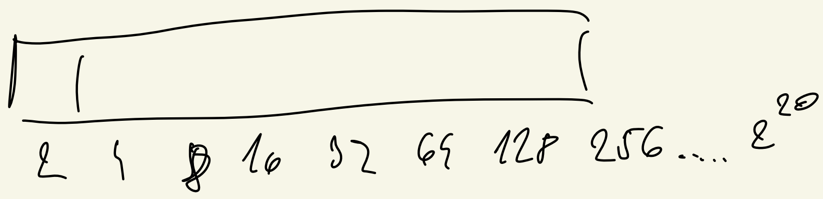
$20 = 16+4$

16 8 4 2

2^4 ~~2^3~~ 2^2

1 0 1 0

16	8	4	2	2	4	8	16
1	0	1	0	0	1	0	1



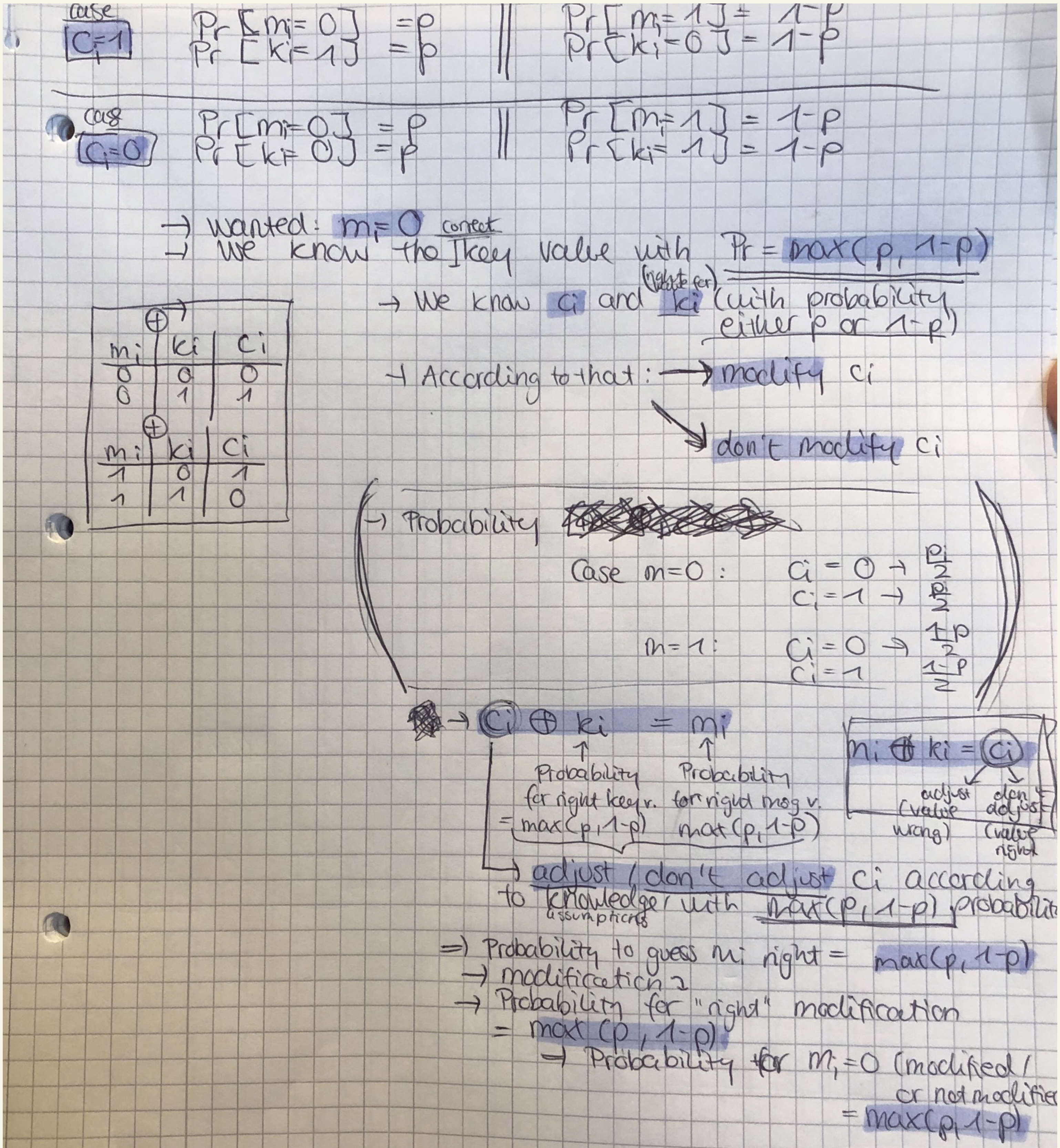
Intercepted → and makes < 1000 000
↳ last bit is then zero, in original number

1. earns < 1000 000, original plain text on 20th position is 0
↳ it doesn't matter what the pad is, it is enough to flip the bit on position 20 to the other value

2. Authenticity / Data integrity

3. As this is a problem of authenticity the cipher itself was not broken → just the ciphertext was tempered with.

4.



case $\boxed{c_i=1}$ $Pr[m_i=0] = p$ || $Pr[m_i=1] = 1-p$
$Pr[k_i=1] = p$ || $Pr[k_i=0] = 1-p$

case $\boxed{c_i=0}$ $Pr[m_i=0] = p$ || $Pr[m_i=1] = 1-p$
$Pr[k_i=0] = p$ || $Pr[k_i=1] = 1-p$

→ wanted: $m_i = 0$ correct
→ We know the key value with $\underline{Pr = max(p, 1-p)}$
→ We know c_i and (right value for) k_i (with probability either p or 1-p)
→ According to that: → modify c_i
→ don't modify c_i

m_i ⊕	k_i	c_i
0	0	0
0	1	1

m_i ⊕	k_i	c_i
1	0	1
1	1	0

→ Probability

Case $m=0$: $c_i = 0 \rightarrow \frac{p}{2}$
$c_i = 1 \rightarrow \frac{p}{2}$

$m=1$: $c_i = 0 \rightarrow \frac{1-p}{2}$
$c_i = 1 \quad \frac{1-p}{2}$

→ $c_i \oplus k_i = m_i$

k_i ↑ Probability for right key v. $= max(p, 1-p)$

m_i ↑ Probability for right msg v. $max(p, 1-p)$

$m_i \oplus k_i = c_i$ → adjust (value wrong) / don't adjust (value right)

→ adjust / don't adjust c_i according to knowledge / assumptions with $max(p, 1-p)$ probability

⇒ Probability to guess m_i right $= max(p, 1-p)$
→ modification?
→ Probability for "right" modification $= max(p, 1-p)$
→ Probability for $m_i = 0$ (modified / or not modified) $= max(p, 1-p)$