Tentamen i

KRYPTOTEKNIK

den 15 december 1997 klockan 14^{00}–19^{00}.

Institutionen för informationsteknologi
Lunds universitet

- Tillåtna hjälpmedel: Formelblad, räknedosa.
- Skriv namn och årskurs överst på varje papper.
- Varje lösning skall skrivas på separat papper.
- Lösningarna måste tydligt dokumentera tillvägagångssättet.
- Lösningarna skall följa de metoder som kursen innehåller.
- Varje lösningssteg skall motiveras.
- Uppgifterna är inte numrerade efter svårighetsgrad.
- Vi kommer att anslå tentamensresultatet på vår `www`-sida. Endast namn och betyg på de godkända anges. Om du inte vill att ditt namn publiceras på detta sätt vill vi att du skriftligen meddelar institutionen detta.

Lycka till!

Problem 1:

A simple cryptosystem is constructed as follows. The plaintext $M = (M_0\, M_1\, \cdots\, M_6)$ is considered as an element in $\mathbb{F}_{2^7}$, $M_i \in \mathbb{F}_2$, such that

$$M = M_0 + M_1\alpha + M_2\alpha^2 + \cdots + M_6\alpha^6,$$

where $\pi(\alpha) = 0$ and $\pi(x) = x^7 + x^3 + 1$. The key $K \in \mathbb{F}_{2^7}$ is similarly written

$$K = K_0 + K_1\alpha + \cdots + K_6\alpha^6,$$

and chosen such that $K \neq 0$. Encryption is done by multiplying in $\mathbb{F}_{2^7}$, i.e.,

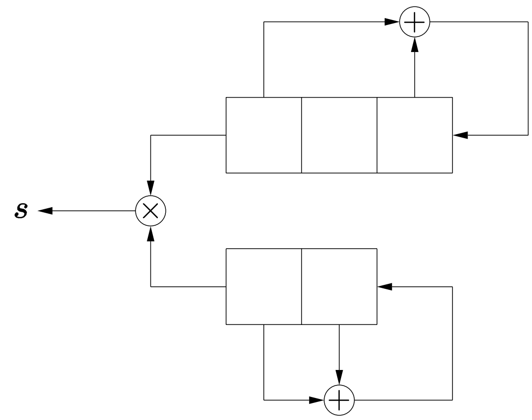
$$C = K \cdot M.$$

a) Decrypt the ciphertext $C = (1000001)$ using the key $K = (1110000)$.

b) Explain why the key must be chosen such that $K \neq 0$.

(10 points)

Problem 2:



Consider the binary sequence $\boldsymbol{s}$ generated in the above figure, where $\otimes$ denotes ordinary multiplication in $\mathbb{F}_2$. The initial states are the all one states, i.e., $< 111 >$ and $< 11 >$, respectively.

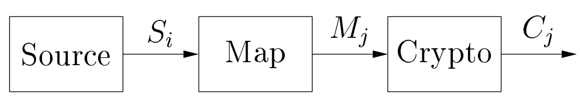
Find the shortest LFSR that generates the sequence $\boldsymbol{s}$. Motivate your answer!

(10 points)

Problem 3:

In this problem you will compare an RSA-system with a substitution cipher.

The cryptosystems should be used to encode a source S according to the figure below.



The source generates uniformly distributed and independent random symbols from an alphabet of size 30. The map-function takes 3 source symbols and maps them to a numerical value in the range $[0, 46917]$. Some values will never appear in the ciphertext.

The RSA-system has public parameters $(n, e) = (46918, 20107)$. The substitution cipher has as many different keys as there are possible values of e when $n = 46918$.

a) Calculate $H(S_i)$ and $H(M_j)$.

b) Find the unicity distance ("entydighetslängden") for the substitution cipher.

c) Find the unicity distance for the RSA-system.

d) Explain why the answer in **c)** is less than in **b)**, and what makes RSA (in general) safe.

e) You observe $C_j = 10164$ in the RSA-system. Find the message M_j.

Hint: $\phi(23458) = 11376$.

(10 points)

Problem 4:

a) Construct a cryptosystem for authentication with three different plaintexts, i.e., $|\mathcal{M}| = 3$, such that

$$P_I \leq 1/3, \text{ and } P_S \leq 1/3.$$

(This will require $|\mathcal{C}| \geq 9$ and $|\mathcal{K}| \geq 9$.)

b) Construct a secret sharing scheme over $\mathbb{F}_5$ for the access structure

$$\Gamma_0 = \{\{P_1, P_2, P_3\}, \{P_1, P_2, P_4\}, \{P_3, P_4\}, \{P_5\}\}.$$

You will be rewarded according to the following. Any secret sharing scheme for Γ_0 gives 1 point. If the scheme is *perfect*, you receive 3 points and if the scheme is *ideal*, you receive 5 points.

(10 points)

Problem 5:

This problem is about "Toy-DES", a DES-like block cipher. You find a description of Toy-DES in an appendix.

a) From a friend you get the encrypted message 0010 1110. Your key is 1000 0000. Find the message that your friend is sending to you.

b) You have found out the following pairs of plaintext and ciphertext from Toy-DES with unknown key:

Plaintext	Ciphertext
0000 0000	0011 0010
0010 0000	0010 1110

Find the input x-or and output x-or for the S-box that will help you in finding the key.

c) Find all possible keys obtained from the input x-or and output x-or in **b)**.

d) Find the key. The parity bit is 0.

For the S-Box we have for some input/output x-or:

Input x-or	Output x-or	Possible S-Box inputs
110011	0011	$\{000100, 110111\}$
110011	0010	$\{000110, 001000, 010011, 100000, 110101, 111011\}$

(10 points)

Description of "Toy-DES": "Toy-DES" has an 8-bit input x_0, an 8-bit output, and an 8-bit key K, where the last bit is a parity bit (odd parity). The block cipher has 3 rounds as follows.

1. The plaintext x_0 is divided in two parts, $x_0 = L_0R_0$, where L_0 is the first 4 (leftmost) bits and R_0 is the 4 last (rightmost) bits.

2. A certain function with start value x_0 is iterated 3 times. If $x_i = L_iR_i$, we compute L_iR_i according to the following iteration:

$$\begin{aligned} L_i &= R_{i-1}, \\ R_i &= L_{i-1} \oplus f(R_{i-1}, K_i), \end{aligned}$$

   where $\oplus$ denotes bitwise addition of the two bitstrings.

3. Finally, the ciphertext is (R_3L_3). Note the reversed order of L_3 and R_3.

We now describe the function f. If we write $f(R_x, K_x)$, then R_x is of length 4 and K_x is of length 6. The function $f(R_x, K_x)$ returns a bitstring of length 4, which is obtained by executing the following steps:

1. R_x is expanded to a bitstring of length 6 using a fixed expansion function E.

2. Compute $B = E(R_x) \oplus K_x$.

3. The next step uses an *S-box* S, which is a fixed 4×16 array whose entries are from the integers $0 - 15$. Given a 6-bit string $B = b_1b_2b_3b_4b_5b_6$, we compute $S(B)$ as follows. The two bits b_1b_6 determine the binary representation of a row r of S, $0 \le r \le 3$, and the four bits $b_2b_3b_4b_5$ determine the binary representation of a column c of S, $0 \le c \le 15$. Then $S(B)$ is defined to be the entry in row r and column c, written in a binary representation as a 4-bit string. In this fashion, we compute $C = S(B)$.

4. The bitstring C obtained from the previous step is defined to be $f(R_x, K_x)$.

The expansion function E is specified by the following table:

$$\mathrm{E} = \begin{bmatrix} 1 & 2 & 3 & 4 & 1 & 2 \end{bmatrix}$$

The S-box is as follows:

$$S = \begin{bmatrix} 14 & 4 & 13 & 1 & 2 & 15 & 11 & 8 & 3 & 10 & 6 & 12 & 5 & 9 & 0 & 7 \\ 0 & 15 & 7 & 4 & 14 & 2 & 13 & 1 & 10 & 6 & 12 & 11 & 9 & 5 & 3 & 8 \\ 4 & 1 & 14 & 8 & 13 & 6 & 2 & 11 & 15 & 12 & 9 & 7 & 3 & 10 & 5 & 0 \\ 15 & 12 & 8 & 2 & 4 & 9 & 1 & 7 & 5 & 11 & 3 & 14 & 10 & 0 & 6 & 13 \end{bmatrix}$$

The partial keys K_i are obtained from K by selecting the following bits:

$$K_1 = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \end{bmatrix}, \quad K_2 = \begin{bmatrix} 2 & 3 & 4 & 5 & 6 & 7 \end{bmatrix}, \quad K_3 = \begin{bmatrix} 3 & 4 & 5 & 6 & 7 & 1 \end{bmatrix}$$

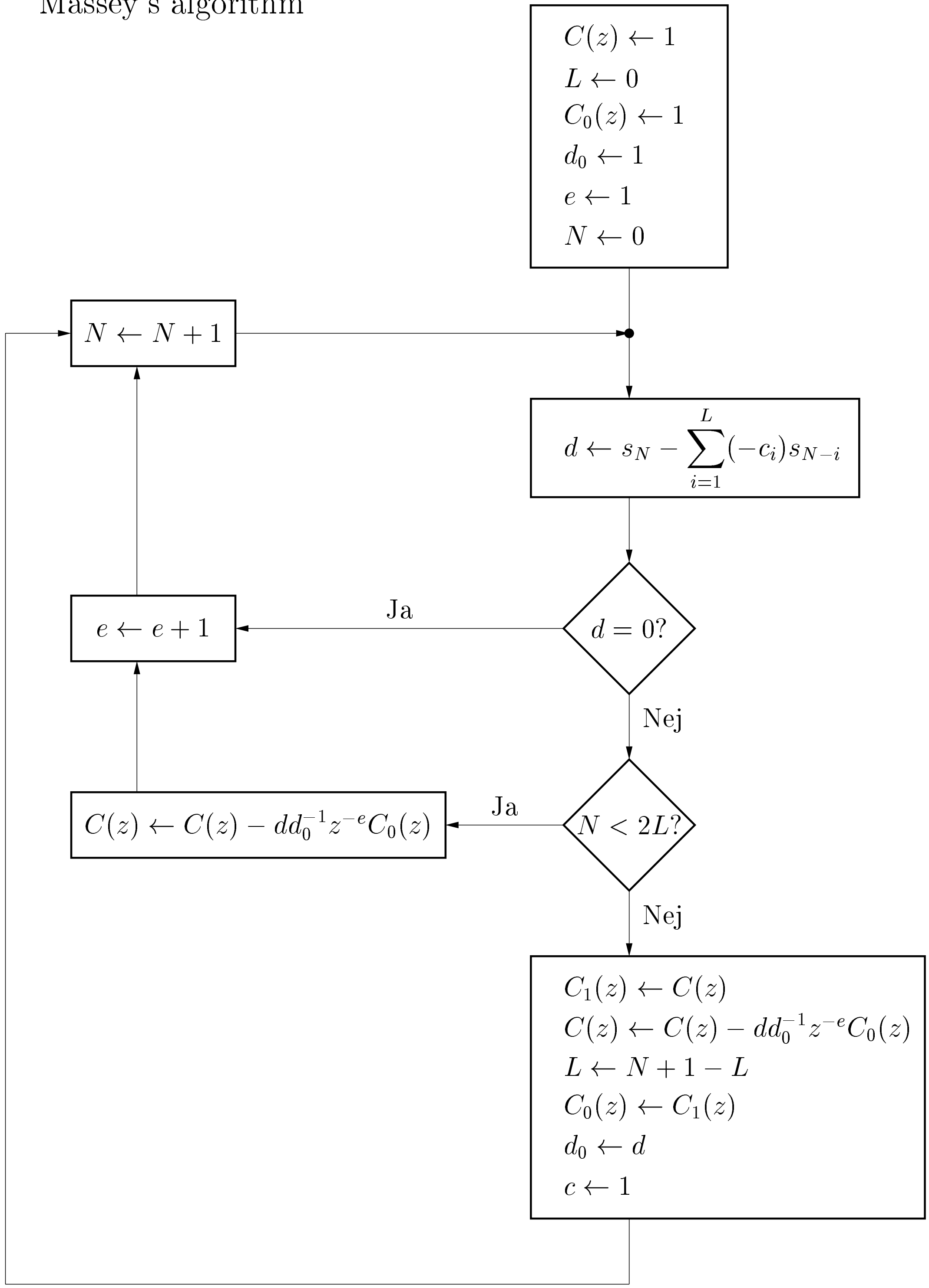
Note: Compared with original DES, the IP and P permutations have been removed.

i	$E_i = E(R_{i-1})$	K_i	$B_i = E_i \oplus K_i$	$C_i = S(B_i)$	L_i	R_i
0	-	-	-	-		
1						
2						
3						
Codeword:						

i	$E_i = E(R_{i-1})$	K_i	$B_i = E_i \oplus K_i$	$C_i = S(B_i)$	L_i	R_i
0	-	-	-	-		
1						
2						
3						
Codeword:						

i	$E_i = E(R_{i-1})$	K_i	$B_i = E_i \oplus K_i$	$C_i = S(B_i)$	L_i	R_i
0	-	-	-	-		
1						
2						
3						
Codeword:						

Massey's algorithm



s_N	d	$C_1(z)$	$C(z)$	L	Skiftregister	$C_0(z)$	d_0	e	N
$-$	$-$	$-$	1	0	$\longleftarrow$	1	1	1	0