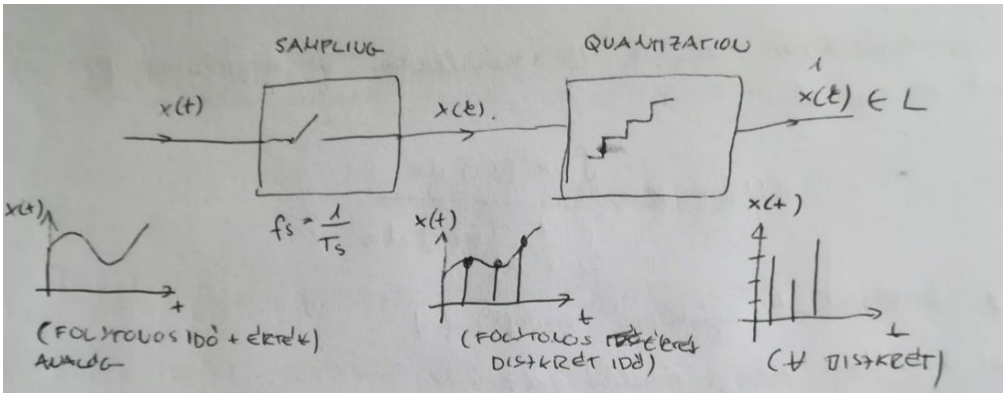
Infokód TL;DR

1. Describe the discrete memoryless source model (sampling, optimal Lloyd Max quantization ...etc.).

- Analóg jel digitálissá alakítása
  - mintavételezés: adott időközönként / mintavételi frekvencia -> diszkrét idő
  - kvantálás: felvett értékeket lekerekítjük / leképezzük diszkrétre -> diszkrét midnen



Mintavételezés

- Memóriamentes = üzenetek függetlenek
- Sávszélesség: x(t) előáll így Fourier trafóból

$$x(t) = \int_{-B}^{B} X(f) e^{j2\pi ft} df$$

- veszteségmentes
- Mintavételi tétel: $f_s \geq 2B$
- pillanat ~ rövid idő átlaga
- $f_s = 2B$ nem jártható -> ideális szűrő

Kvantálás

- Jel zaj viszony (kvantálás minősége)

$$\mathrm{SNR} = \frac{P_{\mathrm{jel}}}{P_{\mathrm{zaj}}} = \left(\frac{A_{\mathrm{jel}}}{A_{\mathrm{zaj}}}\right)^2, \qquad \mathrm{SNR}^{\mathrm{dB}} = 10\log_{10}\left(\frac{P_{\mathrm{jel}}}{P_{\mathrm{zaj}}}\right) = 20\log_{10}\left(\frac{A_{\mathrm{jel}}}{A_{\mathrm{zaj}}}\right)$$

Ekvidisztáns kvantáló

- Egyenlő részeken az intervallum
- csak teljesen kivezérelt jelre

$$SQNR = \tfrac{3}{2}2^{2n} \quad \Delta = x_{k+1} - x_k = \tfrac{2C}{N} \forall k = 1 \ldots N \ .$$

Logaritmikus kvantáló

- beszéd különböző SNR ekvidiszt.
- kvantálási szintek eloszlása logaritmikus

$$\mathrm{SNR} = K' \frac{\int_{-C}^{C} x^2 p(x)\,\mathrm{d}x}{\int_{-C}^{C} \frac{1}{c'^2(x)} p(x)\,\mathrm{d}x}$$

Differenciális kvantáló

- Előző minta tartalma alapján
  - különbség
  - vagy valószínűség
- (valszeg nem memoryless)

Lloyd-Max algoritmus

- Optimális kvantálóhoz algoritmus
- Nagyobb valségű helyeken kisebben lesznek a kvantálási intervallumok
- Nem globálisan optimális
- SNR-t szeretnénk maximalizálni
- Van egy hűségkritérium

$$J(\Delta, Q) := \sum_{i=1}^{N} \int_{\Delta_i} (x - q_i)^2 p(x)\ \mathrm{d}x$$

- rekurzív

2. Derive the Nyquist criterion for ISI free communication over band limited channels.

Szimbólum közti interferencia

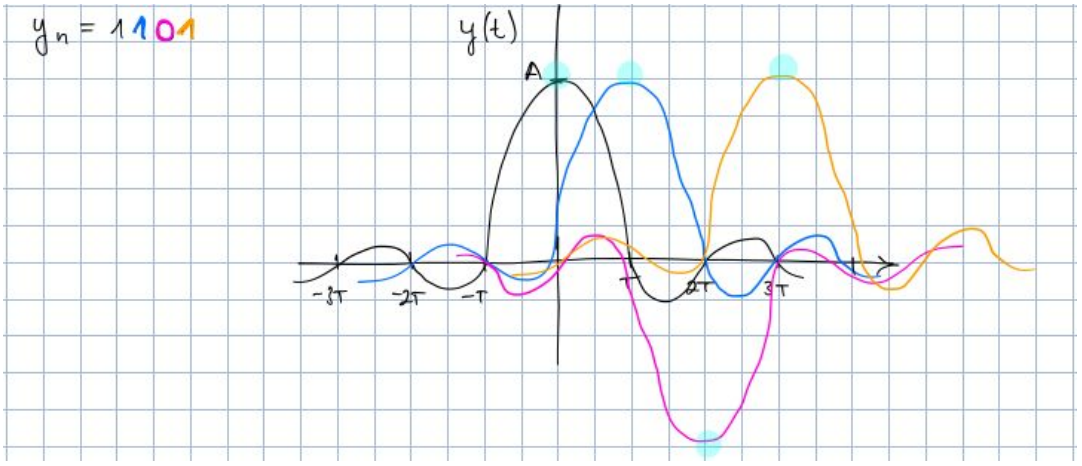
- négyzetes jel időtartományban szinuszos jel frekvenciában
- végtelen sávszélesség
- minden impulzus átlapolódik

Nyquist

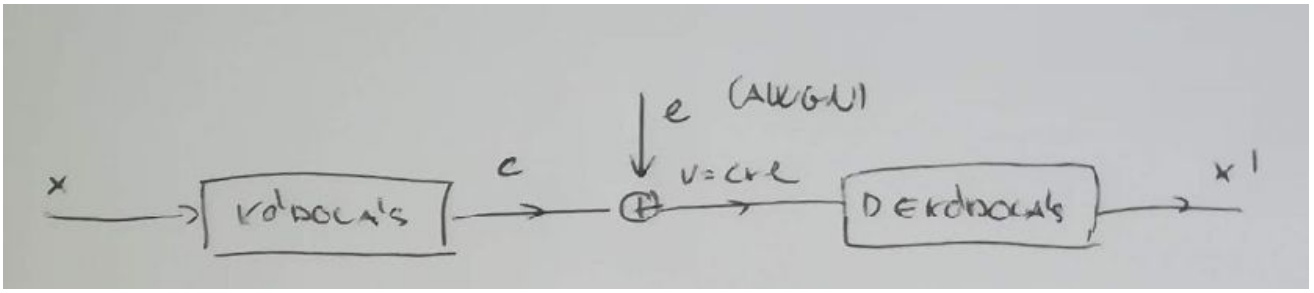
- Szimbólumok átlapolódnak, de 0 az ISI, ha mindenki 0 a másik impulzus közepén

$$\frac{1}{T} \sum_{k=-\infty}^{+\infty} H(f - \frac{k}{T}) = 1,$$

- Legjellemzőbb impulzus: emelt koszinusz



3. Describe the memoryless channel model (AWGN channel and BSC), derive the bit error probability as a function of the signal-to-noise ratio.



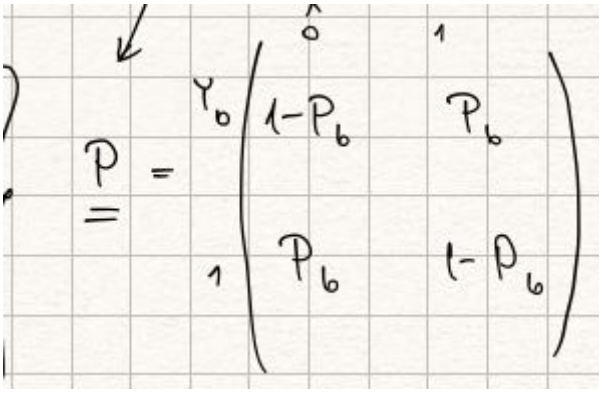
- memóriamentes: csak önmagától függ egy bit

BSC: Binary Symmetric Channel

- bináris adatok
- annak valsége, hogy egy rossz csak a bittől függ
- Jellemzi: bithibaarány

$$P_b = P(v = 1|c = 0) = P(v = 0|c = 1).$$

- állapot átmeneti mátrix



AWGN: Additive White Gaussian Noise

- A beérkező jel az elküldött és egy normál eloszlású random zaj összege.
- Szórásnégyzete jellemzi

$$SNR_{[\mathrm{dB}]} = 10 \cdot \log\left(\frac{P_{signal}}{\sigma^2}\right).$$

- Bithibaarány az SNR függvényében:
  - A bithibaarány definíciójából és az AWGN SNR-jéből vezettük le

$$P_b = \Phi\left(-0.5 \cdot \sqrt{\frac{10^{\frac{1}{10}SNR_{[\mathrm{dB}]}}}{P_{signal}}}\right)$$

4. Define and describe the properties of entropy, joint entropy, conditional entropy and mutual information.

Entrópia

- Bizonytalanság mértéke (nulla pl a biztosan bekövetkező konstans valség változónak, maximum egy egyenletes eloszlásúnak
- Tömörítés elméleti határa
- Szimbólum információja:

$$\mathrm{I}(x) = \mathrm{ld}\,\frac{1}{p(x)}$$

- Entrópia az üzenetek átlagos információja:

$$\mathrm{H}(X) = \mathbb{E}(\mathrm{I}(x)) = \sum_{x \in X} p(x) \cdot \mathrm{I}(x) = \sum_{x \in X} p(x)\,\mathrm{ld}\,\frac{1}{p(x)}$$

- Az entrópia tulajdonságai: (N az üzenet hossza)
  1. $0 \leq H(X) \leq ld\ N$
  2. $\forall X\ \forall g(X) \quad H(g(X)) \leq H(X)$

Együttes entrópia

- Két forrás együttes entrópiája:

$$\mathrm{H}(X,Y) = \sum_x \sum_y p(x,y)\,\mathrm{ld}\,\frac{1}{p(x,y)}$$

- Tulajdonságok:

1. $H(X,Y) \geq \max(H(X), H(Y))$
2. $H(X_1, \ldots, X_n) \geq \max(H(X_1), \ldots, H(X_n))$
3. $H(X,Y) \leq H(X) + H(Y) \longrightarrow H(X,Y) = H(X) + H(Y) \Leftrightarrow$ X és Y független
4. $H(X_1, \ldots, X_n) \leq H(X_1) + \ldots + H(X_n)$

Feltételes entrópia

- Mennyi a bizonytalanság marad átlagosan X-re nézve, ha előtte megfigyeltük Y-t.

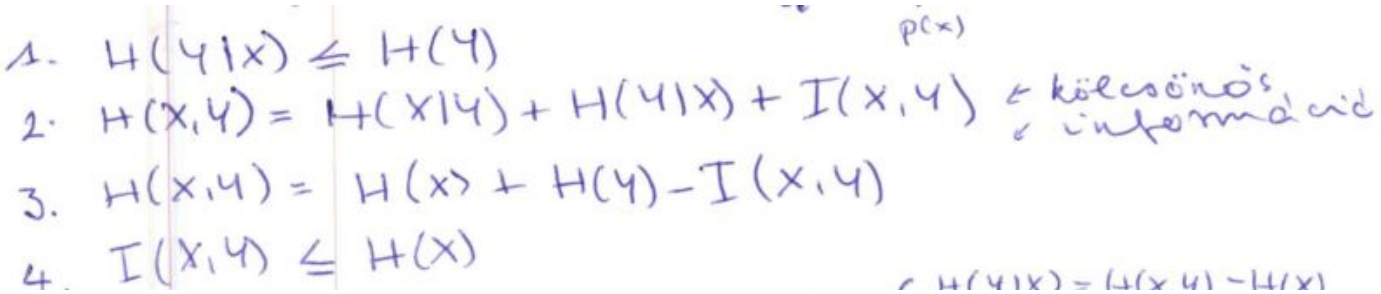
$$\mathrm{H}(X|Y) = \sum_x \sum_y p(x,y)\,\mathrm{ld}\,\frac{1}{p(x|y)}$$

- Az együttes entrópia ebből:

$$\mathrm{H}(X,Y) = \mathrm{H}(X) + \mathrm{H}(Y|X) = \mathrm{H}(Y) + \mathrm{H}(X|Y)$$

(független esetben a két entrópia összege)

- Tulajdonságok:



Kölcsönös információ

- Kullback-Leibeler távolsága két valségi eloszlásnak

$$\mathbb{D}(p(x)||q(x)) = \sum_x p(x) \operatorname{ld} \frac{p(x)}{q(x)}$$

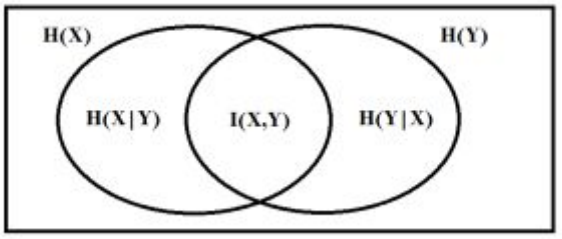
- Kölcsönös információ az együttes valség és a kül valség szorzatainak kullback leibeler távolsága.

$$\mathrm{I}(X,Y) = \mathbb{D}(p(x,y)||p(x)\cdot(y))$$

- Tulajdonságok:
  - 1. Nagyobb egyenlő mint 0 (akkor nulla, ha X, Y függetlenek)
  - 2. Kommutatív
- Még:

$$\mathrm{I}(X,Y) = \mathrm{H}(X) - \mathrm{H}(X|Y) = \mathrm{H}(Y) - \mathrm{H}(Y|X)$$

- Erről kell megjegyezni mindent:



5. Define the typical set of an IT source (AEP) and derive its properties.

- Asymptotic Equipartition Propertyp
- Nagy számok törvénye

$$\lim_{n\to\infty} \frac{1}{n} \operatorname{ld} p(x_1,...,x_n) = -\mathrm{H}(X)$$

- x-ek valségi változók, a cucc a sorozat valsége
- $p(x_1, \ldots, x_n)$ valség közel lesz az 2^{-nH} értékhez
- Ez alapján az összes szekvenciát szét osztjuk:
  - tipikus halmaz (közel az entrópiához)
  - nem tipikus halmaz (minden mást)
- majdnem minden esemény majdnem egyformán meglepő
- Tipikus halmaz formális def:

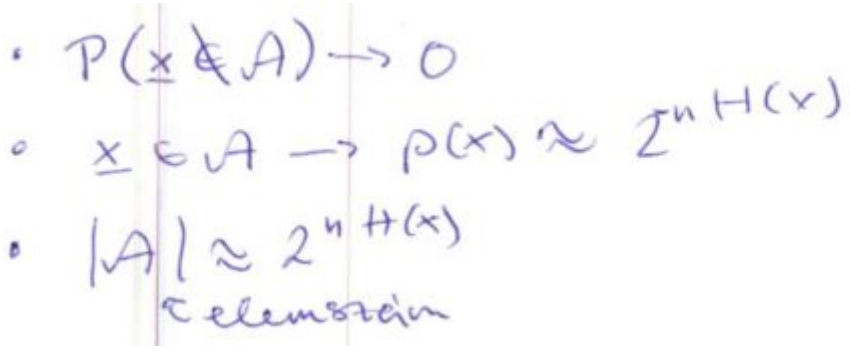
$$A = \{\mathbf{x} = (x_1, x_2, ..., x_n) \mid 2^{-n\,\mathrm{H}(X)-\varepsilon} \leq p(\mathbf{x}) \leq 2^{-n\,\mathrm{H}(X)+\varepsilon}\}$$

- valség, hogy egy random vektor a halmazban van:

$$(1-\varepsilon) \leq P(\mathbf{x} \in A) \leq 1$$

- Tipikus halmaz mérete

$$(1-\varepsilon) \cdot 2^{n\,\mathrm{H}(X)-\varepsilon} \leq |A| \leq 2^{n\,\mathrm{H}(X)+\varepsilon}$$

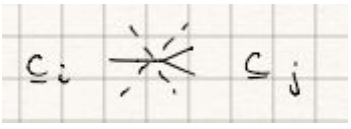


6. Define the properties of uniquely decodable codes, and discuss the source coding theorem.

- **Forráskódolás**: adattömörítés (forrásábécé, kódábécé, reprezentálás rövid, viasszafejthető)
- Az $f : X \to Y^*$ **egyértelműen dekódolható**, ha minden kódsorozat csak egy üzenetet kódol, pontosabban, ha u, v üzenetekre

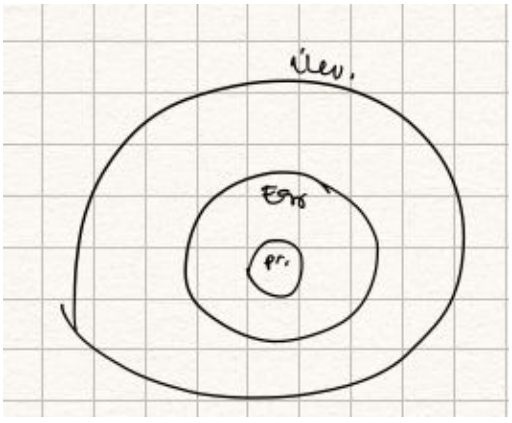
  $$f(u_1)f(u_2)..f(u_k) = f(v_1)f(v_2)..f(v_m)$$

  akkor és csak akkor, ha $u = v$.
- **Prefix kód**: egyik kódszó sem folytatása a másiknak (prefix ~ egyértelműen dek.)

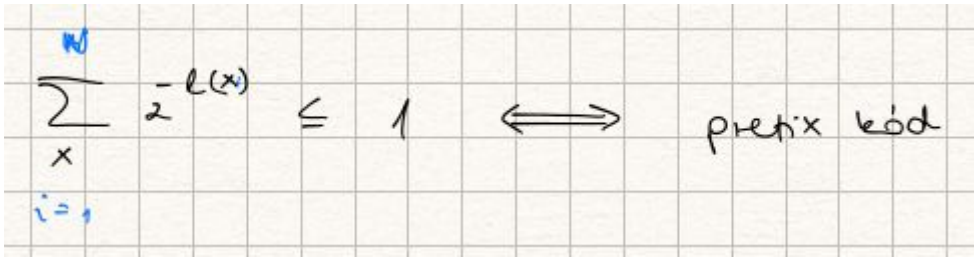


- **Invertálható kód**:

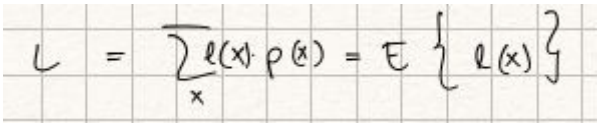
$$\mathbf{x_1 \neq x_2 \rightarrow C(x_1) \neq C(x_2)}$$



- **Kraft egyenlőtlenség** (l(x) a kódszó hossza)



- Az átlagos kódhossz:



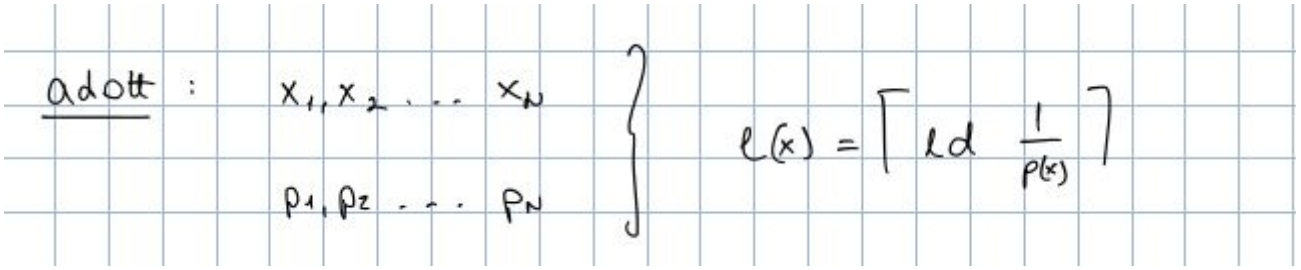
Forráskódolás alaptétele

- Adatömörítés alsó határát határozza meg.

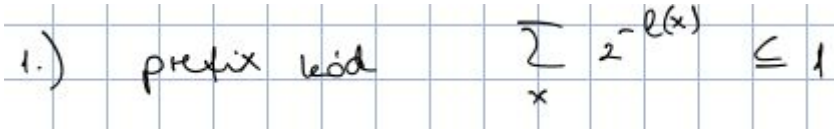
$$\mathrm{H}(X) \leq L = \sum_{x} p(x) \cdot l(x)$$

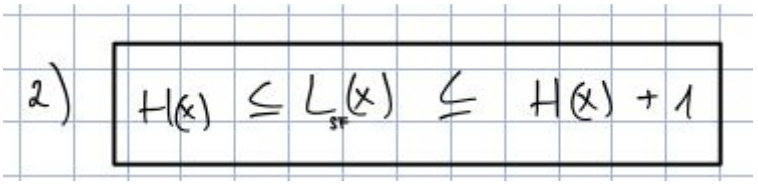
7. Describe the Shannon-Fano, Huffman, and arithmetic coding and discuss their performance.

Shannon-Fano

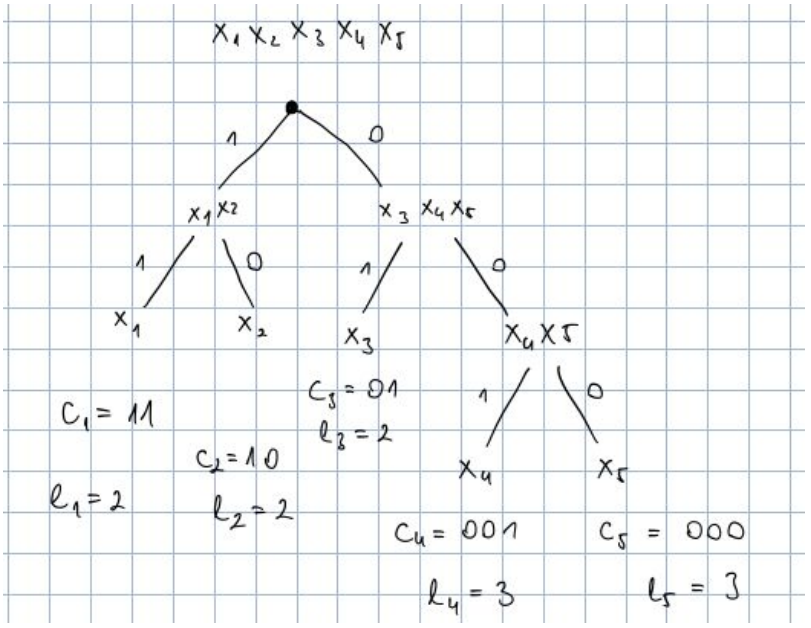


- Tulajdonságok:



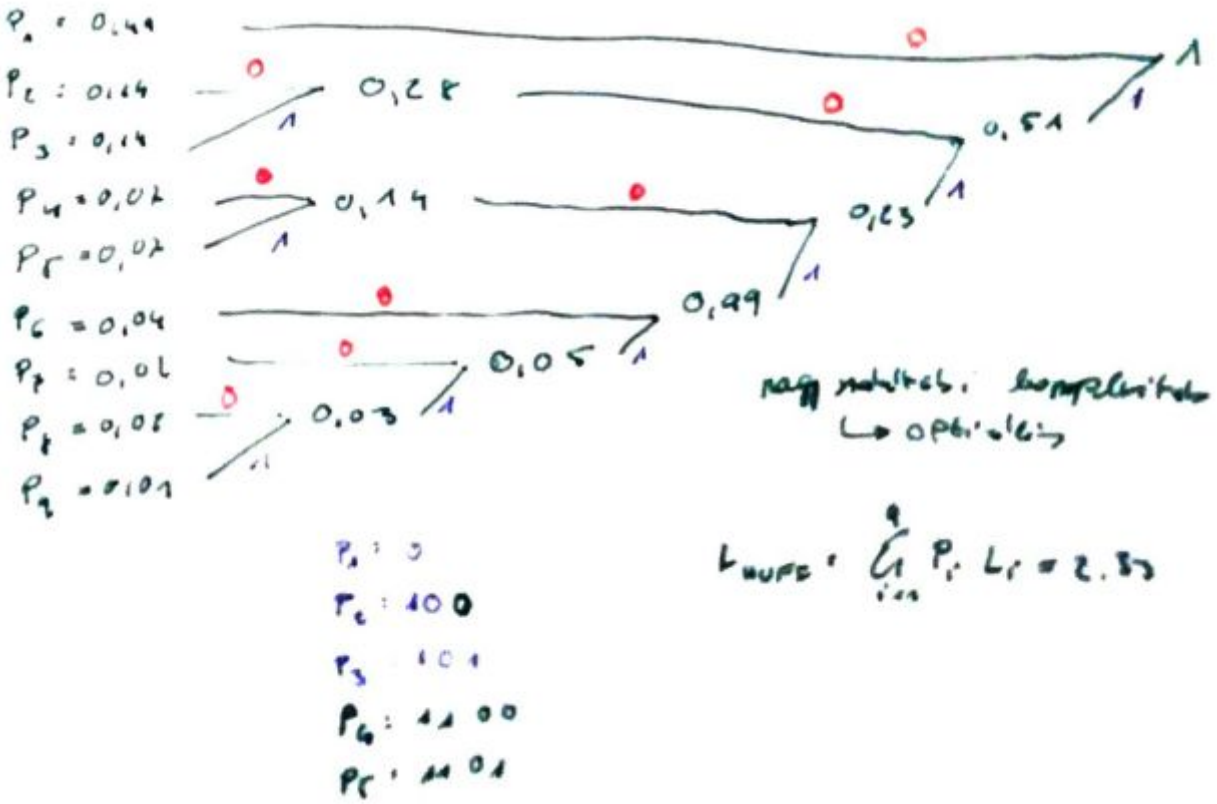


- Algoritmus



Huffman

- Optimális kód
- Opt. kód kritériumai:
  - $p_1 < p_2 < \ ... \ < p_N$
  - $l_N = l_{N-1}$ és az utolsó biten különbözőek
  - $p_i > p_j \rightarrow l_i \leq l_j$
- Ordó n négyzet



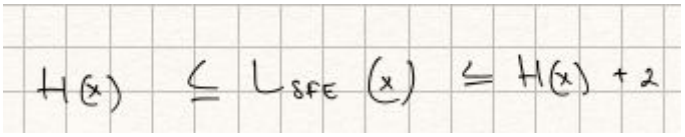
Adaptív Huffman

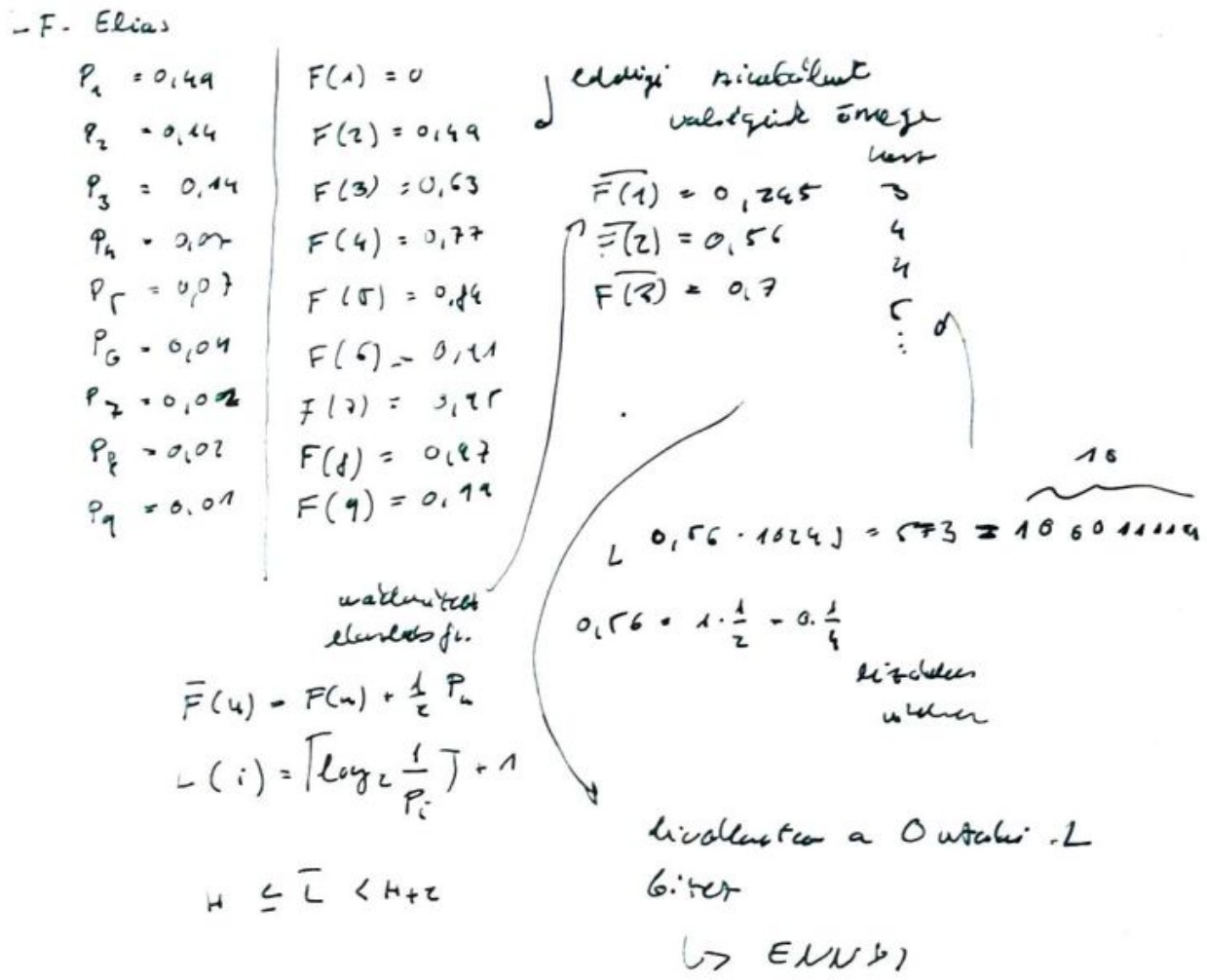
- Eddig: forrásbetűk relatív gyakoriságát ismertük, először elemezzük statisztikailag, aztán ez alapján kódolunk

- Többnyire ezt nem célszerű megvárni, és adaptívan kódolunk: változik a kód az éppen feldolgozott adat függvényében.
- A forrásbetűt az előzőek alapján létrehozott optimális kóddal hajtjuk végre.

Shannon-Fano-Elias kód

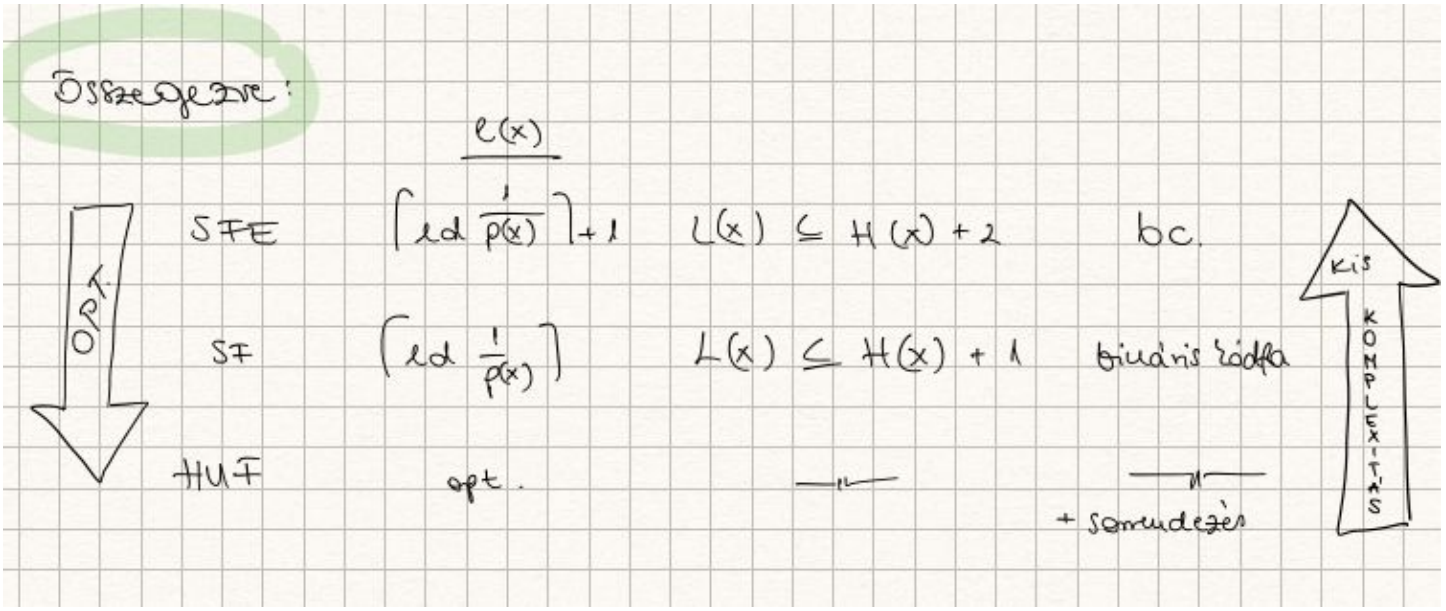
- Aritmetikai kódolás: valós időben történik a kódszó előállítása és visszafejtése, nem lép fel késleltetés akkor sem, ha nagyon hosszú blokkokat kódolunk.





Összehasonlítás

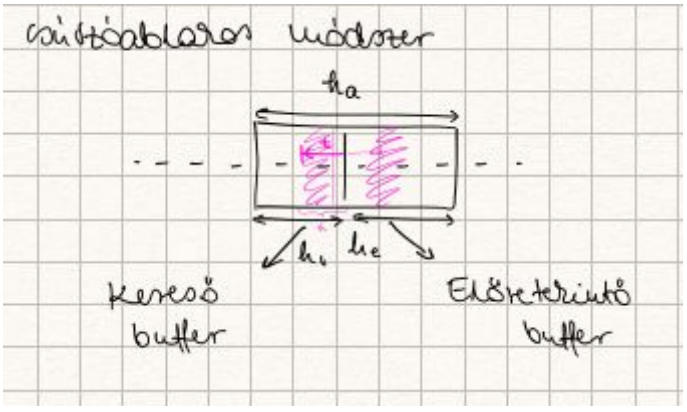
- Dióhéjban: Ahogy nő a teljesítőképesség úgy csökkent az implementálhatóság. Komplexitás vs teljesítőképesség.



8. Describe the LZ based compression algorithm.

- Eddig két infót vittünk át: blokk kódot leíró infó (független az üzenet hossztól, állandó - kód statisztikai leírása) + üzenet kódszavai
- Ha az üzenetet végtelen hosszúnak tekintjük az állandó költség 0-hoz tart, gyakorlatban nem az
- Ötlet: gyűjtsünk menet közben információt a szimbólumokról, vagyis az aktuális szimbólumot, az előzőek alapján kódoljuk -> Adaptív kódok (pl. Adap. Huffman)

LZ77



- A kódoló ezután küld egy <t, h, c> (MARIHUÁNA LOL) hármast, ahol
  - t: kereső pufferben megtalált szimbólum távolsága az előretekintő puffertől
  - h: a kereső és az előretekintő puffer egyező szimbólumainak legnagyobb hosszúsága
  - c: az első előretekintő pufferben lévő nem egyező karakter kódszava
    - hogy elkerüljük azt, hogy az előretekintő puffer szimbólumait nem találjuk a kereső pufferben. Ilyenkor t és h értéke 0.

- Hatékony
- Az a baj, hogy az kell, hogy a szimbólumok periodicitása kisebb legyen mint a keresőbuffer hossza

LZ78

- A kódoló és a dekódoló is szótárt épít az előzőleg előfordult sorozatokból.

a kódoló kimenete	szótár index	bejegyzés	a kódoló kimenete	szótár index	bejegyzés
$\langle 0, f(d)\rangle$	1	d	$\langle 4, f(c)\rangle$	10	bac
$\langle 0, f(a)\rangle$	2	a	$\langle 9, f(b)\rangle$	11	$dabb$
$\langle 0, f(b)\rangle$	3	b	$\langle 8, f(d)\rangle$	12	acd
$\langle 3, f(a)\rangle$	4	ba	$\langle 0, f(e)\rangle$	13	e
$\langle 0, f(c)\rangle$	5	c	$\langle 13, f(c)\rangle$	14	ec
$\langle 1, f(a)\rangle$	6	da	$\langle 1, f(e)\rangle$	15	de
$\langle 3, f(b)\rangle$	7	bb	$\langle 14, f(d)\rangle$	16	ecd
$\langle 2, f(c)\rangle$	8	ac	$\langle 13, f(e)\rangle$	17	ee
$\langle 6, f(b)\rangle$	9	dab			

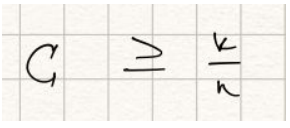
- A kódoló megkeresi a forrásszimbólumok aktuális pozíciójától kezdődő leghosszabb egyezést a szótárban.
- Átküld egy <i, c> párt:
  - i: az egyező karaktersorozat szótárbeli indexe
  - c: az első nem egyező karakter kódja
- Baj: korlátlanul nó a szótár
  - Megoldás: néha eltávolítjuk a ritkán használt részeket, vagy egy idő után fix szótárt

LZW

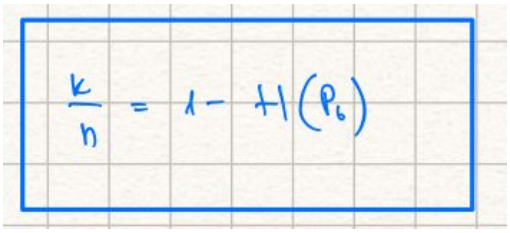
- Terry Welch módosítja az LZ78-at úgy hogy megtakarítható legyen az <i, c> -ből a c elküldése.
- A kódolás során az aktuális pozíciótól kezdve addig olvassuk be a forrásszimbólumot a p pufferbe, amíg a sorozat szerepel a szótárban. Ha az a karakter az első olyan amelyre pa (konkatenált) nincs benne a szótárban, akkor átküldjük a p sorozat indexét, a pa sorozatot felvesszük a szótárba és az a karaktertől kezdve folytatjuk az eljárást.

9. Describe the channel coding theorem, and define the channel capacity and elaborate on its calculation for symmetric channels.

- Csatorna kódolás lényege: zajos csatornán (egy átviteli közeg) átvinni információt
- A csatorna kódolás elvi lehetőségeit a csatornakódolási tétel mutatja meg
- A csatornakapacitás az elvi határa az átvihető információnak.
- k/n az adatátviteli sebesség. mi a max értéke? (shannon tétel - csatorna kódolási tétel)



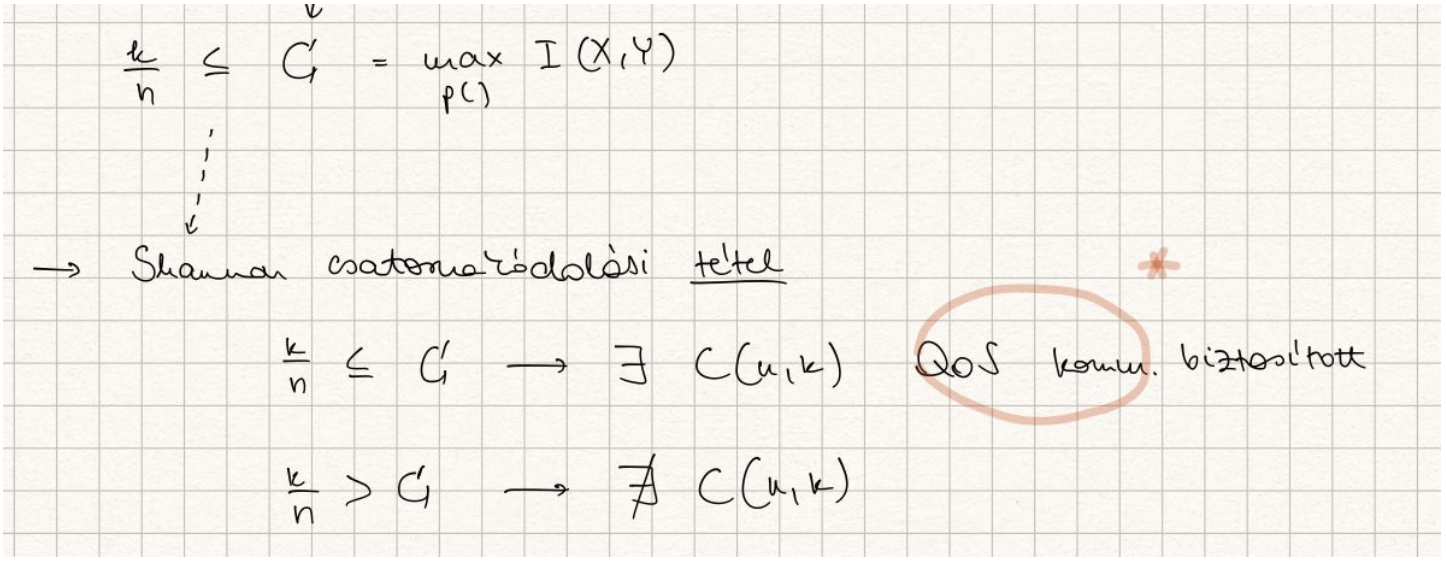
- BSC eset:



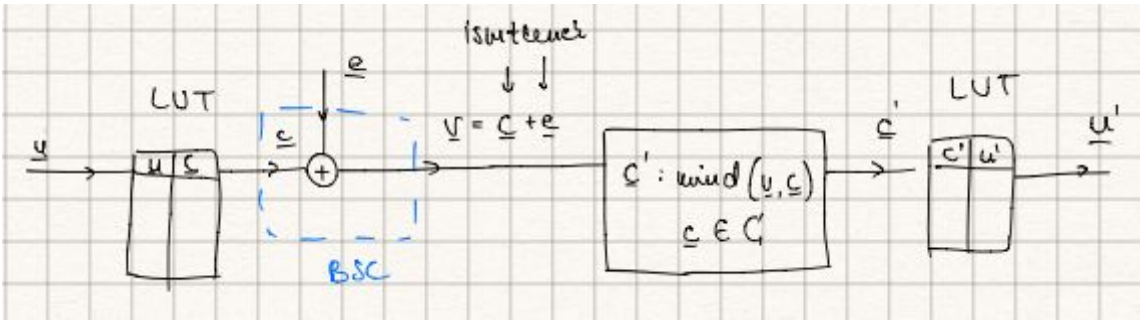
Csatornakapacitás

- adatátviteli sebességet határozza meg.
- memóriamentes csatorna esetén:

$$C = \max_{p(x)} I(x,y) = \max_{p(x)} \{\mathrm{H}(Y) - \mathrm{H}(Y|X)\}$$



10. Define and explain the relationship between the following properties and parameters of error correcting codes: minimum code distance; code-length and message-length versus performance (Singleton and Hamming bounds); general algorithmic complexity of coding with tables



k: üzenet n: kódszó hossza

Minimum code distance

- Hamming táv: eltérő bitek száma

$$d_{min} : \min_{c \neq c'} d(c, c')$$

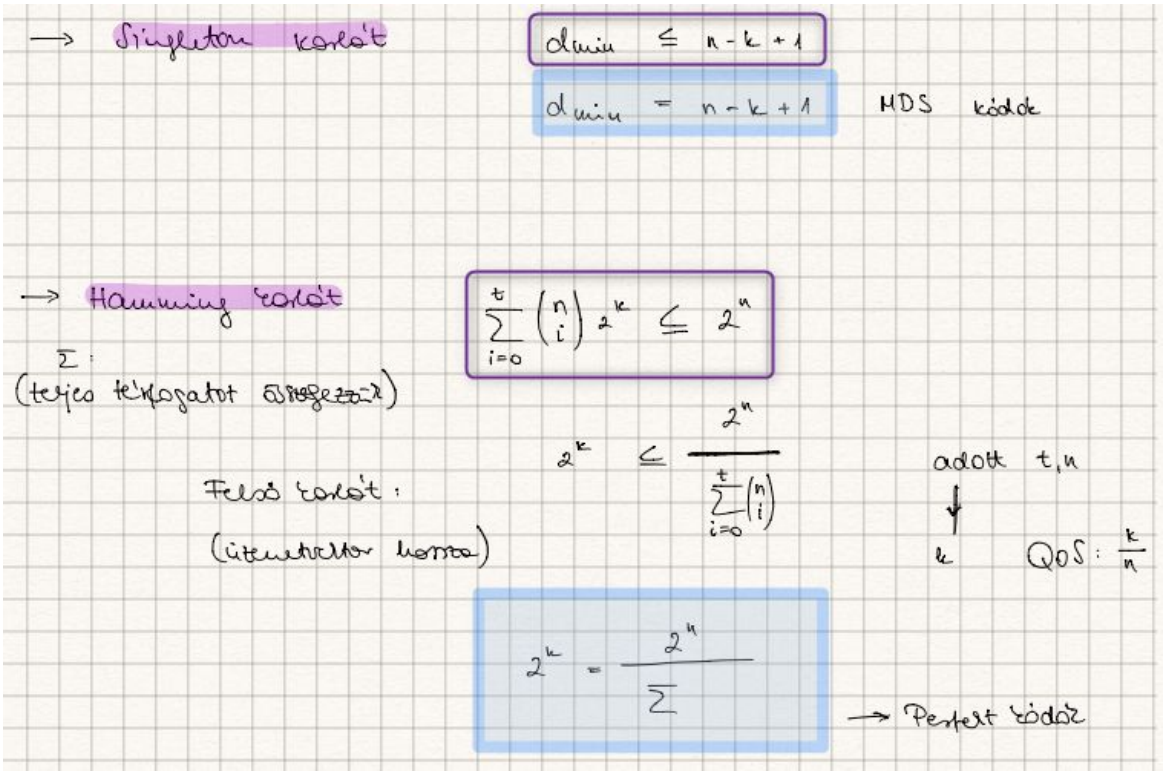
Number of errors

$$l = d_{min} - 1$$

Number of errors we can correct

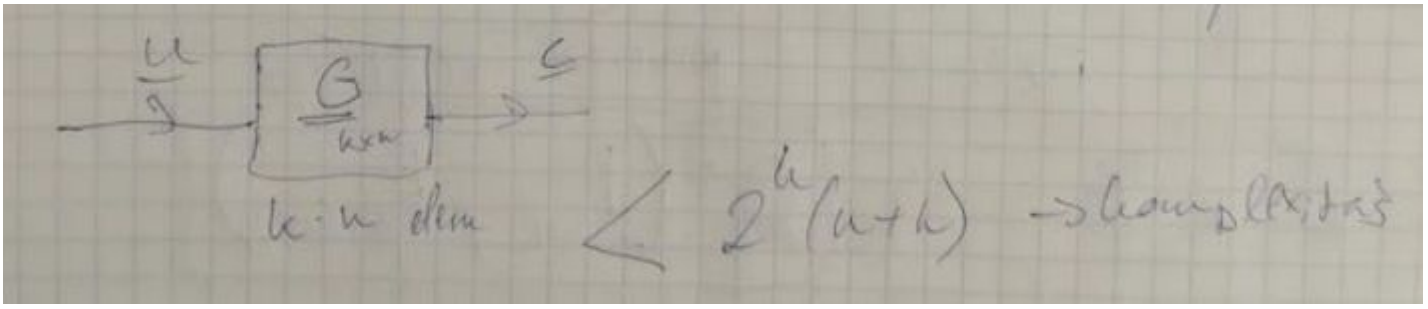
$$t = \left\lfloor \frac{d_{min} - 1}{2} \right\rfloor$$

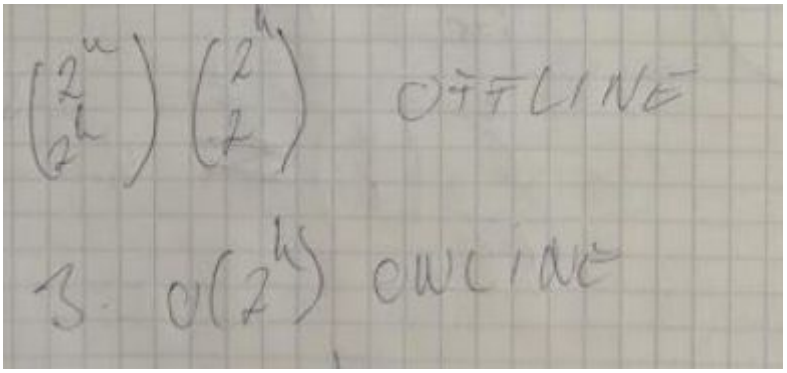
Singleton és Hamming korlát



- MDS: Max. Distance Separable

General algorithmic complexity of coding with tables



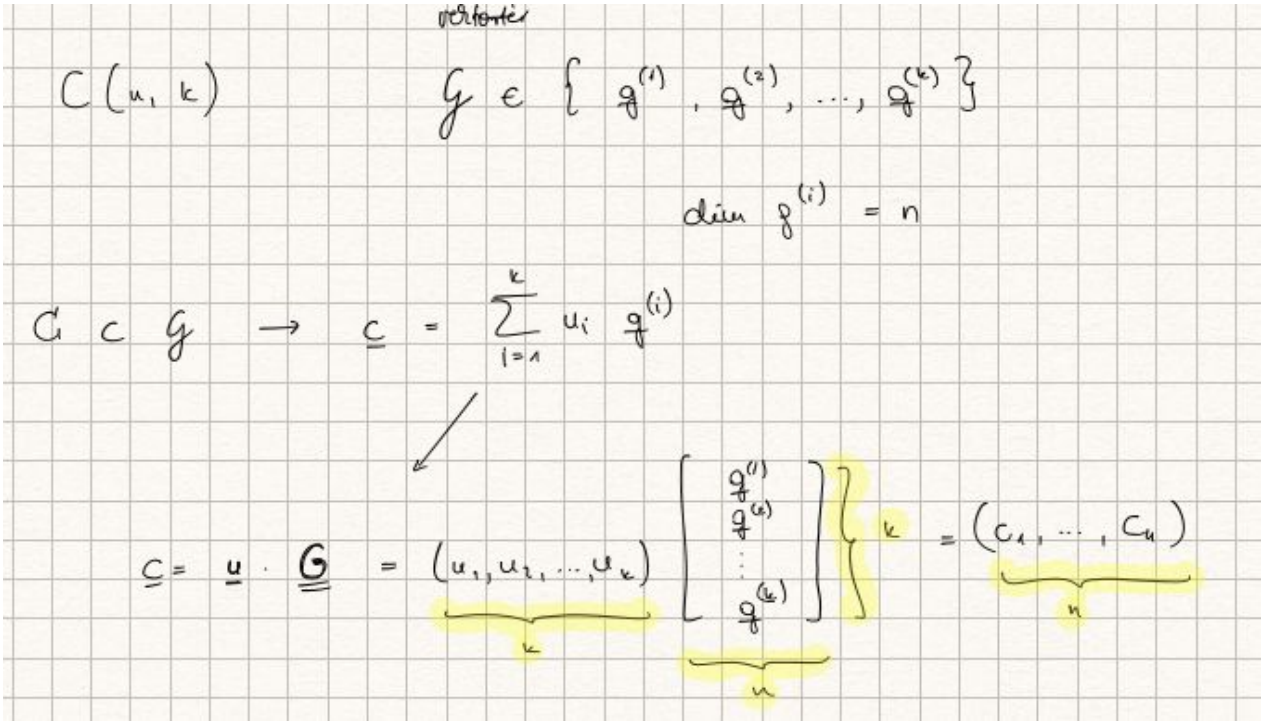


- Online exponenciális -> nem real time megoldható

11. Introduce the concept of linear block coding and explain the meaning of systematic codes, generator matrix and parity check matrix (and their relationship), algorithmic complexity of linear block coding (including detection).

- Az általános look up tables kódolás nem túl hatékony: azért, hogy ezen javítsunk bevezethetjük a lineáris blokk kódolást.

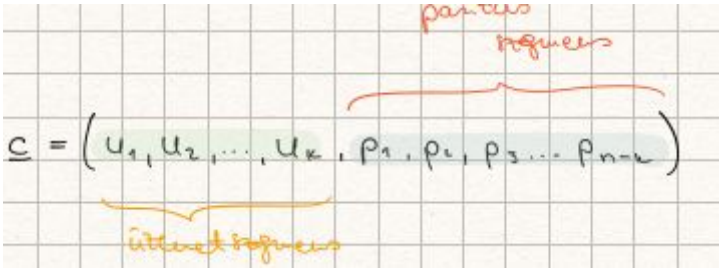
Generátor mátrix

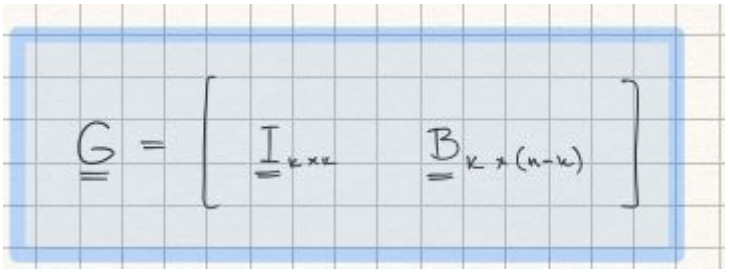


- Ezt a generátor mátrixot összeszorozva az üzenet vektorral megkapjuk a kódvektort.

Szisztematikus kódok

- Egy kódot szisztematikusnak hívunk leírható a üzenet bitjeivel és még pár kiegészítő bittel, valahogy így:

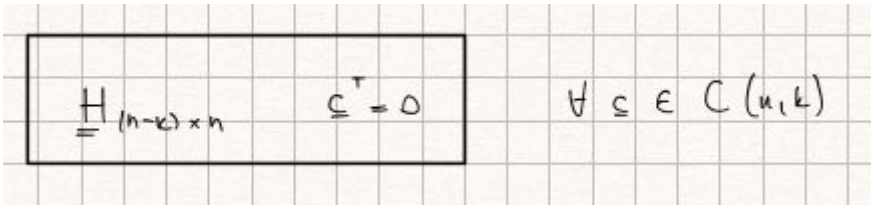




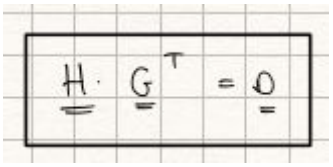
$$\boldsymbol{H} = \left[\boldsymbol{A}_{(n-k)\times k} | \boldsymbol{I}_{(n-k)\times(n-k)}\right], \text{ where } \boldsymbol{A} = \boldsymbol{B}^T$$

- Dekódolásnál csak el kell hagyni az üzenet végét. (értelemszerűen)

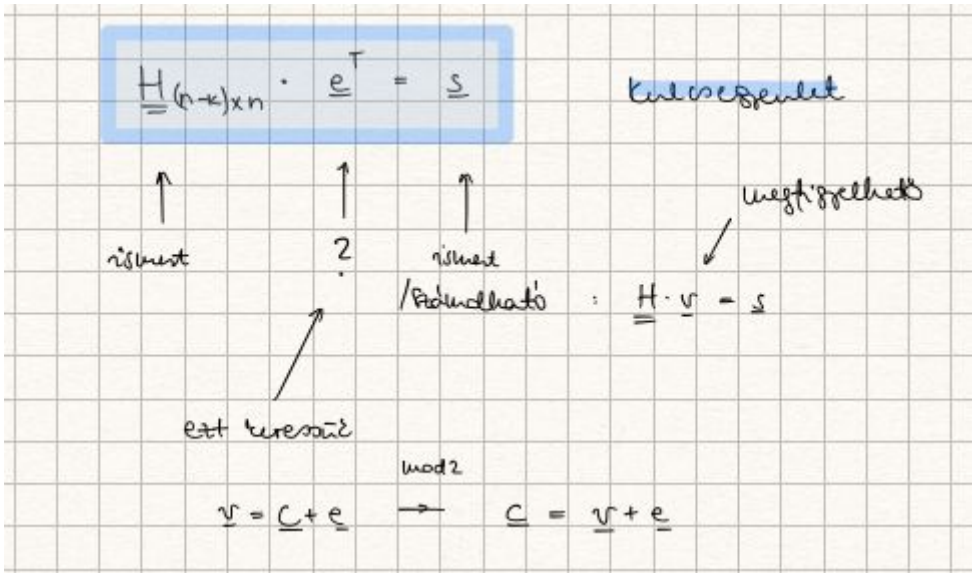
Paritásellenőrző mátrix



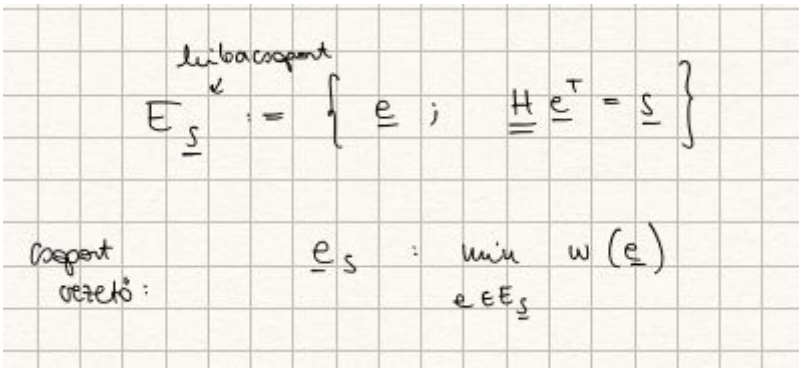
- H segítségével tehát meg tudjuk állapítani, hogy egy vett szó valóban kódszó e.



Szindrómavektor



- Vett szóból ki tudod számolni a szindrómát, amiből ugye e-t, amiből pedig becslést adunk c-re.
- e-re több megoldás -> hibacsoport (2^k elem) (legkisebb súlyú e a csoportvezető)



Extra definíciók

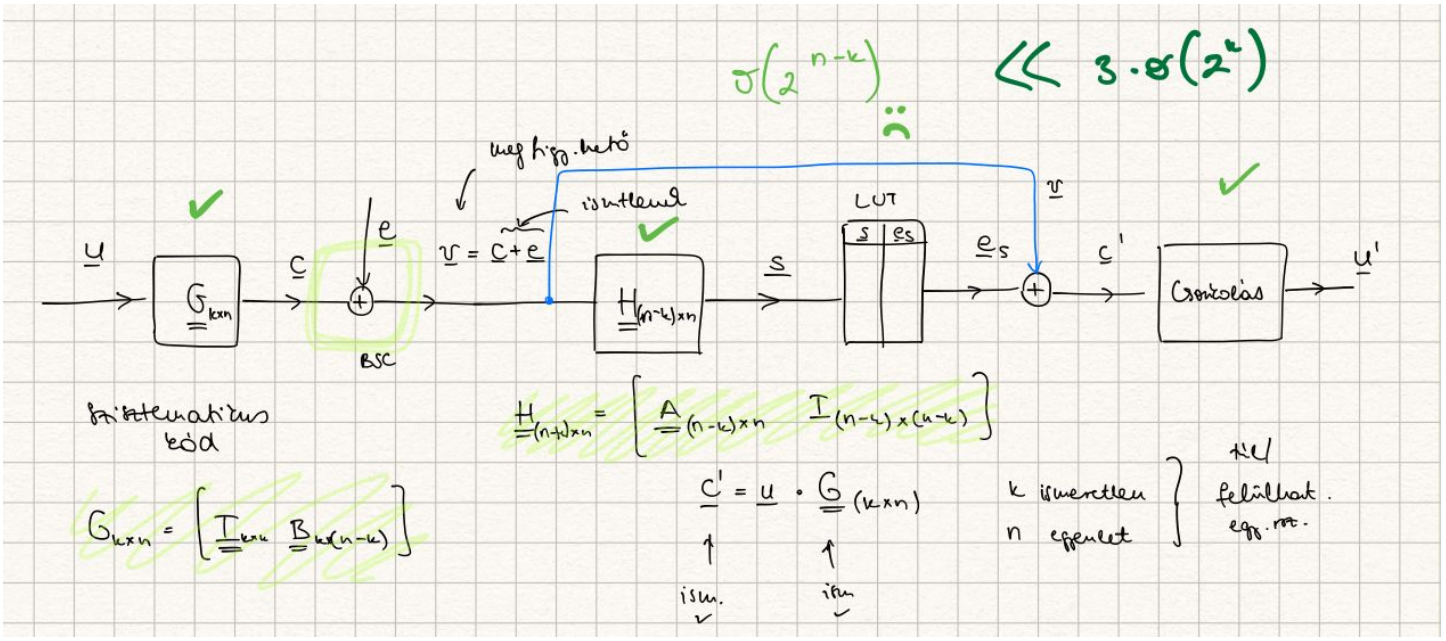
4.9. definíció. *Egy* **c** *vektor* ***súly****a a koordinátái között levő nem nulla elemek száma, jelölése* $w(\mathbf{c})$.

4.4. tétel. *Ha C lineáris kód, akkor a kódtávolsága megegyezik a minimális súlyával, azaz*

$$d_{min} = w_{min}.$$

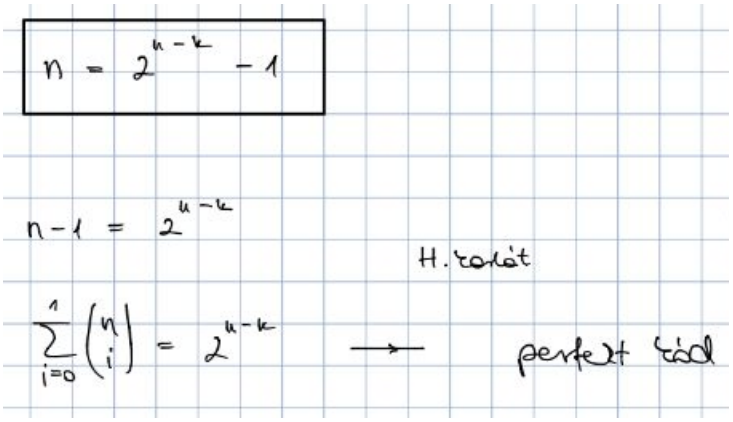
Komplexitás

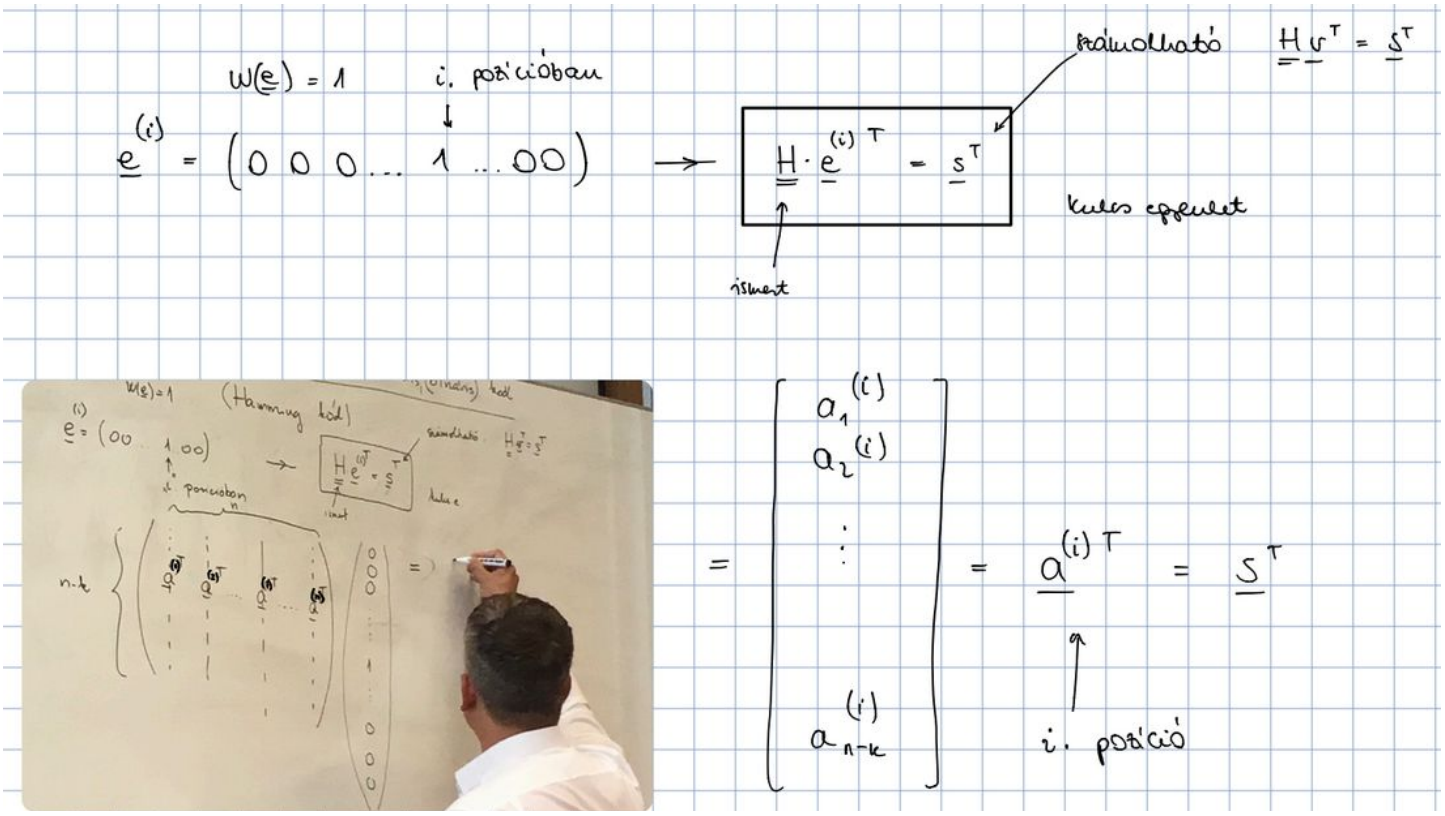
Algorithmic complexity of detection (offline): $\mathcal{O}(2^{n-k})$.
Algorithmic complexity of encoding and decoding (online): $\mathcal{O}(n \cdot k)$.
Algorithmic complexity of the whole scheme: 1 LUT (detection) + 2 matrix-vector multiplication + truncation.



12. Give the construction of binary Hamming codes (define the corresponding matrices and the error correcting capability)

- A Hamming kód egy olyan bináris, lineáris kód, ami 2 hibát tud jelezni és 1-et javítani.
- A Hamming kódok perfekt kódok, mert teljesül a Hamming korlát.
- Helyes detekció feltétele: H oszlopai lineárisan függetlenek (nem nulla, nem egyezik egy sem)





- Ebben a konstrukcióban, mivel tudjuk hogy legfeljebb 1 hiba, az vektor vagy nullvektor, vagy egységvektor, amiből következik, hogy az s megegyezik valamelyik oszloppal.

Multiple error correction

Theorem. If a Hamming-code can correct every t error, $\boldsymbol{A}$ must have at least $2t$ independent column vectors.

Detection rule for $t = 2$: the syndrome vector matches with one of the columns (single error) or with one of the two-element-sums of the column vectors. This is highly inefficient to check.

13. Describe the Reed Solomon codes (generator matrix, parity check matrix, performance)

- több mint 2 hibát akarunk javítani bináris helyett -> q-áris kódok
  - zaj hatása nagyobb

Galois Field

$$GF(q) = \{0, 1, 2, ..., q-1\}$$

- q prím
- Összeadunk meg kivonunk (zárt, asszociatív, kommutatív, egység/semleges elem, inverz elem) + disztributív - MODULO q-val
- elem rendje:

$$ord(\alpha) = \min_{1 \leq k} \{\alpha^k = 1\}$$

- Extra tulajdonságok:

Theorem. $\forall \alpha \in GF(q) \backslash \{0\} : \alpha^{q-1} = 1.$

Definition. $\alpha \in GF(q)$ is a *primitive element* if $\alpha, \alpha^2, ..., \alpha^{q-1}$ expand the field.
Equivalent definition: α is a primitive element, if $ord(\alpha) = q - 1$.

Theorem. Every Galois field has at least one primitive element.

Definition. A *polynom* over GF(q):

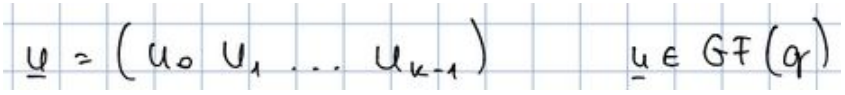
$$a(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \cdots + a_n \cdot x^n$$

This polynom can also be represented as a vector:

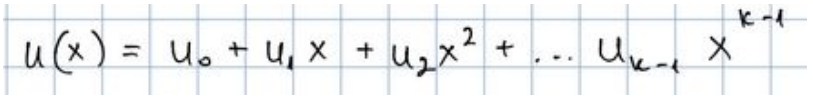
$$a(x) \leftrightarrow \boldsymbol{a} = (a_0, a_1, \ldots, a_n).$$

Reed-Solomon codes

- **C_{RS}(n,k)**, GF(q) felett, ahol **n = q - 1**
- az üzenetet vektoros reprezentációban írjuk fel:



- ehhez rendeljük az üzenetpolinomot:



- Generátor mátrix (alfa primitv element)

$$G_{k\times n} = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \ldots & \ldots & \ldots & \ldots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \ldots & \alpha_n^{k-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ 1 & \alpha & \ldots & \alpha^{n-1} \\ 1 & \alpha^2 & \ldots & \alpha^{2(n-1)} \\ \ldots & \ldots & \ldots & \ldots \\ 1 & \alpha^{k-1} & \ldots & \alpha_n^{(k-1)(n-1)} \end{bmatrix}$$

- Paritás ellenőrző mátrix

$$H_{(n-k)\times n} = \begin{bmatrix} 1 & \alpha & \alpha^2 & \ldots & \alpha^{(n-1)} \\ 1 & \alpha^2 & \alpha^4 & \ldots & \alpha^{2(n-1)} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 1 & \alpha^{n-k} & \alpha^{2(n-k)} & \ldots & \alpha^{(n-1)(n-k)} \end{bmatrix}$$

- Az RS kódok MDS kódok. (következik a Singleton korlátból és, hogy lineáris)
- q^(n-k) LUT komplexitás a hátránya

14. Describe the steps of the Error Trapping Algorithm for error detection in the case of cyclic codes

Lineáris ciklikus kódok

4.21. definíció. *Egy*

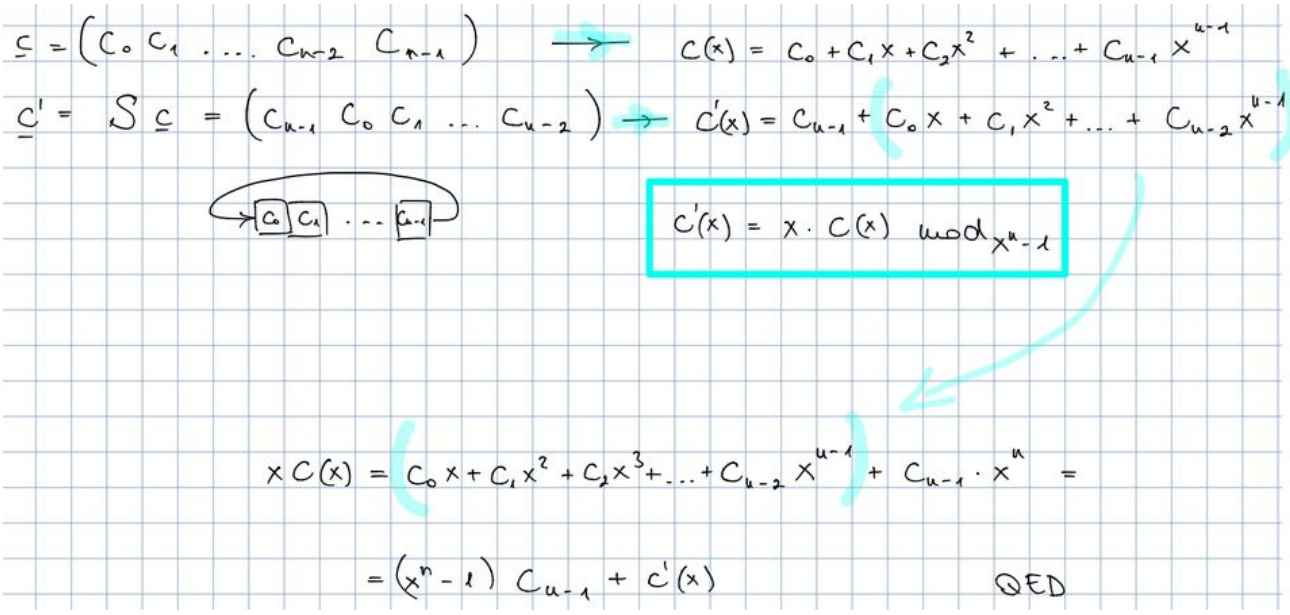
$$\mathbf{c} = (c_0, c_1, \ldots, c_{n-1})$$

vektor ciklikus eltoltja a

$$S\mathbf{c} = (c_{n-1}, c_0, \ldots, c_{n-2}).$$

S-et a ***ciklikus eltolás*** *operátorának nevezzük.*

4.22. definíció. *A C kódot* ***ciklikus****nak nevezzük, ha bármely kódszó ciklikus eltoltja is kódszó.*

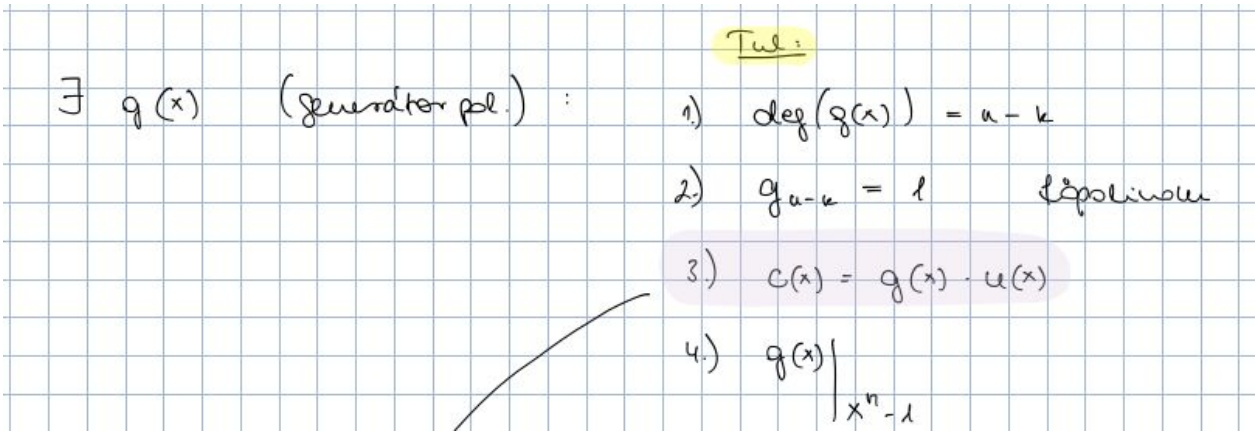


4.23. definíció. *Rendeljünk polinomot az egyes kódszavakhoz a következő módon:*

$$\mathbf{c} = (c_0, c_1, \ldots, c_{n-1}) \mapsto c(x) = c_0 + c_1 x + \cdots + c_{n-1} x^{n-1},$$

ekkor a $\mathbf{c}$ *kódszónak megfeleltetett* $c(x)$ *polinomot* ***kódszópolinom****nak vagy röviden kódpolinomnak nevezzük. A kódszópolinomok halmazát* $C(x)$*-szel jelöljük.*

- Lineáris ciklikus kódokra - ez a generátor polinom



Theorem 6: *If the generator polynomial g(x) of C has degree n-k then C is an [n,k]-cyclic code. If* $g(x) = a_0 + a_1 x + a_2 x^2 + \ldots + a_{n-k} x^{n-k}$ *then a generator matrix for C is the* $k \cdot n$ *matrix,*

$$\begin{bmatrix} g(x) \\ x\,g(x) \\ x^2 g(x) \\ \vdots \\ x^{k-1} g(x) \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{n-k} & 0 & 0 & \cdots & 0 \\ 0 & a_0 & a_1 & \cdots & a_{n-k-1} & a_{n-k} & 0 & \cdots & 0 \\ 0 & 0 & a_0 & \cdots & a_{n-k-2} & a_{n-k-1} & a_{n-k} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & a_{n-k} \end{bmatrix}.$$

(az utsó együttható 1)

ETA

- Az ETA egy módszer, amivel azonosítani tudunk ciklikus kódhibákat, anélkül, hogy LUT-okat használnánk.
- Csak burst errorokat képes felismerni.
  - Eddig minden olyan hibákat javított, amíg random oszlottak el a kódban a burst error lényege, hogy csomósodva van.
- Alap ötlet:
  - (első rész a fogadó oldal, 2. rész a küldés a definícióból)

1) $\underline{v}(x) = u(x) \cdot g(x) + e(x)$
2) $\underline{v}(x) = a(x) \cdot g(x) + r(x)$

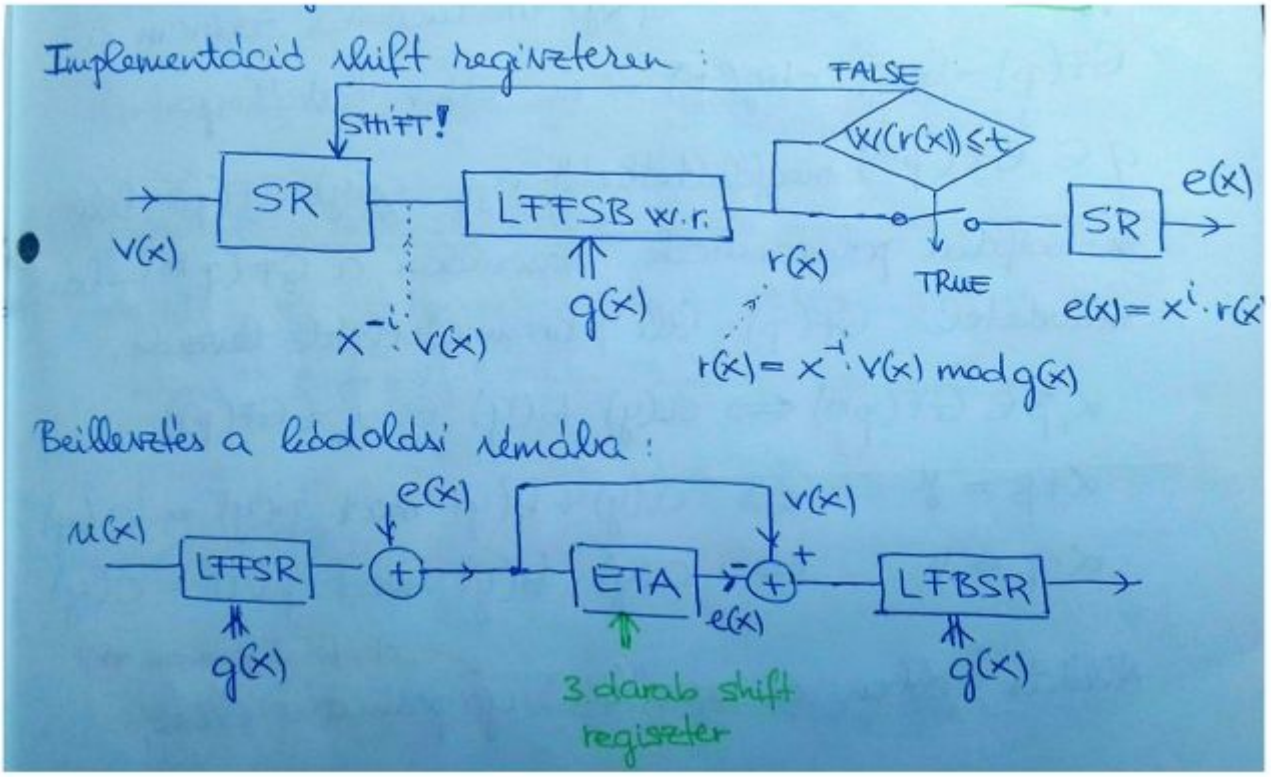
- Tehát ha a megkapott v(x)-et osztjuk g(x)-szel akkor e(x)-et megkapjuk osztási maradékként. Ebből következik, hogy

$$deg(e(x)) \leq n - k - 1.$$

- Mivel azt feltételezzük, hogy burst error, azért azt mondjuk hogy az első és utolsó hibás biz közötti táv max. n-k.
- e(x) felírható így, ahol i_o a hiba kezdete

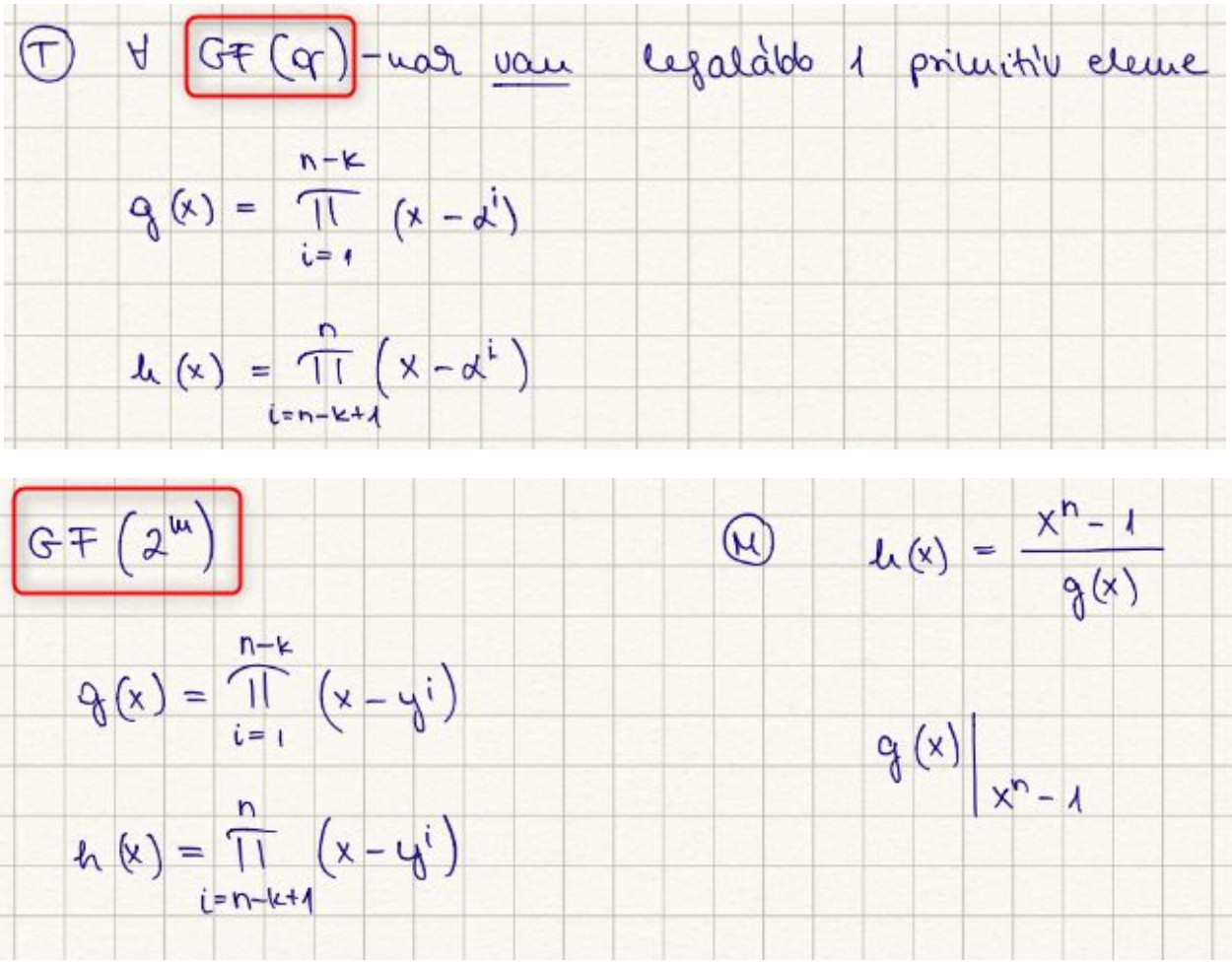
$$e(x) = x^{i_0} r(x).$$

- Ekkor fogjuk és
- NEM ÉRTJÜK ITT AZ ÁBRA:

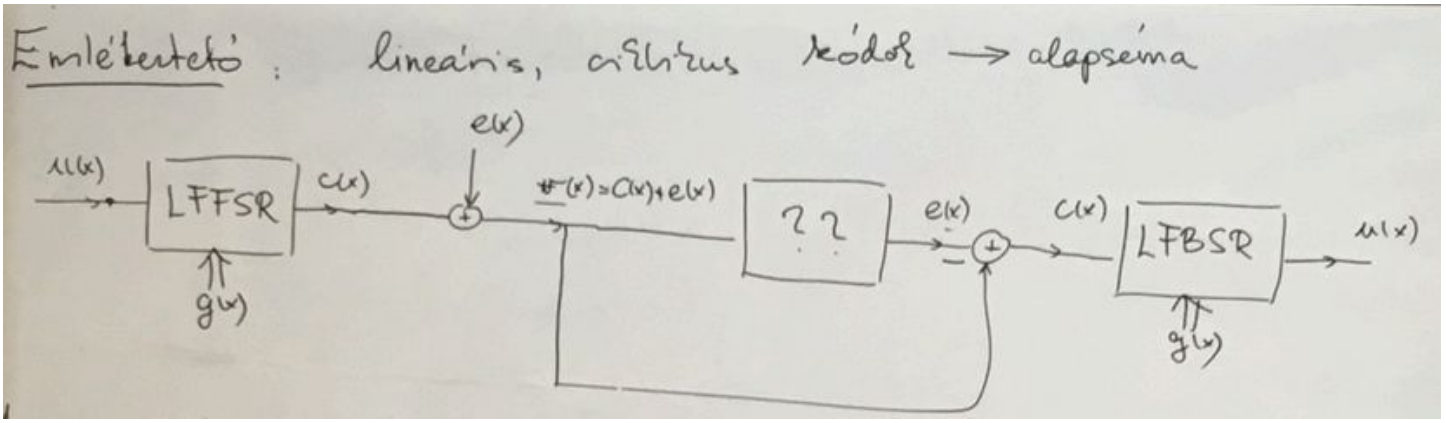


15. Describe the cyclic RS codes (generator polynom, parity check polynom, implementation)

- RS kód: ciklikus MDS



Implementáció



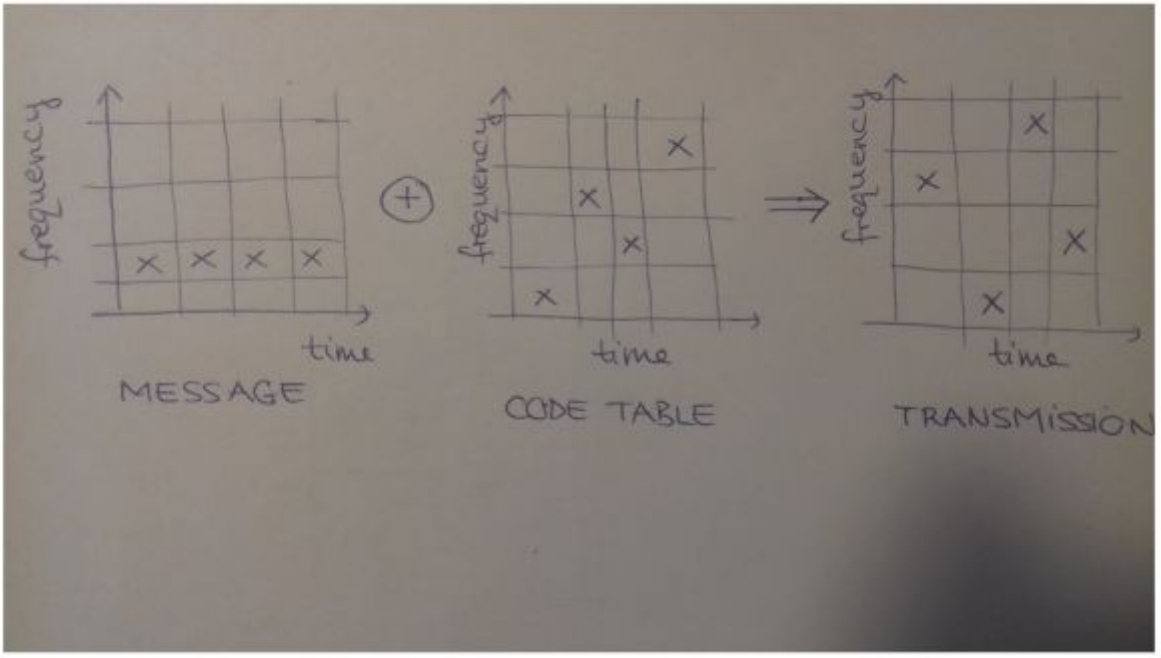
- Shiftregiszterek szarok a qáris domainben, ezért nem implementáljuk őket. Ehelyett használjuk a minimum polinomokat valahogy.

16. Describe the CDMA/FH system and the CDMA/DS system with Walsh-Hadamard codes and with random codes

- Sokan akarnak adni ugyanott egy közös csatin
- Megoldások: TDM, FDM, ortogonális részcsatornák
- Felhasználóknak saját kód: CDMA (Code Division Multiple Access), midnenkié összadódva érkezik mindenkihez

Frequency Hopping

- Van egy frekvencia sorozat (mindketten tudjuk) előnye hogy privát is nan.
- GSM-ben van ilyen. kódtábák
- Majority decision



Direct Sequence

- Na ez egy másik módszer, alternatíva az FH helyett
- modulációs séma: az üzeneteket jellel moduláljuk.
- Ez a jel a signature signal: pszeudo random, rövid rádió pulzusokból áll. Egy ilyen pulzus hossza chiptime. Minél kisebb annál nagyobb lesz a sávszélesség.
- az üznetet szétdarabolja így nagyobb sávszélességet kapunk.

https://www.youtube.com/watch?v=-1mxYWvfVWQ

- Felírtunk nagy képleteket és az a lényeg, hogy kaptunk egy tagot ami a Multi User Interfernce. Ezt küszöböljük ki az ortogonális kódokkal.

Walsh-Hadamard

- lehetővé teszi az M=2^k ortogonális kódok konstrukcióját.
- Truly orthogonal codes: Two codes are said to be orthogonal if when they are multiplied together the result is added over a period of time they sum to zero. For example a codes 1 -1 -1 1 and 1 -1 1 -1 when multiplied together give 1 1 -1 -1 which gives the sum zero.
- Rekurzívan képezzük ezt a Walsh cuccost így:

$$\boldsymbol{C}(0) = 1. \quad \boldsymbol{C}(k+1) = \left[\begin{array}{c|c} \boldsymbol{C}(k) & \boldsymbol{C}(k) \\ \hline \boldsymbol{C}(k) & -\boldsymbol{C}(k) \end{array}\right]$$

- Ezek a kódok ortogonálisak lesznek, szóval nem lesz MUI.

Random kódos CDMA/DS

- Walsh: kevés user használható, mert szigorú nagyon a feltétel.
- Ezért nem teljesen ortogonális kódokat használunk, hanem random generált KVÁZI ortogonális kódokat, amik követik a bernoulli eloszlást:

$$P(c_m^{(i)} = 1) = P(c_m^{(i)} = -1) = 0{,}5$$

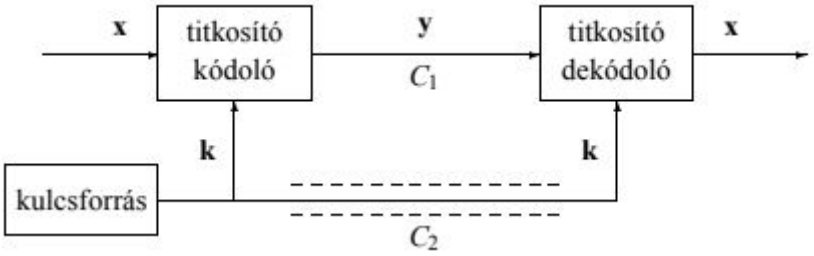
- Nem teljes MUI fix, nagyobb a P_b
- Ezt lehet minimalizálni valahogy

17. Describe the OTP method and RSA algorithm for cryptography

- cél: titkosítani szeretnénk, csak a fogadó tudjon olvasni üzenetet

OTP (one time pad /véletlen átkulcsolás/ Vernam féle rejtjelező)

- kulcs hossz = szöveg hossz
- kulcs teljesen random, ugyanazzal XOR-olunk mindkét oldalt.
- kulcsot tikos csatin küldjük
- tökéletes titkosítást valósít meg, tehát a titkosított szöveg vizsgálata nem juttatja a a vizsgáló személyt új információhoz.



- előny: optimális, nagy a kulcs, minden üzenethez saját kulcs kell.

Kriptográfiai támadások

- passzív: lehallgatás
- aktív: üzenetmódosítás

RSA

- Nyilvános kulcsú rejtjelezés (publikus, privát kulcs)
- Adatátvitel / digitális aláírás

Matematikai háttér

- Euklideszi algoritmus
  - Két szám LNKO-ját tudjuk vele kiszámolni
  - Tehát fogod b-t és elosztod c-vel maradékosan. Aztán c-t osztod az előző maradékkal maradékosan. Stb. Az utolsó nem nulla maradék b és c LNKO-ja.
- Kis Fermat
  - Ha van egy p prímünk, és egy a számunk, ami nem osztható p-vel, akkor

$$a^{p-1} \equiv 1 \pmod{p}.$$

- Fermat tétel általánosítása
  - Ha van p_1 és p_2 prímünk és a, p_1, p_2 legnagyobb közös osztója 1, akkor:

$$a^{(p_1-1)(p_2-1)} = 1 \pmod{p_1 p_2}.$$

- b modulo m inverze a-nak, ha ab=1 mod m

Az RSA algoritmus lépései

1. Válasszunk két random nagy prímet: p_1, p_2
2. Számoljuk ki az alábbi paramétereket (fi: euler fv)

$$m = p_1 p_2 \qquad \phi(m) = (p_1 - 1)(p_2 - 1)$$

3. Válasszunk egy véletlenszerű e-t, úgy hogy

$$1 \leq e < \phi(m) \qquad (\phi(m), e) = 1$$

4. Számoljuk ki e inverzét modulo fi(m)

$$d = e^{-1} \pmod{\phi(m)}$$

5. Az m, e egészeket nyilvánosságra hozzuk (nyilvános kulcs), és a d, p_1, p_2 értékeit titokban tartjuk (privát kulcs). Tehát

$$\mathbf{k}^p = (m, e) \quad \mathbf{k}^s = (d, p_1, p_2)$$

6. A titkosító kódolás az 1 <= x < m nyílt üzenetre egy 1 <= y < m rejtett üzenetet ad, ahol
   a. Kódolás

$$y = x^e \pmod{m},$$

   b. Dekódolás

$$x = y^d \pmod{m}$$

- Ugye érdemes megjegyezni, hogy az egész azon a problémán alapul, hogy feltételezzük, hogy ez egy nehéz probléma. Sejtések alapján az m prímtényezőkre bontásának nehézségével egyezik meg. Most jelenleg úgy tudjuk hogy ez lehetetlen belátható időn belül megoldani, ha elég nagy a két prím, de simán lehet hogy valami nagyon okos ember megoldja ezt a problémát.