

Információbiztonság és Adatvédelem

2025/2026

László Emő

Félév teljesítése

- **Évközi feladatok házi feladatok**
 - nem teljesítés esetén nem lehet vizsgázni
- Vizsga / ZH
 - Teszt feladat – elmélet (40%)
 - Gyakorlati feladat (60%)

Miről lesz szó

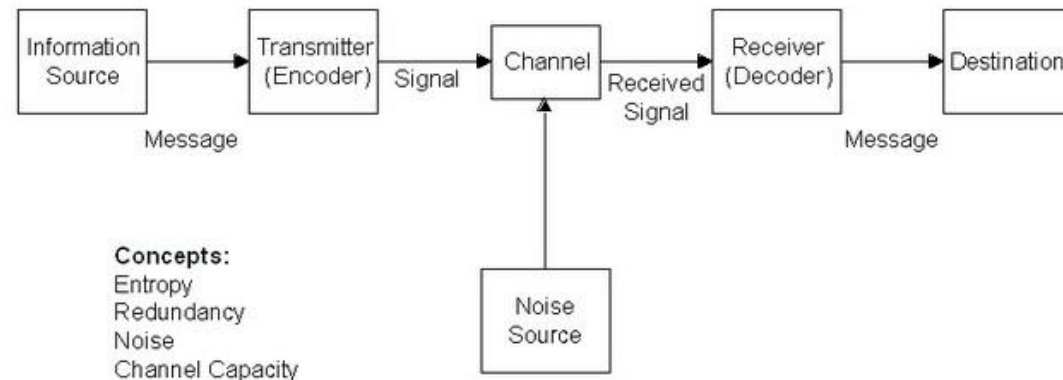
- **Információelmélet alapfogalmak** – Shannon, entropia, tömörítés
- **Klasszikus kriptográfia** – Caesar, Vigenère, Enigma
- **Hibajavítás és csatornamodellek** – Hamming-kód, Reed–Solomon
- **Modern kriptográfia** – AES, RSA, hash-ek, digitális aláírás
- **Hálózati biztonság** – SSL, TLS, VPN, MITM támadások
- **Adatvédelem, GDPR** – anonimizálás, nyomkövetés, metaadat
- **Támadások és védekezés** – brute force, phishing, ransomware, sql injection
- **Jövő és kutatási irányok** – kvantumkriptográfia, AI biztonság

Gyakorlati feladatok

Információ

- **Információ = bizonytalanság csökkenése.**(Shannon)
(<https://hu.wikipedia.org/wiki/Shannon%E2%80%93Weaver-modell>)
- Matematikai megközelítés (mérhető)
 - Bit
 - Redundancia
 - Entrópia

The Shannon-Weaver Mathematical Model, 1949



Példák

- Holnap vizsga lesz!

Mekkora az információ tartalma?

- Minél kevésbé várt egy üzenet, annál több információt hordoz.

$$I(x) = -\log_2(P(x))$$

- **I(x)** az x esemény információtartalma
- **P(x)** az esemény valószínűsége
 - Ha valami **valószínűtlen** (pl. 1/1000), akkor **sok bit** információt hordoz.
 - Ha valami **nagyon valószínű** (pl. 0.99), akkor **kevés bitet** (közel 0).

Példák

- Feldobott érme információ tartalma: 1 bit ($-\log_2(0.5)$)
- Ha az érme mindig fejet mutatna akkor nincs információ tartalma mert nem közöl váratlant

Miért fontos ez?

- PI kódolásnál Morsee kód . E a leggyakoribb betű ezért ez a legrövidebb mert a legkisebb az információ tartalma
- Q ritkább ezért hosszabb kód.

Esemény	Valószínűség	Információ (bit)	Megjegyzés
Napsütés nyáron	0.9	~0.15 bit	Nem sok új, várt esemény
Hóesés nyáron	0.01	~6.6 bit	Nagyon váratlan → sok információ
Érmefeldobás (fej/írás)	0.5	1 bit	Klasszikus példa
Új jelszó megadása	1/1,000,000	~20 bit	Ritka → magas információtartalom

Miért számít a biztonság?

- Ti hol tároltok jelszavakat?
- Törték már fel bármilyen online fiókot?
- Mi baj származhat belőle?

Az információelmélet és az adatbiztonság kapcsolata

- Információelmélet: tömörítés, hatékony adatátvitel → **mit tudunk megspórolni?**
- Adatbiztonság: titkosítás, integritás, hitelesítés → **mit kell megvédeni?**
- A kettő közös alapja: **mintázatok felismerése és elrejtése.**
- Példa:
 - Tömörítés = a *mintázat* alapján eldobunk redundáns adatot (ZIP, JPEG)
 - Titkosítás = a *mintázat* elrejtése, hogy még az se értse meg, aki elfogja.

Adatszivárgás, adatlopás esetek (1)

– Enigma feltörése – mit okozott?

Év	Esemény
1918	Enigma-gép első változata (német fejlesztés)
1932	Lengyel matematikusok feltörik az Enigma egyszerűbb verzióját
1939	Turing csatlakozik a Bletchley Parkhoz (UK)
1940	Bombe gép működni kezd (Turing és Welchman)
1941	Daily kód feltörve – fordulópont
1945	Háború vége – a munka titkos marad évtizedekig

Adatszivárgás, adatlopás esetek (2)

- Enigma ismerttő:
https://www.youtube.com/watch?v=G2_Q9FoD-oQ
- Titkosító eljárások kipróbálása:
<https://cryptii.com/pipes/enigma-machine>

Adatlopás céljai (1)

- **1. Pénzügyi nyereségszerzés**
 - **Bankkártyaadatok** ellopása → közvetlen pénzlehívás, online vásárlás
 - **Ransomware:** fájlok titkosítása után váltságdíj követelése
 - **Adatok eladása** a dark weben (pl. jelszavak, email listák)
- **2. Kémkedés, hírszerzés**
 - Állami szintű kibertámadások → katonai, diplomáciai vagy gazdasági titkok megszerzése
 - Ipari kémkedés: versenytárs szellemi tulajdonának megszerzése
- **3. Profilalkotás és manipuláció**
 - Felhasználói szokások elemzése marketing vagy politikai célokra
 - Algoritmusok tanítása (AI modellek) ellopott adatokkal
 - Facebook / Cambridge Analytica → szavazók manipulálása

Adatlopás céljai (2)

- **4. Személyes zsarolás vagy bosszú**
 - Intim vagy kényes adatok kiszivárogtatása (pl. Ashley Madison hack)
 - „Doxing” – valaki személyes adatainak nyilvánosságra hozása
- **5. Szabotázs / rombolás**
 - Adatbázis törlés, módosítás → működésképtelenné tenni egy rendszert
 - Bizalom lerombolása (pl. nyilvánosságra hozott ügyféladatok)

Adatlopás által okozott károk

– 1. Gazdasági kár

- Válságdíjak, bírságok (pl. GDPR → akár éves árbevétel 4%-a)
- IT-rendszerek helyreállítási költsége
- Ügyfelek elvesztése → bevételkiesés

– 2. Reputációs kár

- Ügyfelek, partnerek, befektetők **bizalmának elvesztése**
- Negatív sajtómegjelenések → márka értékének csökkenése
- Cégérték esése (pl. Yahoo felvásárlási ára is csökkent emiatt)

– 3. Személyes következmények

- Az érintett személyek:
 - elveszíthetik állásukat (pl. bizalmas e-mailek szivárgása)
 - pszichológiai terhelést élnek meg (megfélemlítés, stressz)
 - anyagi károkat szenvednek el (identitáslopás)

Adatlopás által okozott károk

- **4. Jogi következmények**

- GDPR / adatvédelmi hatósági eljárás
- Peres eljárások (kártérítési igények)
- Hatósági vizsgálatok, törvényi felelősség

- **5. Szolgáltatás-kimaradás / leállás**

- Adatlopás után a rendszer működése **leállhat** (pl. zsarolóvírus)
- Kórházak, tömegközlekedés, közigazgatás működését is érintheti

Adatlopás által okozott károk

- **4. Jogi következmények**

- GDPR / adatvédelmi hatósági eljárás
- Peres eljárások (kártérítési igények)
- Hatósági vizsgálatok, törvényi felelősség

- **5. Szolgáltatás-kimaradás / leállás**

- Adatlopás után a rendszer működése **leállhat** (pl. zsarolóvírus)
- Kórházak, tömegközlekedés, közigazgatás működését is érintheti

Adatszivárgás, adatlopás esetek

1. Feladat

Mindenki lehetőleg különböző adatszivárgás esetről írjon egy 1-2 oldalas ismertetőt.

- Mi történt?
- Mi volt a cél?
- Elkerülhető lett volna?
- Használták az információt?

Szteganográfia

- Görög eredetű szó: *steganos* = fedett, *graphein* = írni
- Jelentése: információ elrejtése, nem csak titkosítása
- Cél: ne is derüljön ki, hogy üzenet van
- Kriptográfiával szemben nem a tartalom védelmét, hanem a **kommunikáció tényének** elrejtését célozza

Történelmi példák

- Hérodotosz: *viasztábla* (a viasz alá karcolt üzenet)
- Hisztiaiosz: rabszolga fejére tetovált titkos üzenet
- Láthatatlan tinta: citromlé, tej → meleg hatására válik láthatóvá
- Gárdonyi Géza: *Egy magyar rab levele* – akrosztichon (első betűk rejtik a valódi üzenetet)

Gárdonyi Géza – Egy rab levele

Kedves ezüstös, drága dadám!
Ezer nemes arany tizedét örömmel ropog-
tasdörök keserűség ivó magzatodért.
Egészségem gyöngy. A vaj árt. Ritkán óhaj-
tóm sóval, borossal. Ócska lepedőben szá-
rítkozomálmomban, zivataros estén. Matyi
bátyám, bízom rózsát, reze, tús-
egy, lapos leveleddel eressze hajlékomba.
Erzsi, tűt, faggyút, ollót, gombot, levendulát adj!
Laci, nefelejs!

Imre

„Kedves ezüstös, drága dadém!
Ezer nemes arany tizedét örömmel ropog-
tasdörök keserűség ivó magzatodért.
Egészségem gyöngy. A vaj árt. Ritkán óhaj-
tomsóval, borssal. Ócska lepedőben szá-
rítkozomálmomban, zivataros estén. Matyi
bátyám, egypár rózsát, rezet, ezüstöt, libát
egy lapos leveleddel eressze hajlékomba.
Erzsi, tűt, faggyút, ollót, gombot, levendulát
adj!

Laci, nefelejts!

Imre”

Gárdonyi Géza – Egy rab levele

„Kedves ezüstös, drága dádém!

Ezer nemes arany tizedét örömmel ropogtasd
örök keserűség ivó magzatodért.

Egészségem gyöngy. A vaj árt. Ritkán óhajtom
sóval, borssal. Ócska lepedőben szárítkozom
álmomban, zivataros estén. Matyi bátyám,
egypár rózsát, rezet, ezüstöt, libát egy lapos
leveleddel eressze hajlékomba.

Erzsi, tűt, faggyút, ollót, gombot, levendulát adj!
Laci, nefelejts!

Imre”

Gárdonyi Géza – Egy rab levele

„Kedves ezüstös, drága dadém!

Ezer nemes arany tizedét örömmel ropogtasd
örök keserűség ivó magzatodért.

Egészségem gyöngy. A vaj árt. Ritkán óhajtom
sóval, borssal. Ócska lepedőben szárítkozom
álmomban, zivataros estén. Matyi bátyám,
egypár rózsát, rezet, ezüstöt, libát egy lapos
leveleddel eressze hajlékomba.

Erzsi, tűt, faggyút, ollót, gombot, levendulát adj!
Laci, nefelejts!

Imre”

Kedden a török kimegy a városból. Száz emberrel el lehet foglalni.

Modern szteganográfia

- Digitális képek: pixelértékek legalacsonyabb bitjeiben rejthető adat (LSB)
- Hangfájlok: frekvenciák manipulálása az információ elrejtésére
- Videók, dokumentumok: metaadatok, fájlstruktúra használata
- Tipikus formátumok: PNG, WAV, MP4

Alkalmazási területek és kihívások

- Cenzúra megkerülése (diktatúrákban)
- Digitális vízjelek (pl. szerzői jogvédelem)
- Malware rejtése (kibertámadások)
- **Kihívások:** felismerés nehézsége, automatikus detektálás, ellenőrzés
- **Kriptográfiával** kombinálva hatékonyabb
- Szubliminális üzenetek - 25. képkocka
 - Éhes vagy!, Igyál kólát! => 58%-al nőtt a kóla és 18%-al a pattogatott kukorica eladása

Kriptográfia

- Ógörög eredetű: κρυπτός (kryptós) = „rejtett”, γράφειν (gráphein) = „írni”, tehát „titkosírás”
- Kriptográfia: információrejtés
- Kriptoanalízis: visszafejtés
- Kriptológia: kriptográfia + kriptoanalízis
- Állandó „harc”: rejtjelezők vs. kódfeltö

Kriptográfia célja

- **Titkosítás (Confidentiality)** – csak az olvassa, akinek szabad
- **Integritás (Integrity)** – ne lehessen észrevétlenül megváltoztatni
- **Hitelesítés (Authentication)** – tudjuk, ki küldte
- **Letagadhatatlanság (Non-repudiation)** – ne lehessen utólag letagadni

Klasszikus Kriptográfia

- **Példák:**
 - **Caesar-kód:** betűk eltolása egy fix kulccsal
Pl. HELLO → KHOOR (3 pozícióval eltolva)
 - **Vigenère-kód:** kulcsszó alapján váltakozó eltolások
 - **Enigma-gép:** bonyolult rotoros mechanikus rendszer (WW2)
- könnyen megérthető, de könnyen feltörhető a mai technikával.

Klasszikus Kriptográfia

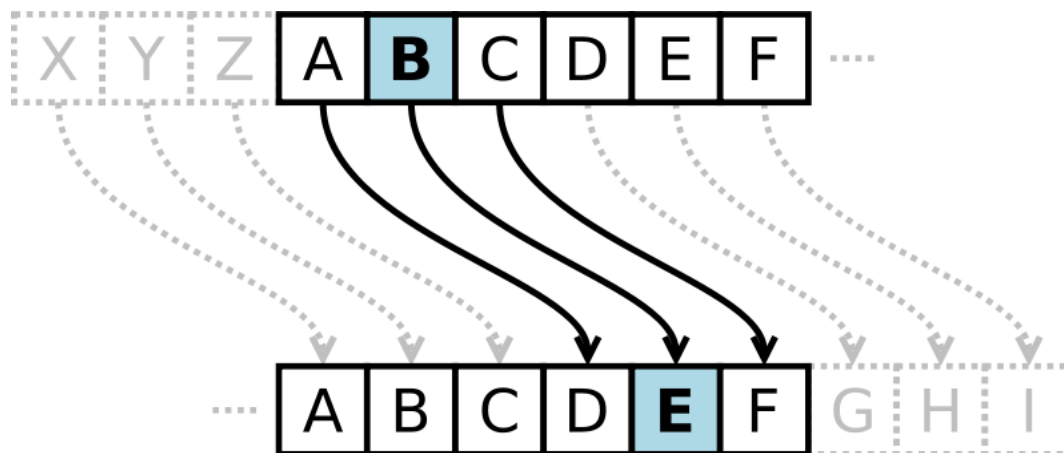
– Caesar-kód

Minden betűt kicserél egy, az ABC-bentőle k távolságra lévő betűvel

Általánosítva:

Minden betű helyett egy másikat használunk => bonyolultabb
26! Lehetőség

Ezt biztos nem lehet megfejteni, hiszen rengeteg párosítást kell végignézni... gondolták hosszú évszázadokig



Klasszikus Kriptográfia

Iszlám világ – Korán – több verzió – Mohamed szóban terjesztette később írták le

Arab tudósok vizsgálták mely részek származnak Mohamedtől és melyek nem.



Szavak előfordulását elemezték, majd a betűket is vizsgálták

Megszületett a **gyakoriságelemzés**

Klasszikus Kriptográfia

Smidla József tanár úr jegyzetei

Miről lesz szó

-  **Információelmélet alapfogalmak** – Shannon, entropia, tömörítés
-  **Klasszikus kriptográfia** – Caesar, Vigenère, Enigma
- **Hibajavítás és csatornamodellek** – Hamming-kód, Reed–Solomon
- **Modern kriptográfia** – AES, RSA, hash-ek, digitális aláírás
- **Hálózati biztonság** – SSL, TLS, VPN, MITM támadások
- **Adatvédelem, GDPR** – anonimizálás, nyomkövetés, metaadat
- **Támadások és védekezés** – brute force, phishing, ransomware, sql injection
- **Jövő és kutatási irányok** – kvantumkriptográfia, AI biztonság

Gyakorlati feladatok

A csatorna kihívásai (1)

IT megjelenése előtt

A futárt elfogják, lelövik, kicserélik az üzenetet

Informatika megjelenésével mi változott?

SEMMI

Csomagküldés – (nagyon titkos információ küldése) (1)

Küldő:

- 1) Bőröndbe rakod a titkot**
- 2) Gyártasz egy kulcsot a bőröndhöz**
- 3) Bezárod a kulccsal a bőröndöt**
- 4) A kulcsot egy számozás „borítékba” rakod amit a vevőtől kapsz**
- 5) A bőröndöt és a borítékot belerakod egy dobozba ami hungarocellel van kipárnázva (sérülések ellen)**

Csomagküldés – (nagyon titkos információ küldése) (2)

Vevő:






- 1) Megkapja a dobozt**
- 2) Ellenőrzi a sérüléseket és javítja (ha tudja)**
- 3) A borítékot ki tudja nyitni mert ismeri a kódot**
- 4) A borítékban lévő kulccsal kinyitja a bőröndöt**
- 5) Elolvassa a titkos üzenetet**

Csomagküldés – IT oldalról

– Küldő oldal (Te)

-  **Van egy adatod** (pl. egy .docx fájl)
-  **Véletlenszerűen generálsz egy AES kulcsot** (olyan, mint egy zár a fájlodhoz)
-  **AES-sel titkosítod a fájlt**
→ most már olvashatatlan, mint egy lakatolt bőrönd
-  **RSA publikus kulccsal titkosítod az AES kulcsot**
→ csak a címzett tudja majd visszafejteni (ő a kulcs tulajdonosa)
-  **Reed–Solomon hibajavító kódot generálsz a teljes csomagra**
→ így ha néhány byte sérül útközben, a vevő helyre tudja állítani
-  **Elküldöd a következőket:**
 - AES-sel titkosított adat (a bőrönd)
 - RSA-val titkosított kulcs (a lakat kulcsa, titkos rekeszben)
 - Reed–Solomon ellenőrző kód (védőcsomagolás)

Vevő oldal (barátod)

-  **Megkapja az adatot + hibajavító kódokat**
-  **Reed–Solomon hibajavítással kijavítja az esetleges sérült byte-okat**
-  **RSA privát kulcsával visszafejti az AES kulcsot**
-  **AES kulccsal visszafejti az adatot**
-  **Megkapja az eredeti fájlt, teljesen helyreállítva és titoktartva**

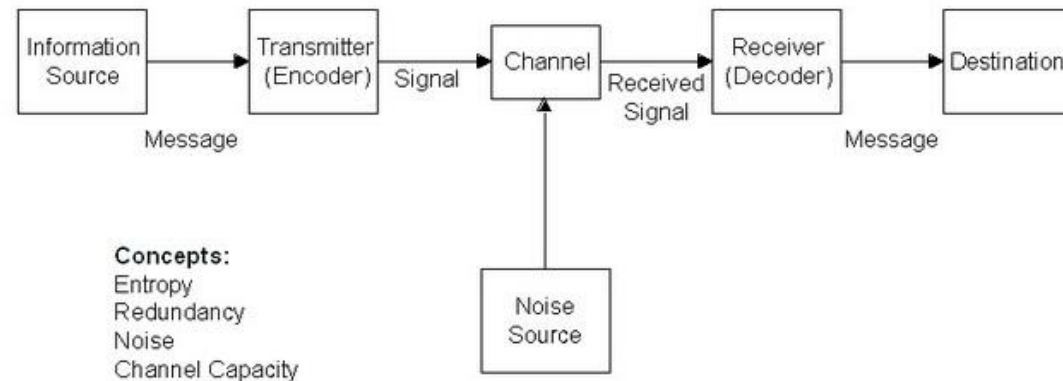
A csatorna kihívásai (2)

Információ átadás biztosítása minden téren

Adat sérülés megelőzése

Pl. Hamming kód

The Shannon-Weaver Mathematical Model, 1949



Hamming-kód (1)

Egyetlen hiba javítására képes kódolás

Motiváció: Ha a csatorna jó minőségű (pl. vezetékes összeköttetés), akkor a többszörös hibák generálásának valószínűsége kicsi. Ezért elegendő minden egyszeres hibát javítani, mert ezek fordulnak elő nagy valószínűséggel.

Több hibát észlel de nem tud javítani

Hamming-kód (2)

Hamming perfekt kódok: $C(n,k)$

$$n = 2^{n-k} - 1 \iff \sum_{i=0}^{n-k} \binom{n-k}{i} = 2^{n-k}$$

Felépítésük:

- 1) a H paritás ellenőrző mátrix oszlopait úgy választjuk meg, hogy mind különböző legyen és a csupa nulla oszlop ne szerepeljen köztük
- 2) meghatározzuk a generátormátrixot
- 3) megtervezzük az illesztő kapukat a szindróma dekódoláshoz
- 4) implementáljuk az egész sémát.

C(7,4) Hamming-kód (1)

$n = 7, k = 4$, tehát $n + 1 = 8 = 2^{7-4}$ teljesül

A paritás ellenőrző mátrix konstrukciója:

$$H = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

A generátormátrix:

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

A számolástól most eltekintünk, jegyzetben megtalálható

C(7,4) Hamming-kód (2)

- **Adatbitek:** d_1, d_2, d_3, d_4
- **Paritásbitek (ellenőrző bitek):** p_1, p_2, p_3

a paritásbiteket mindig a **2 hatványainak pozícióira** tesszük: 1, 2, 4, 8, stb

Elküldött csomag:

$p_1 \ p_2 \ d_1 \ p_3 \ d_2 \ d_3 \ d_4$

C(7,4) Hamming-kód (2)

Paritásbitek kiszámítása:

$$P1 = D1 \oplus D2 \oplus D4$$

$$P2 = D1 \oplus D3 \oplus D4$$

$$P3 = D2 \oplus D3 \oplus D4$$

$$\oplus = \text{XOR}$$

$$- \quad 0 \oplus 0 = 0$$

$$- \quad 1 \oplus 0 = 1$$

$$- \quad 0 \oplus 1 = 1$$

$$- \quad 1 \oplus 1 = 0$$

C(7,4) Hamming-kód példa (1)

Adat: 1 0 1 1

Pozíció	1	2	3	4	5	6	7
Bit	p1	p2	d1	p3	d2	d3	d4

paritásbitek:

$$P1 = D1 \oplus D2 \oplus D4 = 1 \oplus 0 \oplus 1 = 0$$

$$P2 = D1 \oplus D3 \oplus D4 = 1 \oplus 1 \oplus 1 = 1$$

$$P3 = D2 \oplus D3 \oplus D4 = 0 \oplus 1 \oplus 1 = 0$$

Végleges kód:

0 1 1 0 0 1 1

C(7,4) Hamming-kód példa (2)

Eredeti üzenet:

Pozíció	1	2	3	4	5	6	7
Bit	0	1	1	0	0	1	1

Hiba elhelyezése:

Pozíció	1	2	3	4	5	6	7
Bit	0	1	1	0	0	0	1

C(7,4) Hamming-kód példa (3)

Kapott üzenet:

Pozíció	1	2	3	4	5	6	7
Bit	0	1	1	0	0	0	1

Hiba detektálása:

$$P1 \oplus D1 \oplus D2 \oplus D4 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

$$P2 \oplus D1 \oplus D3 \oplus D4 = 1 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$P3 \oplus D2 \oplus D3 \oplus D4 = 0 \oplus 0 \oplus 0 \oplus 1 = 1$$

Hibahely meghatározása:

$P3 \ P2 \ P1 = 1 \ 1 \ 0 = 4 + 2 + 0 = 6 \Rightarrow$ a 6. pozíció a hibás fordítsuk meg

0 1 1 0 0 1 1

C(7,4) Hamming-kód példa (4)

Kapott üzenet 2 hibával:

Pozíció	1	2	3	4	5	6	7
Bit	0	1	1	0	1	0	1

Hiba detektálása:

$$P1 \oplus D1 \oplus D2 \oplus D4 = 0 \oplus 1 \oplus 1 \oplus 1 = 1$$

$$P2 \oplus D1 \oplus D3 \oplus D4 = 1 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$P3 \oplus D2 \oplus D3 \oplus D4 = 0 \oplus 1 \oplus 0 \oplus 1 = 0$$

Hibahely meghatározása:

$P3 \ P2 \ P1 = 0 \ 1 \ 1 = 0 + 2 + 1 = 3 \Rightarrow$ a 3. pozíció a hibás fordítsuk meg

0 1 0 0 1 0 1 \Rightarrow rossz adat lesz

Hamming-kód variációk

Hamming-kód	Adatbitek (k)	Paritásbitek (r)	Teljes hossz (n)	Javítási képesség
(7,4)	4	3	7	1 bit javítható
(15,11)	11	4	15	1 bit javítható
(31,26)	26	5	31	1 bit javítható
(63,57)	57	6	63	1 bit javítható

A Hamming-kód egy fontos kiterjesztése a **SECDED**:

Single Error Correction Double Error Detection

Ez úgy jön létre, hogy a Hamming-kódhoz még egy **globális paritásbitet** adunk hozzá (összes bitre), így:

Pl. **(8,4)** kód → az eredeti (7,4) Hamming + 1 extra paritásbit

Képes 2 bit hibát felismerni, és 1-et javítani

Alkalmazásuk

(7,4) – oktatás, egyszerű RAM védelem

(15,11) – memóriák, hibajavító modulok, kommunikációs protokollok (pl. úrkutatásban is használták)

SECDED – ECC RAM, processzorok, tárolók (BIOS, SSD)

Hamming-kód feladatok

Egy bináris csatorna bit hiba valószínűsége $P_b = 0.001$

- 1) Átküldünk egy 57 bit hosszú üzenetet a csatornán kódolás nélkül. Mennyi a blokk hiba valószínűsége (vagyis annak az esélye, hogy az üzenet hibásan érkezik meg)?

$$(1 - P_b)^{57} \approx 0.9446$$

blokk hiba valószínűsége

$$1 - (1 - P_b)^{57} \approx 0.0554$$

Hamming-kód feladatok

Egy bináris csatorna bit hiba valószínűsége $P_b = 0.001$

2) Átküldünk egy 57 bit hosszú üzenetet a csatornán $C(63,57)$ Hamming-kóddal kódolva. Mennyi a blokk hiba valószínűsége (vagyis annak az esélye, hogy hibás az üzenet dekódolása)?

Hamming kód használata esetén a dekódolás helyes lesz, ha 0 vagy 1 hiba van a vett kódszóban. A helyes dekódolás valószínűsége

Hamming-kód feladatok

Egy bináris csatorna bit hiba valószínűsége $P_b = 0.001$

2) Átküldünk egy 57 bit hosszú üzenetet a csatornán C(63,57) Hamming-kóddal kódolva. Mennyi a blokk hiba valószínűsége (vagyis annak az esélye, hogy hibás az üzenet dekódolása)?

$$(1 - P_b)^{63} + 63(1 - P_b)^{62} * P_b \approx 0.99812$$

blokk hiba valószínűsége

$$1 - (1 - P_b)^{63} + 63(1 - P_b)^{62} * P_b \approx 0.00188$$

Hamming-kód feladatok

$C(63,57)$ Hamming kód használatával csökkentettük a blokk hiba valószínűségét 0.0554-ről 0.00188-re

Ennek ára pedig az, hogy a kód ráta $57/63$, vagyis az effektív csatorna kapacitás az eredeti kapacitás $57/63$ részére csökkent.

Hamming távolság

két azonos hosszúságú bitlánc között **az eltérő bitek száma**

A: 10**1**10**1**1

B: 10**0**10**0**1

Hamming távolság $d = 2$

Maximálisan javítható hiba: $t = \frac{d - 1}{2}$

Maximálisan detektálható hiba: $d - 1$

Milyen kódolást használjak?

- Mekkora az adatblokkom (hány bit)?
- Hány hibát akarok biztosan kijavítani vagy legalább észrevenni?
- Mekkora a maximális redundancia, amit el tudok viselni?
- Mekkora az átviteli vagy tárolási hibák valószínűsége?
- Kell-e a hibajavítás valós időben?

Reed-Solomon (1)

- nem bitenként, hanem **szimbólumonként** dolgozik (pl. 1 szimbólum = 1 byte = 8 bit)
- képes egyszerre **több hiba javítására** – akár **több egymást követő hibás szimbólumot** is
- **Javít és detektál is**, anélkül hogy újraküldésre lenne szükség

Reed-Solomon (2)

k = eredeti adat szimbólumok száma

n = teljes kódolt szimbólumok száma

n – k = redundáns szimbólumok száma (hibajavításra szolgál)

t szimbólum javítására képes

$$t = \frac{n - k}{2}$$

RS(255,223)

223 byte adat, 32 byte hibajavító kód = 255 byte adat küldve

16 byte hibás adat javítása lehetséges

Reed-Solomon vs. Hamming kód

	Hamming-kód	Reed–Solomon-kód
Bit vagy szimbólum	Biteken dolgozik	Szimbólumokon (ált. byte)
Hibatípus	Egyedi bit	Szimbólumhibák / blokkok
Javítási képesség	1 bit	t szimbólumhibát is javít
Használat	RAM, egyszerű rendszerek	QR-kód, DVD, úrtávközlés, mobil
Rugalmasság	Alacsony	Magas (paraméterezhető)

Reed-Solomon működése

- Mi lenne, ha egy QR-kódban 15 négyzet elmaszatólódna – szerintetek még olvasható maradna?

Igen, mert RS kódoslást használ

Reed-Solomon kódolása (1)

Adat = [15, 22, 7]

$$P(x) = a^0 + a_1 x^1 + a_2 x^2$$

$$P(x) = 15 + 22x + 7x^2$$

(lehet fordítva is)

Kiértékeljük a polinomot több x értékre pl. $X = 1, 2, 3, 4, 5, 6$

x	$P(x) = 15 + 22x + 7x^2$
1	$15 + 22 \times 1 + 7 \times 1 = 44$
2	$15 + 44 + 28 = 87$
3	$15 + 66 + 63 = 144$
4	$15 + 88 + 112 = 215$
5	$15 + 110 + 175 = 300$
6	$15 + 132 + 252 = 399$

Reed-Solomon kódolása (2)

x	P(x)	P(x) mod 256
1	44	44
2	87	87
3	144	144
4	215	215
5	300	44
6	399	143

Mivel az RS egy véges testben dolgozik ezért (pl $GF(256)$ – 0-255 közötti érték) minden eredmény **mod(256)**-al kell venni

Reed-Solomon kódolása (3)

Elküldött adat = [44, 87, 144, 215, 44, 143]

- A vevő tudja, hogy a Reed–Solomon kódolás $k = 3$ adatpontból állt
- Ezért a polinom **legfeljebb fokszáma** = $k - 1 = 2$

3 ép pont elég lenne az egyértelmű visszafejtéshez

Ha van 4–5–6 pont, az segít az esetleges hibák javításában

$$P(x) = a^0 + a_1 x^1 + a_2 x^2$$

Jelen esetben 6 egyenletből pl Lagrange interpolációval visszafejti, hogy

$$P(x) = 15 + 22x + 7x^2$$

Egy ilyen 6-os kódolással 2 hiba egyértelműen javítható

Reed-Solomon hatékonysága

Hibajavítási cél	Adatszimbólumok	Hibajavító szimbólumok	Összesen (n)	RS(n,k)
1 hiba javítása	3	2	5	RS(5,3)
2 hiba javítása	4	4	8	RS(8,4)
3 hiba javítása	10	6	16	RS(16,10)
16 hiba javítása	223	32	255	RS(255,223)

Ha valakit érdekel több infó: <https://www.pclviewer.com/rs2/calculator.html>

Reed-Solomon használata

Ma is használt eljárás leginkább ott ahol

- Fizikai használatból adódó sérülések léphetnek fel
- Nem lehet az adatot újraküldeni
- Részben streaming

REED-SOLOMON KÓDOLÁS: FELHASZNÁLÁSI TERÜLETEK



QR-kódok

Hiányzó, sérült adat helyreállítása



CD / DVD / Blu-ray

Fizikai sérülések (karcolás, por) ellen



Űrkommunikáció

Nem lehet újraküldeni



Mobilhálózat

Radió zaj, interferencia elleni védelem



SSD / NAND memória

Elhasználódó cellák javítása



Digitális TV




Kép/hang hibátlan lejátszása gyenge jel mellett



RAID, adatmentés

Több elem kiesését is elviseli

Miről lesz szó

-  **Információelmélet alapfogalmak** – Shannon, entropia, tömörítés
-  **Klasszikus kriptográfia** – Caesar, Vigenère, Enigma
-  **Hibajavítás és csatornamodellek** – Hamming-kód, Reed–Solomon
- **Modern kriptográfia** – AES, RSA, hash-ek, digitális aláírás
- **Hálózati biztonság** – SSL, TLS, VPN, MITM támadások
- **Adatvédelem, GDPR** – anonimizálás, nyomkövetés, metaadat
- **Támadások és védekezés** – brute force, phishing, ransomware, sql injection
- **Jövő és kutatási irányok** – kvantumkriptográfia, AI biztonság

Gyakorlati feladatok

AES