

Algebra3, alkalmazott matematikus

ELTE Algebra és Számelmélet Tanszék

Előadó: Kiss Emil

<http://ewkiss.web.elte.hu/wp/wordpress>

ewkiss@gmail.com

10/11. előadás

A kódok típusai

Kódolás: adatok megváltoztatása.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).
A megváltoztatott adat illetéktelenek által nem olvasható.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).

A megváltoztatott adat illetéktelenek által nem olvasható.

Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímek szorzatára.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).

A megváltoztatott adat illetéktelenek által nem olvasható. Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímek szorzatára.

- **Forráskódolás:** adatok tömörítése.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).
A megváltoztatott adat illetéktelenek által nem olvasható.
Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímek szorzatára.
- **Forráskódolás:** adatok tömörítése.
Kevesebb tárolóhely,

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).
A megváltoztatott adat illetéktelenek által nem olvasható.
Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímek szorzatára.
- **Forráskódolás:** adatok tömörítése.
Kevesebb tárolóhely, gyorsabb adattovábbítás.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).
A megváltoztatott adat illetéktelenek által nem olvasható.
Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímelek szorzatára.
- **Forráskódolás:** adatok tömörítése.
Kevesebb tárolóhely, gyorsabb adattovábbítás.
- **Hibajelző és hibajavító kódok.**

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).
A megváltoztatott adat illetéktelenek által nem olvasható.
Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímelek szorzatára.
- **Forráskódolás:** adatok tömörítése.
Kevesebb tárolóhely, gyorsabb adattovábbítás.
- **Hibajelző és hibajavító kódok.**
A megváltoztatott adatot zajos „csatornán” továbbítjuk.

A kódok típusai

Kódolás: adatok megváltoztatása.

Dekódolás: a megváltoztatott adatból az eredeti visszanyerése.

Célok

- **Titkosítás** (kriptográfia).
A megváltoztatott adat illetéktelenek által nem olvasható.
Például az **RSA-módszer** azt az elvet alkalmazza, hogy nagy számok nem bonthatók gyorsan prímek szorzatára.
- **Forráskódolás:** adatok tömörítése.
Kevesebb tárolóhely, gyorsabb adattovábbítás.
- **Hibajelző és hibajavító kódok.**
A megváltoztatott adatot zajos „csatornán” továbbítjuk.
A címzett mégis képes lehet visszaállítani az eredetit.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A kódolás szempontjai

- Minél több hiba felismerhető/javítható legyen.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A kódolás szempontjai

- Minél több hiba felismerhető/javítható legyen.
- Mégis, minél kevésbé hosszabbodjon meg az üzenet.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A kódolás szempontjai

- Minél több hiba felismerhető/javítható legyen.
- Mégis, minél kevésbé hosszabbodjon meg az üzenet.
- Elegendően gyors kódolás/dekódolás.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A kódolás szempontjai

- Minél több hiba felismerhető/javítható legyen.
- Mégis, minél kevésbé hosszabbodjon meg az üzenet.
- Elegendően gyors kódolás/dekódolás.
- A csatorna tipikus hibáinak a jellege.

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A kódolás szempontjai

- Minél több hiba felismerhető/javítható legyen.
- Mégis, minél kevésbé hosszabbodjon meg az üzenet.
- Elegendően gyors kódolás/dekódolás.
- A csatorna tipikus hibáinak a jellege.
Például betűcsere;

A hibajavító kódok alkalmazási területei

- Adattárolás merevlemezen, kompakt lemezen, SSD-n, melyek egyes részei meghibásodhatnak.
- Egy űrszonda elküldi a képeket, mérési adatokat.
- Műsorszórás (műholdról; az interneten át).
- Mobiltelefonos, internetes kommunikáció, adatátvitel.

A kódolás szempontjai

- Minél több hiba felismerhető/javítható legyen.
- Mégis, minél kevésbé hosszabbodjon meg az üzenet.
- Elegendően gyors kódolás/dekódolás.
- A csatorna tipikus hibáinak a jellege.
Például betűcsere; sok egymás melletti betű hibája.

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 0

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 0 kódolva 000

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 00 kódolva 000

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 00 kódolva 000000

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001** kódolva **000000**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001** kódolva **000000111**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **0010** kódolva **000000111**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **0010** kódolva **000000111000**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **00101** kódolva **000000111000**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **00101** kódolva **000000111000111**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **000000111000111**

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **000000111000111111**.

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **000000111000111111**.

Dekódolás: többségi alapon:

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **000000111000111111**.

Dekódolás: többségi alapon: **aaa, aab, aba, baa** \mapsto **a**.

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **0000001111000111111**.

Dekódolás: többségi alapon: **aaa, aab, aba, baa** \mapsto **a**.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás,

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **0000001111000111111**.

Dekódolás: többségi alapon: **aaa, aab, aba, baa** \mapsto **a**.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például **001011** kódolva **0000001111000111111**.

Dekódolás: többségi alapon: **aaa, aab, aba, baa** \mapsto **a**.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: *aaa, aab, aba, baa* $\mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón,

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón, sercenés rádió vételkor,

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón, sercenés rádió vételkor, neve **csomós hiba**,

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón, sercenés rádió vételkor, neve **csomós hiba**, angolul **burst error**),

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón, sercenés rádió vételkor, neve **csomós hiba**, angolul **burst error**), akkor érdemes a betűket még össze is keverni

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón, sercenés rádió vételkor, neve **csomós hiba**, angolul **burst error**), akkor érdemes a betűket még össze is keverni (**kódátfüzés**).

Háromszorozás

9.1.2. Példa

Minden betűt háromszor egymás után elküldünk.

Például 001011 kódolva 000000111000111111.

Dekódolás: többségi alapon: $aaa, aab, aba, baa \mapsto a$.

Elemzés

Ha bármely három szomszédos betűből legfeljebb **egy** hibás, akkor az eredeti üzenet visszakapható (**1-hibajavító kód**).

Az üzenet **háromszorosára** nyúlik.

Ha szomszédos bitek hajlamosak egyszerre meghibásodni (karcolás az adathordozón, sercenés rádió vételkor, neve **csomós hiba**, angolul **burst error**),

akkor érdemes a betűket még össze is keverni (**kódátfüzés**).

Ha betű kimaradhat, akkor szinkronjelek is kellhetnek.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q ,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$, $k = 1$,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$, $k = 1$, $n = 3$,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi : Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$, $k = 1$, $n = 3$, $\varphi(x) = xxx$,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi: Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$, $k = 1$, $n = 3$, $\varphi(x) = xxx$,
 $C = \{000, 111\} \subseteq Q^3$.

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi: Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$, $k = 1$, $n = 3$, $\varphi(x) = xxx$,
 $C = \{000, 111\} \subseteq Q^3$. Ez egy 3 hosszú,

Alapfogalmak, jelölések

9.1.1. Definíció

A betűk halmaza Q , ez az **ábécé**, elemszáma q .

A Q elemeiből készített k hosszú sorozatok: **szavak**.

Halmazuk Q^k (a betűket vessző nélkül egymás mellé írjuk).

Kódolás: $\varphi: Q^k \rightarrow Q^n$ injektív függvény.

$\varphi(Q^k) = C \subseteq Q^n$ a φ értékkészlete, elemei a **kódszavak**.

A $C \subseteq Q^n$ egy (n, k) **paraméterű kód**. Az n a kód **hossza**.

A kód megadásakor sokszor csak a C halmaz szerepel,
a φ kódoló függvény nem.

Példa

A **háromszorozásnál** $Q = \{0, 1\}$, $k = 1$, $n = 3$, $\varphi(x) = xxx$,
 $C = \{000, 111\} \subseteq Q^3$. Ez egy 3 hosszú, $(3, 1)$ paraméterű kód.

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**,

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk
(ezek a helyek mások lehetnek v , mint w esetében),

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk
(ezek a helyek mások lehetnek v , mint w esetében),
akkor nem kaphatjuk Q^n -nek ugyanazt az elemét.

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk
(ezek a helyek mások lehetnek v , mint w esetében),
akkor nem kaphatjuk Q^n -nek ugyanazt az elemét.

Például a háromszorozó kód **1**-hibajavító

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk
(ezek a helyek mások lehetnek v , mint w esetében),
akkor nem kaphatjuk Q^n -nek ugyanazt az elemét.

Például a háromszorozó kód **1**-hibajavító és **2**-hibajelző.

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk
(ezek a helyek mások lehetnek v , mint w esetében),
akkor nem kaphatjuk Q^n -nek ugyanazt az elemét.

Például a háromszorozó kód **1**-hibajavító és **2**-hibajelző.
Ha **2** helyen megváltozik **000**, akkor az nem kódszó.

Hibajelzés, hibajavítás

Érkezik egy $u \in Q^n$, ami nem kódszó (és így hiba történt).
Keresünk egy kódszót, amitől u a **legkevesebb helyen** tér el.

9.1.3. Definíció

Legyen $t \geq 1$ egész szám. A $C \subseteq Q^n$ kód **t -hibajelző**,
ha egy kódszót legfeljebb t helyen megváltoztatva
az eredmény nem lehet kódszó.

A C kód **t -hibajavító**, ha bárhogy veszünk két $v \neq w$ kódszót,
ha v -t is és w -t is legfeljebb t helyen megváltoztatjuk
(ezek a helyek mások lehetnek v , mint w esetében),
akkor nem kaphatjuk Q^n -nek ugyanazt az elemét.

Például a háromszorozó kód **1**-hibajavító és **2**-hibajelző.

Ha **2** helyen megváltozik **000**, akkor az nem kódszó.

Ha **1** helyen változik, akkor rekonstruálható az eredeti.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(\underline{00}1011\underline{00}10, 0\underline{1}1011\underline{1000}) = 3$.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(\underline{00}1011\underline{00}1\underline{0}, 0\underline{1}1011\underline{1000}) = 3$.

A $C \subseteq Q^n$ kód **minimális távolsága** a különböző kódszavak Hamming-távolságainak minimuma.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(\underline{00}1011\underline{00}10, 0\underline{1}1011\underline{1000}) = 3$.

A $C \subseteq Q^n$ kód **minimális távolsága** a különböző kódszavak Hamming-távolságainak minimuma.
Vagyis két legközelebbi kódszó távolsága.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(\underline{00}1011\underline{00}1\underline{0}, 0\underline{1}1011\underline{1000}) = 3$.

A $C \subseteq Q^n$ kód **minimális távolsága** a különböző kódszavak Hamming-távolságainak minimuma.

Vagyis két legközelebbi kódszó távolsága. **Jele:** $d(C)$.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(\underline{00}1011\underline{00}10, 0\underline{1}1011\underline{1000}) = 3$.

A $C \subseteq Q^n$ kód **minimális távolsága** a különböző kódszavak Hamming-távolságainak minimuma.

Vagyis két legközelebbi kódszó távolsága. **Jele:** $d(C)$.

Például a háromszorozó kód minimális távolsága 3.

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(\underline{00}1011\underline{0010}, 0\underline{1}1011\underline{1000}) = 3$.

A $C \subseteq Q^n$ kód **minimális távolsága** a különböző kódszavak Hamming-távolságainak minimuma.

Vagyis két legközelebbi kódszó távolsága. **Jele:** $d(C)$.

Például a háromszorozó kód minimális távolsága 3.

9.1.6. Gyakorlat (HF)

A C kód pontosan akkor t -hibajelző, ha $t < d(C)$,

Hamming-távolság

9.1.4. Definíció

A $v, w \in Q^n$ **Hamming-távolsága** azoknak a koordinátáknak a száma, ahol a két szó eltér. **Jele:** $d(v, w)$.

Például $d(00\underline{1}011\underline{0}0\underline{1}0, 0\underline{1}1011\underline{1}0\underline{0}0) = 3$.

A $C \subseteq Q^n$ kód **minimális távolsága** a különböző kódszavak Hamming-távolságainak minimuma.

Vagyis két legközelebbi kódszó távolsága. **Jele:** $d(C)$.

Például a háromszorozó kód minimális távolsága 3.

9.1.6. Gyakorlat (HF)

A C kód pontosan akkor t -hibajelző, ha $t < d(C)$,
és pontosan akkor t -hibajavító, ha $2t < d(C)$.

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú,

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson,

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet,

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1},$$

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Bizonyítás

Minden v kódszót változtassunk meg az első $d - 1$ helyen

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Bizonyítás

Minden v kódszót változtassunk meg az első $d - 1$ helyen minden lehetséges módon.

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Bizonyítás

Minden v kódszót változtassunk meg az első $d - 1$ helyen minden lehetséges módon. Ekkor páronként diszjunkt halmazokat kapunk,

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Bizonyítás

Minden v kódszót változtassunk meg az első $d - 1$ helyen minden lehetséges módon. Ekkor páronként diszjunkt halmazokat kapunk, mert ha v és w egy-egy megváltoztatottja egyenlő lenne,

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Bizonyítás

Minden v kódszót változtassunk meg az első $d - 1$ helyen minden lehetséges módon. Ekkor páronként diszjunkt halmazokat kapunk, mert ha v és w egy-egy megváltoztatottja egyenlő lenne, akkor v és w csak $d - 1$ helyen térhetne el.

Korlátok

$C \subseteq Q^n$ egy d minimális távolságú, (n, k) paraméterű kód.

Ellentmondó követelmények:

- Minél több hibát javítson, azaz d nagy legyen.
- Minél kevésbé nőjön az üzenet, azaz $n - k$ kicsi legyen.

9.1.9. Singleton-korlát

$$q^{n-k} = \frac{q^n}{|C|} \geq q^{d-1}, \text{ azaz } n - k \geq d - 1.$$

Bizonyítás

Minden v kódszót változtassunk meg az első $d - 1$ helyen minden lehetséges módon. Ekkor páronként diszjunkt halmazokat kapunk, mert ha v és w egy-egy megváltoztatottja egyenlő lenne, akkor v és w csak $d - 1$ helyen térhetne el. Így $|C|q^{d-1} \leq q^n$. \square

Perfekt kódok

9.1.7. Hamming-korlát

$$\text{Ha } 2t < d, \text{ akkor } q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$

Perfekt kódok

9.1.7. Hamming-korlát

Ha $2t < d$, akkor $q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i$.

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$.

Perfekt kódok

9.1.7. Hamming-korlát

Ha $2t < d$, akkor $q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i$.

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen.

Perfekt kódok

9.1.7. Hamming-korlát

$$\text{Ha } 2t < d, \text{ akkor } q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen.

A kapott halmazok $2t < d$ miatt páronként diszjunktak lesznek. \square

Perfekt kódok

9.1.7. Hamming-korlát

$$\text{Ha } 2t < d, \text{ akkor } q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen.

A kapott halmazok $2t < d$ miatt páronként diszjunktak lesznek. \square

Perfekt kód: egyenlőség áll,

Perfekt kódok

9.1.7. Hamming-korlát

$$\text{Ha } 2t < d, \text{ akkor } q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i.$$

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen. A kapott halmazok $2t < d$ miatt páronként diszjunktak lesznek. \square

Perfekt kód: egyenlőség áll, azaz minden $u \in Q^n$ szóhoz van tőle legfeljebb t Hamming-távolságra eső kódszó.

Perfekt kódok

9.1.7. Hamming-korlát

Ha $2t < d$, akkor $q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i$.

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen. A kapott halmazok $2t < d$ miatt páronként diszjunktak lesznek. \square

Perfekt kód: egyenlőség áll, azaz minden $u \in Q^n$ szóhoz van tőle legfeljebb t Hamming-távolságra eső kódszó. Például a korábban látott „háromszorozó” kód perfekt.

Perfekt kódok

9.1.7. Hamming-korlát

Ha $2t < d$, akkor $q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i$.

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen.

A kapott halmazok $2t < d$ miatt páronként diszjunktak lesznek. \square

Perfekt kód: egyenlőség áll, azaz minden $u \in Q^n$ szóhoz van tőle legfeljebb t Hamming-távolságra eső kódszó.

Például a korábban látott „háromszorozó” kód perfekt.

A **Golay-kódok** perfektek (lásd 9.4.7. Definíció).

Perfekt kódok

9.1.7. Hamming-korlát

Ha $2t < d$, akkor $q^{n-k} = \frac{q^n}{|C|} \geq \sum_{i=0}^t \binom{n}{i} (q-1)^i$.

Bizonyítás

Legyen w kódszó és $0 \leq i \leq t$. Válasszunk ki i koordinátát, és ezeken a helyeken változtassuk meg w -t tetszőlegesen.

A kapott halmazok $2t < d$ miatt páronként diszjunktak lesznek. \square

Perfekt kód: egyenlőség áll, azaz minden $u \in Q^n$ szóhoz van tőle legfeljebb t Hamming-távolságra eső kódszó.

Például a korábban látott „háromszorozó” kód perfekt.

A **Golay-kódok** perfektek (lásd 9.4.7. Definíció).

Elég, ha egy kód csak „közel perfekt” (de más szempontból jobb.)

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek,

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek,
akkor C **lineáris kód**.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súly**.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súly**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súly**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súly**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súly**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$. Ez a kód **generátormátrixa**.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súlya**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$. Ez a kód **generátormátrixa**.

Szisztematikus kódolás: Gu első k koordinátája u ,

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súlya**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$. Ez a kód **generátormátrixa**.

Szisztematikus kódolás: Gu első k koordinátája u , azaz „ellenőrző betűket” írunk a kódolandó szó után.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súlya**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$. Ez a kód **generátormátrixa**.

Szisztematikus kódolás: Gu első k koordinátája u , azaz „ellenőrző betűket” írunk a kódolandó szó után.
Ilyenkor G első k sora az egységmátrix.

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súlya**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$. Ez a kód **generátormátrixa**.

Szisztematikus kódolás: Gu első k koordinátája u , azaz „ellenőrző betűket” írunk a kódolandó szó után.

Ilyenkor G első k sora az egységmátrix.

Ez nem csorbítja az általánosságot (9.2.4. Gyakorlat).

Lineáris kód

9.2.1. Definíció

Ha Q egy véges test és C altere a Q^n vektortérnek, akkor C **lineáris kód**.

Ilyenkor $d(u, v) = d(u - v, 0)$. A $d(w, 0)$ a w **súlya**.

A kódolandó sorozatokat és a kódszavakat oszlopvektorokba írjuk.

9.2.2. Definíció

Végezzük a kódolást a $G \in Q^{n \times k}$ mátrixszal való szorzással:
 $u \mapsto Gu = v$. Ez a kód **generátormátrixa**.

Szisztematikus kódolás: Gu első k koordinátája u , azaz „ellenőrző betűket” írunk a kódolandó szó után.

Ilyenkor G első k sora az egységmátrix.

Ez nem csorbítja az általánosságot (9.2.4. Gyakorlat).

Példa: A „háromszorozó” kódolás generátormátrixa $(1 \ 1 \ 1)^T$.

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in \mathbb{Q}^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa,

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C ,

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor

$P = [-M \quad E_{n-k}]$ ellenőrző mátrix lesz.

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor

$P = [-M \quad E_{n-k}]$ ellenőrző mátrix lesz. Általában adott G -hez P akkor jó,

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód **(paritás)ellenőrző mátrixa**, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor

$P = [-M \ E_{n-k}]$ ellenőrző mátrix lesz. Általában adott G -hez P akkor jó, ha rangja $n - k$

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor

$P = [-M \ E_{n-k}]$ ellenőrző mátrix lesz. Általában adott G -hez P akkor jó, ha rangja $n - k$ és $PG = 0$ (9.2.6. Gyakorlat, HF).

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in \mathbb{Q}^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor

$P = [-M \ E_{n-k}]$ ellenőrző mátrix lesz. Általában adott G -hez P akkor jó, ha rangja $n - k$ és $PG = 0$ (9.2.6. Gyakorlat, HF).

9.2.7. Állítás

Egy P ellenőrző mátrixú kód minimális távolsága a legkisebb olyan d , melyre P -ben van d összefüggő oszlop.

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor

$P = [-M \ E_{n-k}]$ ellenőrző mátrix lesz. Általában adott G -hez P akkor jó, ha rangja $n - k$ és $PG = 0$ (9.2.6. Gyakorlat, HF).

9.2.7. Állítás

Egy P ellenőrző mátrixú kód minimális távolsága a legkisebb olyan d , melyre P -ben van d összefüggő oszlop.

Bizonyítás: Oszlopok egy rendszere akkor lineárisan összefüggő,

Paritásellenőrző mátrix

9.2.5. Definíció

A $P \in Q^{(n-k) \times n}$ a C lineáris kód (paritás)ellenőrző mátrixa, ha magtere C , vagyis $v \in C \iff Pv = 0$.

Ha egy szisztematikus kód generátormátrixa $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$, akkor $P = [-M \ E_{n-k}]$ ellenőrző mátrix lesz. Általában adott G -hez P akkor jó, ha rangja $n - k$ és $PG = 0$ (9.2.6. Gyakorlat, HF).

9.2.7. Állítás

Egy P ellenőrző mátrixú kód minimális távolsága a legkisebb olyan d , melyre P -ben van d összefüggő oszlop.

Bizonyítás: Oszlopok egy rendszere akkor lineárisan összefüggő, ha van olyan nem nulla kódszó, amelyben a többi oszlopnak megfelelő komponensben nulla áll.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

Mivel az oszlopokat „lenormáltuk”, bármely két oszlop független.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

Mivel az oszlopokat „lenormáltuk”, bármely két oszlop független. Ezért a Hamming-kód minimális távolsága (legalább) 3.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

Mivel az oszlopokat „lenormáltuk”, bármely két oszlop független. Ezért a Hamming-kód minimális távolsága (legalább) 3.

Tehát ez 1-hibajavító kód,

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

Mivel az oszlopokat „lenormáltuk”, bármely két oszlop független. Ezért a Hamming-kód minimális távolsága (legalább) 3.

Tehát ez 1-hibajavító kód, és perfekt is, mert

$$\sum_{i=0}^1 \binom{n}{i} (q-1)^i = 1 + n(q-1) = q^{n-k} = q^m.$$

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

Mivel az oszlopokat „lenormáltuk”, bármely két oszlop független. Ezért a Hamming-kód minimális távolsága (legalább) 3.

Tehát ez 1-hibajavító kód, és perfekt is, mert

$$\sum_{i=0}^1 \binom{n}{i} (q-1)^i = 1 + n(q-1) = q^{n-k} = q^m.$$

Ha $m = q = 2$, akkor a „háromszorozó” kódot kapjuk.

A Hamming-kód

9.2.8. Definíció

Legyen Q véges test és $m \geq 2$. A P mátrix oszlopai legyenek azok a Q^m -beli vektorok, amelyek első nem nulla komponense 1. A **Hamming-kód** a P magtere.

Feltehető, hogy az utolsó m oszlop az egységmátrix.

Az oszlopok száma $n = (q^m - 1)/(q - 1) = 1 + q + \dots + q^{m-1}$.

A magtér (azaz a kód) dimenziója tehát $k = n - m$.

Mivel az oszlopokat „lenormáltuk”, bármely két oszlop független. Ezért a Hamming-kód minimális távolsága (legalább) 3.

Tehát ez 1-hibajavító kód, és perfekt is, mert

$$\sum_{i=0}^1 \binom{n}{i} (q-1)^i = 1 + n(q-1) = q^{n-k} = q^m.$$

Ha $m = q = 2$, akkor a „háromszorozó” kódot kapjuk.

A Singleton-korlátban csak $m = 2$ esetén lesz egyenlőség.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.
Érkezik $v + h$, ahol h a hiba.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.
Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott,

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.
Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $G u = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$$P(v + h) = Ph$$

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$$P(v + h) = Ph \text{ (mert } v \text{ kódszó),}$$

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája)
ismerjük.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$,
generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű
változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája)
ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van. Ezért megvan az i , és Ph első nem nulla komponenseként λ_i .

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gv = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van. Ezért megvan az i , és Ph első nem nulla komponenseként λ_i . Ismerjük tehát a h vektort,

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van. Ezért megvan az i , és Ph első nem nulla komponenseként λ_i . Ismerjük tehát a h vektort, és így $(v + h) - h = v$ -t is.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van. Ezért megvan az i , és Ph első nem nulla komponenseként λ_i .

Ismerjük tehát a h vektort, és így $(v + h) - h = v$ -t is.

A v vektor első $k = n - m$ komponense az u üzenet.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van. Ezért megvan az i , és Ph első nem nulla komponenseként λ_i .

Ismerjük tehát a h vektort, és így $(v + h) - h = v$ -t is.

A v vektor első $k = n - m$ komponense az u üzenet.

Ha több, mint 1 hiba történt, akkor a visszafejtés eredménye hibás.

Dekódolás

Az 1-hibajavító Hamming-kód ellenőrző mátrixa $P = [-M \ E_m]$, generátormátrixa legyen $G = \begin{bmatrix} E_k \\ M \end{bmatrix}$. Az üzenet kódja $Gu = v$.

Érkezik $v + h$, ahol h a hiba. Feltesszük, hogy csak 1 betű változott, azaz h -nak csak az i -edik komponense $\lambda_i \neq 0$.

Hogyan lehet meghatározni az u üzenetet?

$P(v + h) = Ph$ (mert v kódszó), így Ph -t (ez v szindrómája) ismerjük. Ph a P mátrix i -edik p_i oszlopának λ_i -szerese.

A P mátrixnak csak egyetlen Ph -val párhuzamos oszlopa van. Ezért megvan az i , és Ph első nem nulla komponenseként λ_i .

Ismerjük tehát a h vektort, és így $(v + h) - h = v$ -t is.

A v vektor első $k = n - m$ komponense az u üzenet.

Ha több, mint 1 hiba történt, akkor a visszafejtés eredménye hibás.

De ha (1 vagy) 2, akkor azt azért tudjuk, hogy történt hiba.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során
szorzunk.

polinommal

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g generátorú polinomkód.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g generátorú polinomkód.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris,

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g **generátorú polinomkód**.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g generátorú polinomkód.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

Példa

$Q = \{0, 1\}$ a kételemű test

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g **generátorú polinomkód**.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

Példa

$Q = \{0, 1\}$ a kételemű test és $g(x) = x^2 + x + 1$.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g generátorú polinomkód.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

Példa

$Q = \{0, 1\}$ a kételemű test és $g(x) = x^2 + x + 1$.

Ekkor $k = 1$, $n = 3$ esetén a háromszorozó kódolást kapjuk.

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g **generátorú polinomkód**.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

Példa

$Q = \{0, 1\}$ a kételemű test és $g(x) = x^2 + x + 1$.

Ekkor $k = 1$, $n = 3$ esetén a háromszorozó kódolást kapjuk.

Akkor is, ha Q tetszőleges test:

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g **generátorú polinomkód**.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

Példa

$Q = \{0, 1\}$ a kételemű test és $g(x) = x^2 + x + 1$.

Ekkor $k = 1$, $n = 3$ esetén a háromszorozó kódolást kapjuk.

Akkor is, ha Q tetszőleges test: ha $u \in Q$ konstans polinom,

A polinomkód fogalma

9.3.1. Definíció

Legyen Q egy véges test. Az $u = u_1 u_2 \dots u_k$ kódolandó szó helyett az $u_1 x^{k-1} + \dots + u_{k-1} x + u_k$ polinomot tekintjük.

A kódolás során (nem generátormátrixszal, hanem) polinommal szorzunk. Legyen $g \in Q[x]$ rögzített, $n - k$ fokú polinom.

$C = \{g(x)u(x) : \text{gr}(u) \leq k\}$ a g **generátorú polinomkód**.

9.3.2. Gyakorlat: Igazoljuk, hogy minden polinomkód lineáris, és írjuk föl az előző kód egy generátormátrixát.

Példa

$Q = \{0, 1\}$ a kételemű test és $g(x) = x^2 + x + 1$.

Ekkor $k = 1$, $n = 3$ esetén a háromszorozó kódolást kapjuk.

Akkor is, ha Q tetszőleges test: ha $u \in Q$ konstans polinom, akkor $g(x)u = ux^2 + ux + u \leftrightarrow uuu$.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a \mathbb{Q} test egy bővítésének, melynek rendje a szorzásra legalább n .

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$,

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3.
Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$,

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$,

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$,

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$, legyen $n = 3$, így $k = 1$.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$, legyen $n = 3$, így $k = 1$. Ez a háromszorozó kód, aminek a minimális távolsága 3.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$, legyen $n = 3$, így $k = 1$. Ez a háromszorozó kód, aminek a minimális távolsága 3. Ha $Q = K$, akkor is a háromszorozó kódot kapjuk, de 4 betűvel.

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$, legyen $n = 3$, így $k = 1$. Ez a háromszorozó kód, aminek a minimális távolsága 3. Ha $Q = K$, akkor is a háromszorozó kódot kapjuk, de 4 betűvel. Bináris üzenetet kétbites részekre vágva kódolhatunk:

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$, legyen $n = 3$, így $k = 1$. Ez a háromszorozó kód, aminek a minimális távolsága 3. Ha $Q = K$, akkor is a háromszorozó kódot kapjuk, de 4 betűvel. Bináris üzenetet kétbites részekre vágva kódolhatunk:
 $0 \leftrightarrow 00$, $1 \leftrightarrow 01$, $\alpha \leftrightarrow 10$, $\beta \leftrightarrow 11$

A minimális távolság becslése

9.3.3. Állítás

Legyen $\alpha \neq 0$ olyan eleme a Q test egy bővítésének, melynek rendje a szorzásra legalább n . Ha $d \leq n$, és egy $n - k$ fokú $g \in Q[x]$ polinomnak gyöke $\alpha, \alpha^2, \dots, \alpha^{d-1}$, akkor a g generátorú polinomkód minimális távolsága legalább d .

9.3.7. Példa

$Q = \mathbb{Z}_2$, $K = \{0, 1, \alpha, \beta\}$ a négyelemű test. Ekkor α rendje 3. Legyen $g(x) = x^2 + x + 1 = (x - \alpha)(x - \beta)$, gyöke $\alpha^2 = \beta$ is. Ezért $d = 3 \leq n$, g foka $2 = n - k$, legyen $n = 3$, így $k = 1$. Ez a háromszorozó kód, aminek a minimális távolsága 3. Ha $Q = K$, akkor is a háromszorozó kódot kapjuk, de 4 betűvel. Bináris üzenetet kétbites részekre vágva kódolhatunk: $0 \leftrightarrow 00$, $1 \leftrightarrow 01$, $\alpha \leftrightarrow 10$, $\beta \leftrightarrow 11$ (azaz $a\alpha + b \leftrightarrow ab$).

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d ,

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$),

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki),

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1+\dots+n_m} \prod_{i>j}(\alpha^{n_i} - \alpha^{n_j})$,

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1+\dots+n_m} \prod_{i>j}(\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in \mathbb{Q}[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1 + \dots + n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n - 1$,

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1 + \dots + n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n - 1$, így minden $n_i < n$.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1 + \dots + n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n - 1$, így minden $n_i < n$. De $o(\alpha) \geq n$,

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1 + \dots + n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n - 1$, így minden $n_i < n$. De $o(\alpha) \geq n$, ezért a szorzat egyik tényezője sem nulla,

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1 + \dots + n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n - 1$, így minden $n_i < n$. De $o(\alpha) \geq n$, ezért a szorzat egyik tényezője sem nulla, azaz $\det(M) \neq 0$.

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1 + \dots + n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n-1$, így minden $n_i < n$. De $o(\alpha) \geq n$, ezért a szorzat egyik tényezője sem nulla, azaz $\det(M) \neq 0$. Ekkor a homogén egyenletrendszernek csak triviális megoldása van,

A becslés bizonyítása

Mivel a kód lineáris, azt kell belátni, hogy a nem nulla kódszavak súlya legalább d , vagyis ha egy $0 \neq f \in Q[x]$ -nek gyöke α^i ($1 \leq i < d$), akkor f -nek legalább d nem nulla együtthatója van. Legyen $f(x) = v_1x^{n_1} + \dots + v_mx^{n_m}$ (a nem nulla együtthatókat írtuk ki), és tegyük föl indirekt, hogy $m < d$. Tehát

$$v_1\alpha^{in_1} + v_2\alpha^{in_2} + \dots + v_m\alpha^{in_m} = 0 \quad (1 \leq i < d).$$

Ez lineáris egyenletrendszer a v_1, \dots, v_m ismeretlenekre.

Vegyük az első m egyenletet, $m < d$ miatt ezt megtehetjük.

Az egyenletrendszer determinánsa $\alpha^{n_1+\dots+n_m} \prod_{i>j} (\alpha^{n_i} - \alpha^{n_j})$, mert ha az i -edik oszlopból α^{n_i} -t kiemelünk minden i -re, akkor Vandermonde-determináns marad.

Mivel f együtthatói kódszót alkotnak, f foka legfeljebb $n-1$, így minden $n_i < n$. De $o(\alpha) \geq n$, ezért a szorzat egyik tényezője sem nulla, azaz $\det(M) \neq 0$. Ekkor a homogén egyenletrendszernek csak triviális megoldása van, vagyis $f = 0$, ellentmondás. □

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében,

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$,

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse,

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$ és így $g(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{d-1})$,

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$ és így $g(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{d-1})$, akkor a **Reed–Solomon-kódot** kapjuk.

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$ és így $g(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{d-1})$, akkor a **Reed–Solomon-kódot** kapjuk.

Mivel $\text{gr}(g) = d - 1$, a Reed–Solomon-kód minimális távolsága d .

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$ és így $g(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{d-1})$, akkor a **Reed–Solomon-kódot** kapjuk.

Mivel $\text{gr}(g) = d - 1$, a Reed–Solomon-kód minimális távolsága d .

A Singleton-korlátban egyenlőség áll: $n - k = d - 1$,

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$ és így $g(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{d-1})$, akkor a **Reed–Solomon-kódot** kapjuk.

Mivel $\text{gr}(g) = d - 1$, a Reed–Solomon-kód minimális távolsága d .

A Singleton-korlátban egyenlőség áll: $n - k = d - 1$,

ezért ez a kód ebből a szempontból optimális.

BCH- és Reed–Solomon-kód

9.3.5. és 9.3.6. Definíció

Legyen $0 \neq \alpha$ legalább n rendű elem Q egy bővítésében, $d \leq n$, és $g(x)$ az $\alpha, \alpha^2, \dots, \alpha^{d-1}$ Q fölötti minimálpolinomjainak legkisebb közös többszöröse, végül $k = n - \text{gr}(g)$.

A g generátorú n hosszú kód neve: **BCH-kód**.

(Bose, Ray-Chaudhuri, Hocquenghem a felfedezők.)

A d szám a kód **tervezett távolsága**.

Ha $\alpha \in Q$ és így $g(x) = (x - \alpha)(x - \alpha^2) \dots (x - \alpha^{d-1})$, akkor a **Reed–Solomon-kódot** kapjuk.

Mivel $\text{gr}(g) = d - 1$, a Reed–Solomon-kód minimális távolsága d .

A Singleton-korlátban egyenlőség áll: $n - k = d - 1$,

ezért ez a kód ebből a szempontból optimális.

Ugyanakkor a betűk száma több, mint a BCH-kód esetében.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test,

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő, ekkor $k = 7 - 3 = 4$.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő, ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő, ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.
Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,
ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.
Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,
ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.
Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő, ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

K elemeit három hosszú 0-1-sorozatok kódolják.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.

Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő, ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

K elemeit három hosszú 0-1-sorozatok kódolják.

A kódszavak hossza $7 \cdot 3 = 21$ bit.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.
Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,
ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

K elemeit három hosszú 0-1-sorozatok kódolják.

A kódszavak hossza $7 \cdot 3 = 21$ bit. A BCH-nál csak 7 bit (ez jobb).

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.
Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,
ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

K elemeit három hosszú 0-1-sorozatok kódolják.

A kódszavak hossza $7 \cdot 3 = 21$ bit. A BCH-nál csak 7 bit (ez jobb).

N bites üzenet kódja a BCH kódnál $(7/4)N$ bites lesz.

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.
Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,
ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

K elemeit három hosszú 0-1-sorozatok kódolják.

A kódszavak hossza $7 \cdot 3 = 21$ bit. A BCH-nál csak 7 bit (ez jobb).

N bites üzenet kódja a BCH kódnál $(7/4)N$ bites lesz.

A Reed–Solomon-kódnál csak $(7/5)N$ bites (ez jobb).

BCH- és Reed–Solomon-kód: Példa

9.3.8. Példa

$Q = \mathbb{Z}_2$, K a nyolcelemű test, $g(x) = x^3 + x + 1$.

Legyen $\alpha \in K$ gyöke g -nek. Ekkor $g(x) = (x - \alpha)(x - \alpha^2)(x - \alpha^4)$.
Mivel K^\times rendje a 7 prím, $o(\alpha) = 7$. Így $n = 7$ és $d = 3$ megfelelő,
ekkor $k = 7 - 3 = 4$. Ez BCH-kód kételemű ábécével.

($d > 3$ esetén $g(x)$ magasabb fokú, így k kisebb lenne).

Ha $Q = K$, akkor $g(x) = (x - \alpha)(x - \alpha^2)$ is megfelelő.

Ez Reed–Solomon-kód, $n = 7$ és $k = 7 - 2 = 5$.

K elemeit három hosszú 0-1-sorozatok kódolják.

A kódszavak hossza $7 \cdot 3 = 21$ bit. A BCH-nál csak 7 bit (ez jobb).

N bites üzenet kódja a BCH kódnál $(7/4)N$ bites lesz.

A Reed–Solomon-kódnál csak $(7/5)N$ bites (ez jobb).

Hogy melyik kód jobb, a csatorna hibáinak jellegétől is függ.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$,

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:
ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban,

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:
ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban,
ha $x^n - 1 \mid p(x)v(x)$.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:
ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban,
ha $x^n - 1 \mid p(x)v(x)$. Nyilván $B : v \mapsto pv \bmod (x^n - 1)$ lineáris.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:
ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban,
ha $x^n - 1 \mid p(x)v(x)$. Nyilván $B : v \mapsto pv \bmod (x^n - 1)$ lineáris.
Ez lehetővé teszi az ellenőrző mátrix felírását.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:
ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban,
ha $x^n - 1 \mid p(x)v(x)$. Nyilván $B : v \mapsto pv \bmod (x^n - 1)$ lineáris.
Ez lehetővé teszi az ellenőrző mátrix felírását.

Szisztematikussá is tehető a kódolás (9.4.4, 9.4.5.).

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz:
ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban,
ha $x^n - 1 \mid p(x)v(x)$. Nyilván $B : v \mapsto pv \bmod (x^n - 1)$ lineáris.
Ez lehetővé teszi az ellenőrző mátrix felírását.

Szisztematikussá is tehető a kódolás (9.4.4, 9.4.5.).

Ez a CRC (Cyclic Redundancy Check, CCITT szabvány).

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz: ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban, ha $x^n - 1 \mid p(x)v(x)$. Nyilván $B : v \mapsto pv \bmod (x^n - 1)$ lineáris. Ez lehetővé teszi az ellenőrző mátrix felírását.

Szisztematikussá is tehető a kódolás (9.4.4, 9.4.5.).

Ez a CRC (Cyclic Redundancy Check, CCITT szabvány).

Merevlemez kontroller: g foka 15.

Ciklikus kódok, CRC

9.4. Szakasz

Ciklikus kód: kódszó ciklikus permutáltja is az. A ciklikus lineáris kódok azok, melyek generátorpolinomja osztója $x^n - 1$ -nek.

Ha a BCH-kódban α rendje pontosan n , akkor a kód ciklikus.

Valóban, ha $\alpha^n = 1$, akkor $m_{\alpha^i}(x) \mid x^n - 1$, ezért $g(x) \mid x^n - 1$.

Legyen $x^n - 1 = g(x)p(x)$, ekkor p ellenőrző polinom lesz: ha $\text{gr}(v) < n$, akkor v pontosan akkor van benne a kódban, ha $x^n - 1 \mid p(x)v(x)$. Nyilván $B : v \mapsto pv \bmod (x^n - 1)$ lineáris. Ez lehetővé teszi az ellenőrző mátrix felírását.

Szisztematikussá is tehető a kódolás (9.4.4, 9.4.5.).

Ez a CRC (Cyclic Redundancy Check, CCITT szabvány).

Merevlemez kontroller: g foka 15.

Ethernet packetek, üvegszál (FDDI), pkzip: g foka 32.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$,

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.
Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$,

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.
Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.
Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.
Ekkor minden betű egy byte, azaz egy 8-bites szó.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$,

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy **2-hibajavító kódot** kapjunk.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$,

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

az n értéke egyszer 28,

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy **2-hibajavító kódot** kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

az n értéke egyszer **28**, egyszer **32**.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy **2-hibajavító kódot** kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

az n értéke egyszer **28**, egyszer **32**.

Mindkétszer a kódátfüzés módszerével is ötvözik.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

az n értéke egyszer 28, egyszer 32.

Mindkétszer a kódátfüzés módszerével is ötvözik.

A végén $8 \leftrightarrow 14$ bites átalakító táblázat.

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

az n értéke egyszer 28, egyszer 32.

Mindkétyszer a kódátfüzés módszerével is ötvözik.

A végén $8 \leftrightarrow 14$ bites átalakító táblázat.

Jobb lejátszónak jobb dekódere lehet

A CD matematikája

9.5. Szakasz

Legyen $m(x) = x^8 + x^4 + x^3 + x^2 + 1 \in \mathbb{Z}_2[x]$, ez irreducibilis.

Ezért $Q = \mathbb{F}_{256} \cong \mathbb{Z}_2[x]/(m)$, legyen $\alpha \in Q$ gyöke m -nek.

Ekkor minden betű egy byte, azaz egy 8-bites szó.

Belátható, hogy m primitív (α generálja Q^\times -et).

A tervezett távolság $d = 5$, hogy 2-hibajavító kódot kapjunk.

Azaz $n - k = 4$, és az α -hoz tartozó Reed–Solomon-kódban

$n \leq o(\alpha) = 255$ lehetséges.

A CD-ken egymás után kétféle kódot is használnak:

az n értéke egyszer 28, egyszer 32.

Mindkétyszer a kódátfüzés módszerével is ötvözik.

A végén $8 \leftrightarrow 14$ bites átalakító táblázat.

Jobb lejátszónak jobb dekódere lehet (több hibát javít).