

# KÓDOLÁSTECHNIKA PZH

2006. december 18.

1. Hibajavító kódolást tekintünk. Egy lineáris bináris blokk kód generátormátrixa

$$G = \begin{bmatrix} 10110 \\ 01101 \end{bmatrix}$$

- Adja meg a kód kódszavait és paramétereit (n, k, d). (3 p)
- Perfekt-e a kód? (4 p)
- Adja meg a szisztematikus paritásellenőrző mátrixát. (2 p)
- Adja meg a szindróma dekódolási táblázatát. (4 p)
- Adja meg a dekódolás lépéseit a táblázat használatával. Dekódolja az 11111 vett szót. (3p)
- Adjon felső becslést a dekódolási hibavalószínűséget emlékezetnélküli BSC(p) esetre!  
(3 p, pontos érték 5 p)

2. Definiálja a következőket:

- minimális távolságú dekódolás (1p)
- perfekt tulajdonságú hibajavító kód (2 p)
- Reed Solomon kód generátorpolinomja (magyarázza el a jelöléseket) (3 p)
- szorzatkód, a kétdimenziós bináris paritáskód paramétere (3 p)

3. RSA kódoló algoritmus esetén a kulcsok előállításához  $p=7$ ,  $q=17$  prímekből indultunk ki.

- Adja meg a lehető legkisebb kódoló kulcsként használható exponenst! (2p)
- Számítsa ki az  $x=11$  nyílt szöveghez tartozó  $y$  rejtett szöveget! (2 p)
- Határozza meg a dekódoló kulcsot! (3p)
- Mi az ismételt négyzetre emelés és szorzás módszere? (2p)
- Mennyi moduláris szorzás és négyzetemelés szükséges esetünkben. (2p)

4. Tömören, formálisan fejtse ki :

- ütközés-ellenálló hash függvény (2 p)
- kulcsstanúsítvány lánc (kis példán elmagyarázva) (3 p)
- üzenet integritásvédelem célja (CBC-MAC példáján elmagyarázva) (3 p)

5. a.) Legyen  $X$  valószínűségi kimenetelei  $(x_1, x_2, x_3, x_4, x_5)$ , eloszlása  $(1/3, 1/4, 1/6, 1/6, 1/12)$ .  
Konstruáljon Huffman kódot ehhez az eloszláshoz. (5 p)

b.) Adja meg a kódszóhossz várható értékét (2p)

6. Tömören, formálisan fejtse ki:

- Prefix kód fogalma. Adjon 4 kódszóból álló prefix kódot. (2 p)
- Optimális prefix kódok tulajdonságai (3 p)
- DPCM kódolás (blokkéma a jelölések magyarázatával) (3 p)

**Pontozás: 1: <=21 2:22- 33 3: 32 - 41 4: 42- 51 5: 52– 62**

# Kódolástechnika PZH eredmények

2006. december 18.

(Ügyeljen a pontos fogalomhasználatra, pontos, formális definíciókra, részletes indoklásokra!)

1.

a.) (3p) kódszavak:

$n=$  ,  $k=$  ,  $d=$

b.) (4p) igen nem indoklás:

c.) (2p)  $H =$

d.) (4p) szindróma dekódolási táblázat (T):

e.) (3p) dekódolás lépései.

11111 dekódolása:

f.) (3/5p) Pe

2. a. b. c. d. (karikázza be, amire válaszolt)

3.

a.) (2p)  $e=$

b.) (2p)  $y=$

c.) (3p)  $d=$

d.) (2p) módszer:

e.) (3p) szorzás db: négyzetremelés db: indoklás:

4. a. b. c. (karikázza be, amire válaszolt)

5. (8p) a.) kódszavak: ( $x_1:$   $x_2:$   $x_3:$   $x_4:$   $x_5:$  )  
(2p) várható kódszóhossz= $=$

6. a. b. c. (karikázza be, amire válaszolt)

Név: .....

Neptun kód:

.....

# Kódolástechnika PZH megoldások

2006. december 18.

1.

a.) (3p) kódszavak: 00000, 10110, 01101, 11011  $n=5$ ,  $k=2$ ,  $d=3$

b.) (4p) igen nem indoklás:  $1+5 < 2^3=8$

11100

c.) (2p)  $H=$  10010

01001

d.) (4p) szindróma dekódolási táblázat (T):

**szindróma**    **hibavektor**

000            00000

001            00001

011            00011

100            00100

101            01000

110            10000

111            10001

e.) (3p) dekódolás lépései.  $s=Hv^T$ ,  $e'=T(s)$ ,  $c'=v-e'$

11111 dekódolása:  $s^T=(100)$ ,  $00100=T(100)$ ,  $11011=11111-00100$

f.) (3/5p)  $P_e < 1 - ((1-p)^5 + 5p(1-p)^4)$ ,  $P_e = 1 - ((1-p)^5 + 5p(1-p)^4 + 2p^2(1-p)^3)$

**3.**  $p=7$ ,  $q=17$

$m = p \cdot q = 7 \cdot 17 = 119$

$\phi(m) = (p-1) \cdot (q-1) = 6 \cdot 16 = 96$

a.) Az exponenssel kapcsolatos feltétel, hogy relatív prím legyen  $\phi(m)$ -hez. A legkisebb ilyen szám a 5. Tehát  $e = 5$ .

b.)  $y = x^e \pmod{m}$

$11^5 = 161051 = 44 \pmod{119}$

Tehát  $y = 44$ .

c.) A dekódoló kulcs a kódoló kulcs multiplikatív inverze a modulo  $\phi(m)$

$96 = 19 \cdot 5 + 1$ ,  $d = -19 = 77 \pmod{96}$

Tehát  $d = 77$ .

d.)  $x = y^{77} \pmod{119}$ ,  $77 = 2^6 + 2^3 + 2^2 + 1$ , 3 szorzás, 6 négyzetremelés (összesen 9 szorzás)

**5.** (8p) két megoldás

a.) kódszavak: (11, 10, 01, 001, 000), (11, 01, 00, 101, 100),

(2p) várható kódszóhossz =  $27/12 = 2.25$  bit

2. Definiálja a következőket:

**a.) minimális távolságú dekódolás (1p)**

A vett szót a Hamming távolságra legközelebbi kódszóra dekódoljuk.

**b.) perfekt tulajdonságú hibajavító kód (2 p)**

Abban az esetben, ha a gömbi dekódolási tartományokkal hézagmentesen le tudjuk fedni a szavak terét, a kódról beszélünk.

Bináris egy hibát javító  $C(n,k,3)$  perfekt kód esetén:  $1+n=2^{n-k}$ .

Bináris  $t$  hibát javító perfekt kód esetén:  $1+n+\frac{n}{2}+\dots+\frac{n}{t}=2^{n-k}$ .

**c.) Reed Solomon kód generátorpolinomja (magyarázza el a jelöléseket) (3 p)**

Tekintsünk egy  $GF(q)$  véges testet, s egy  $n$ -edrendű  $\alpha$  elemet.  $C(n, k, d=n-k+1)$  Reed Solomon kód generátorpolinomja:

**2.20. tétel.** A Reed-Solomon-kódok esetén legyen az  $n$  kódszóhossz egyenlő az ott szereplő  $\alpha$  elem  $m$  rendjével. Ekkor a kód ciklikus, és generátorpolinomja

$$g(x) = \prod_{i=1}^{n-k} (x - \alpha^i),$$

**d.) szorzatkód, a kétdimenziós bináris paritáskód paraméterei**

$$\begin{bmatrix} c_0^{(0)} & c_1^{(0)} & \dots & c_{k_1-1}^{(0)} & c_{k_1}^{(0)} & \dots & c_{n_1-1}^{(0)} \\ c_0^{(1)} & c_1^{(1)} & \dots & c_{k_1-1}^{(1)} & c_{k_1}^{(1)} & \dots & c_{n_1-1}^{(1)} \\ \vdots & \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ c_0^{(k_2-1)} & c_1^{(k_2-1)} & \dots & c_{k_1-1}^{(k_2-1)} & c_{k_1}^{(k_2-1)} & \dots & c_{n_1-1}^{(k_2-1)} \\ \vdots & \vdots & \dots & \vdots & \vdots & \dots & \vdots \\ c_0^{(n_2-1)} & c_1^{(n_2-1)} & \dots & c_{k_1-1}^{(n_2-1)} & c_{k_1}^{(n_2-1)} & \dots & c_{n_1-1}^{(n_2-1)} \end{bmatrix}$$

2.3. ábra. A szorzat-kódszó képzése

Egy  $C_1(n_1, k_1, d_1)$  és egy  $C_2(n_2, k_2, d_2)$  lineáris kód (komponens kódok) felhasználásával  $C_1 \times C_2(n_1 \cdot n_2, k_1 \cdot k_2, d_1 \cdot d_2)$  szorzatkódot készíthetünk, amelynek kódszavai  $n_1 \times n_2$  dimenziós mátrixok, ahol a mátrix sorai  $C_1$  kódbeli, oszlopai  $C_2$  kódbeli kódszavak. Szisztematikus komponens kódok esetén a szorzatkódbeli mátrix-kódszó bal felső  $k_1 \times k_2$  dimenziós minorja tartalmazza az üzenetet. A mátrix-kódszavakat soronként kiolvastva kapjuk a szorzatkód — soros — kódszavát. A kapott  $C_1 \times C_2(n_1 \cdot n_2, k_1 \cdot k_2)$  kód lineáris.

kétdimenziós bináris paritáskód paraméterei:  $n=(k_1+1)*(k_2+1)$ ,  $k=k_1*k_2$ ,  $d=4$

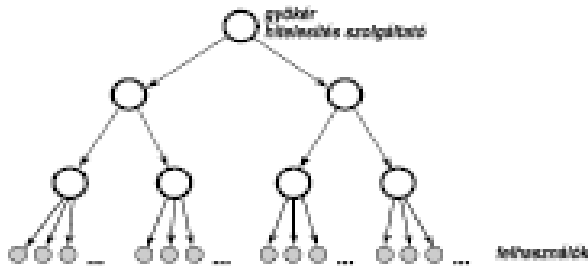
4.) Tömören, formálisan fejtse ki :

**a.) ütközés-ellenálló hash függvény (2 p)**

Nehéz két olyan elemet olyan ösképtérbeli elemet találni, amelynek hash értéke azonos. Iterációs hash függvények esetén, ha az iterációs (kompressziós) függvény ütközésellenálló és MD padding-et alkalmazunk, a hash függvény is ütközésellenálló lesz.

Pl. Digitális aláírás csalást hajthatna végre az aláíró, ha nem lenne ütközésellenálló a hash függvény.

**b.) kulcsánúsítvány lánc (kis példán elmagyarázva) (3 p)**



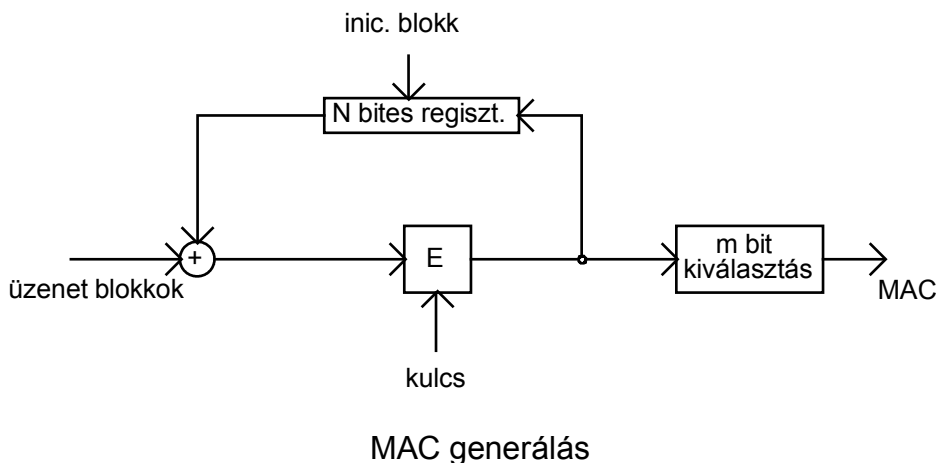
8.1.. ábra. Hitelesítés szolgáltatók tiszta hierarchikus szerveződésének illusztrációja. Minden nyíl egy kibocsátott tanúsítványt reprezentál, ahol a nyíl iránya a kibocsátás irányára utal.

Csakúgy mint az egyetlen hitelesítés szolgáltatót tartalmazó rendszerben, itt is feltesszük, hogy minden felhasználó ismeri a gyökér szolgáltató hiteles nyilvános kulcsát. Ez azonban még nem elég egy adott felhasználó tanúsítványának ellenőrzéséhez, hiszen a felhasználók tanúsítványait nem a gyökér szolgáltató írja alá. Amire szükség van az egy tanúsítvány lánc, mely a következő tulajdonságokkal rendelkezik:

- A lánc első eleme egy olyan tanúsítvány, amit a gyökér szolgáltató adott ki, és így a benne található nyilvános kulcs hitelessége beírási által ellenőrizhető.
- A lánc minden további tanúsítványára igaz, hogy ellenőrizhető a láncban öt közvetlenül megelőző tanúsítványban található nyilvános kulccsal.
- A lánc utolsó tanúsítványa tartalmazza a kérdéses, hitelesíteni kívánt felhasználói nyilvános kulcsot.

Könnyen látható, hogy egy ilyen tanúsítvány lánc birtokában és a gyökér szolgáltató nyilvános kulcsának ismeretében bármely felhasználó bármely másik felhasználó tanúsítványát ellenőrizni tudja. Ehhez a lánc tanúsítványait kell sorrendben ellenőrizni.

**c.) üzenet integritásvédelem célja (CBC-MAC példáján elmagyarázva) (3 p)**  
 $[x_1, \dots, x_n, \text{MAC}(x_1, \dots, x_n)]$



Formálisan  $\text{MAC} = g(y_n)$ , ahol  $y_n$  az  $y_i = E_k(x_i \oplus y_{i-1})$ ,  $i=1, \dots, n$  rekurzió  $n$ -edik lépésbeli eredménye, továbbá  $y_0 = I$  (inicializáló. blokk). A  $g$  függvény  $y_n$   $N$  bitjéből valamely  $m$  bitet választja ki, s az eredmény a kriptográfiai ellenőrző összeg.

6. Tömören, formálisan fejtse ki:

**a.) Prefix kód fogalma. Adjon 4 kódszóból álló prefix kódot. (2 p)**

4.2. definíció. Az  $f$  kód prefix, ha a lehetséges kódszavak közül egyik sem folytatása a másiknak, vagyis bármely kódszó végéből bármekkora szegmenst levágva nem kapunk egy másik kódszót.

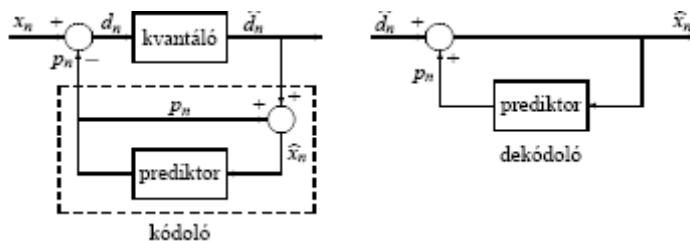
Pl.  $\{01, 10, 11, 001\}$

### b.) Optimális prefix kódok tulajdonságai (3 p)

4.4. tétel. Ha az  $f: X \rightarrow \{0, 1\}^*$  prefix kód optimális, és  $X$  elemei úgy vannak indexelve, hogy  $p(x_1) \geq p(x_2) \geq \dots \geq p(x_{n-1}) \geq p(x_n) > 0$ , akkor feltehető, hogy  $f$ -re a következő három tulajdonság teljesül:

- $|f(x_1)| \leq |f(x_2)| \leq \dots \leq |f(x_{n-1})| \leq |f(x_n)|$ , vagyis nagyobb valószínűségekhez kisebb kódszóhosszak tartoznak.
- $|f(x_{n-1})| = |f(x_n)|$ , vagyis a két legkisebb valószínűségű forrásbetűhöz tartozó kódszó egyenlő hosszú.
- Az  $f(x_{n-1})$  és az  $f(x_n)$  kódszavak csak az utolsó bitben különböznek.

### c.) DPCM kódolás (blokséma a jelölések magyarázatával) (3 p)



5.9. ábra. DPCM

Ezt felhasználva az előző lépéseink így módosulnak:

$$\begin{aligned}
 d_1 &= x_1 - x_0 \\
 \hat{d}_1 &= Q(d_1) = d_1 + q_1 \\
 \hat{x}_1 &= \hat{x}_0 + \hat{d}_1 = x_0 + d_1 + q_1 = x_1 + q_1 \\
 d_2 &= x_2 - \hat{x}_1 \\
 \hat{d}_2 &= Q(d_2) = d_2 + q_2 \\
 \hat{x}_2 &= \hat{x}_1 + \hat{d}_2 = \hat{x}_1 + d_2 + q_2 = x_2 + q_2.
 \end{aligned}$$

Általánosítva:

$$\hat{x}_n = x_n + q_n,$$

tehát a kvantálási hiba nem akkumulálódik.