

VILNIAUS UNIVERSITETAS  
MATEMATIKOS IR INFORMATIKOS FAKULTETAS

## **Kodavimo teorija. Tiesinis kodas**

### **Coding theory. Linear Code**

Kodavimo teorijos praktinis darbas

Atliko:	4 kurso 3 grupės studentas	
	Justas Tvarijonas	(parašas)
Darbo vadovas:	Gintaras Skersys, Asist., Dr.	(parašas)

Vilnius – 2019

## TURINYS

1. ĮVADAS .....	2
1.1. Įgyvendinimas .....	2
1.2. Naudojamos trečiųjų šalių bibliotekos .....	2
1.3. panaudotas laikas.....	2
2. PROGRAMOS APRAŠYMAS .....	3
2.1. Programos paleidimas .....	3
2.2. Programos tekstų failu aprašymas .....	3
2.3. Vartotojo sąsajos aprašymas .....	3
2.3.1. Pagrindinis langas .....	3
2.3.2. Žinutės persiuntimo langas .....	4
2.3.3. Paveikslėlio persiuntimo langas .....	4
2.4. Programiniai sprendimai .....	4
2.4.1. Siuntimas kanalu .....	4
2.4.2. Vektoriaus aprašymas .....	4
2.4.3. Teksto kodavimas/dekodavimas .....	4
2.4.4. Paveikslėlio kodavimas/dekodavimas .....	4
3. ATLIKTI EKSPERIMENTAI .....	5
3.1. Žinutės siuntimas keičiant kodo ilgį .....	5
4. NAUDOTA LITERATŪRA .....	6

# **1. Įvadas**

## **1.1. Įgyvendinimas**

Šiame darbe įgyvendintas kodavimas tiesiniu kodu, jo grandininis dekodavimas. Atiliktos visos užduoties dalys.

## **1.2. Naudojamos trečiųjų šalių bibliotekos**

Naudojamos tik java pakete teikiamos bibliotekos:

- java swing - grafinio interfeiso kūrimui.
- java util - darbui su masyvais ir sąrašais.
- java awt, java io, java imageio - paveikslėlio nuskaitymui.
- java math - naudojama tik BigInteger dalis - naudojamaa konvertuojant bitus į int reikšmę.

## **1.3. panaudotas laikas**

- Gilinimuisi į kodavimo algoritmą ~ 10h
- Projektavimui ~ 2h
- Programavimui ~ 10h (iš jų paveikslėlio apdorojimui ~ 4h)
- Klaidų ieškojimui ir taisymui ~ 4h
- Ataskaitos ruošimui ~ 2h

## 2. Programos aprašymas

### 2.1. Programos paleidimas

Programa paleidžiama paleidus "kodavimas.jar" failą.

### 2.2. Programos tekstų failu aprašymas

- codingtheory\gu\MainScreen - Pagrindinis programos langas, jame suvedami parametrai, galima persiųsti žinutę iš dvejetainio kodo.
- codingtheory\gui\MessageFrame - Tekstinės žinutės persiuntimo langas.
- codingtheory\gui\PhotoFraame - Paveikslėlio persiuntimo langas.
- codingtheory\logic\Channel - Kanalo klasė, ji persiunčia žinutę ją iškraipydama.
- codingtheory\logic\Decoder - Dekodavimo klasė, ją sukuriant nusiunčiama generuojanti matrica, pagal ją ji dekoduoja gautą vektoriu.
- codingtheory\logic\Encoder - Kodavimo klasė, ją sukuriant nusiunčiama generuojanti, pagal ją ji užkoduoja vektoriu.
- codingtheory\logic\Matrix - Matricos klasė, ją galima sukurti perduodant matricą ar jos parametrus. Turi metodus gauti generalinę matricą, bei dauginimą iš vektoriu.
- codingtheory\logic\Util - Klasė be konkrečios paskirties, metodai į ją iškeliami norint išvengti kodo duplikavimo. Ji neturi savo būsenos.
- codingtheory\logic\Vector - vektoriaus klasė, turi 3 pagrindinius parametrus: masyvą, masyvą kaip eilutę bei sveikojo dešimtainio skaičiaus reikšmę, gali būti sukurta perdaavus bet kurią iš šių reikšmių.

### 2.3. Vartotojo sąsajos aprašymas

#### 2.3.1. Pagrindinis langas

Šiame lange suvedami reikiami parametrai, generuojanti matrica yra automatiškai sugeneruojama, jeigu jos laukas yra paliekamas tuščias, kitus atveju reikia suvesti matricą stulpelius atskiriant tarpais, o eilutes naujos eilutės simboliu. Išsaugojus šiuos parametrus automatiškai atsidaro žinutės bei paveikslėlio persiuntimo langai su suvestais parametrais. Pagrindiame lange taip pat galima suvesti žinutę iš dvejetainių simbolių(su arba be tarpų), kurios ilgis lygus k. Įvedus šią žinutę ir klaidos tikimybę(pvz 0.02), bei paspaudus "Siųsti" Atvaizduoja užkoduota kanalu persiunta žinutė kartu su klaidų vektoriumi, prieš tęsiant galima ją paredaguoti, norint dekoduoti spaudžiame "Tęsti".

### 2.3.2. Žinutės persiuntimo langas

Šis langas atsidaro įvedus parametrus pagrindiniame lange. Jame įvedus norimą žinutę ir klaidos tikimybę ir paspaudus "Siųsti" ekrane pamatysime persiūtą žinutę su kodavimo ir be jo.

### 2.3.3. Paveikslėlio persiuntimo langas

Šis langas atsidaro įvedus parametrus pagrindiniame lange. Jama pasirenkame paveikslėlį bmp formatu paspaudus ant "Pasirinkti paveiksleli" mygtuko, įvedame norimą klaidos tikimybę ir spaudžiame "siųsti", ekrane matome paveikslėlį persiunta naudojant kodavimą ir jo nenaudojant (Siunčiant geresnės kokybės paveikslėlį gali tekti palaukti, kadangi jo apdorojimas užtrunka).

## 2.4. Programiniai sprendimai

### 2.4.1. Siuntimas kanalu

Siuntimo kanalu metodas prima masyvą, kurį siunčiame kanalu, o gražina klaidų masyvą, kadangi masyvas yra *reference* tipo, tai pačios žinutės grąžinti nėra butinybės, ji yra pakeičiama kviečiančioje klasėje.

### 2.4.2. Vektoriaus aprašymas

Šioje programoje vektorius susideda iš 4 laukų, jau minėtų masyvo, masyvo kaip eilutės, jo sveikąjo dešimtainio skaičiaus reikšmės bei jo svorio. Pirmieji trys parametrai yra ekvivalentūs, tačiau vistiek kuriami dėl patogesnio naudojimo kode.

### 2.4.3. Teksto kodavimas/dekodavimas

Gautas tekstas yra suskaidomas į baitus, kurie yra konvertuojami į bitus. Šie bitai sujungiami į bendrą eilutę. Šios eilutės ilgis yra patikrinamas ir jeigu trūksta bitų norint ją pilnai persiūsti kanalu, pridedamas atitinkamas kiekis "0", kurie vėliau yra nukerpami. Ši eilutė yra siunčiama atkarpomis po k, ją koduojant, persiunčiant kanalu, dekoduojant ir tada prijungiant prie rezultatų eilutės. gavus pilną rezultato eilutę ji yra skaidoma į bitus po 8, verčiama į skaitinę reikšmę, bei verčiama į simbolį (kadangi java kalboje baitai talpina tik 7 bitus (aštuntasis yra ženkle bitas), jeigu skaičiaus reikšmė yra daugiau negu 127, jis yra invertuojamas).

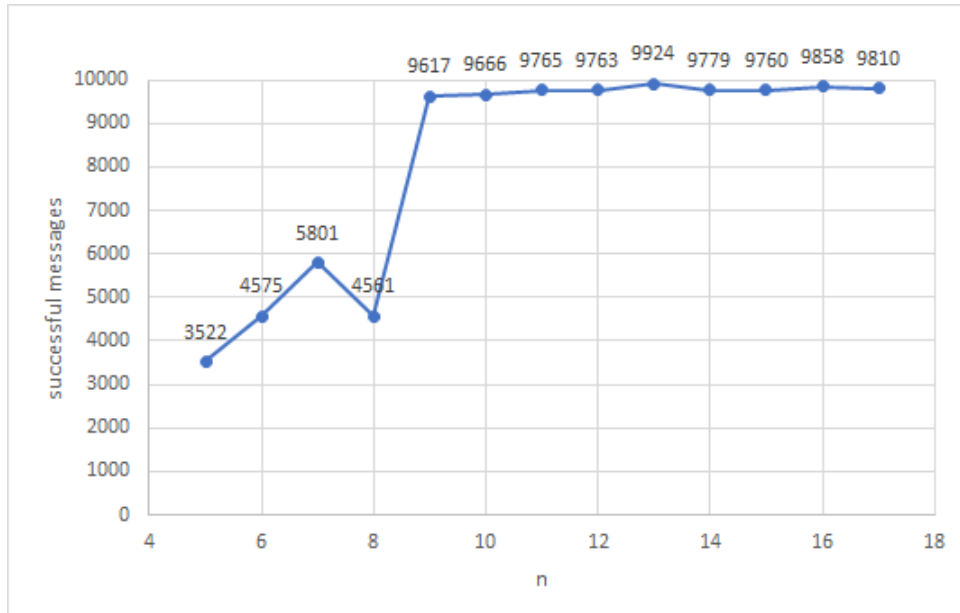
### 2.4.4. Paveikslėlio kodavimas/dekodavimas

Iš gauto paveikslėlio yra ištraukiami visi jo pixeliai (jų yra ilgis x plotis), šie pixeliai yra verčiami į dvejetaines eilutes, atskirai išsaugant, kokia ilgio eilutė buvo gauta kiekvienam bitui, kad vėliau konvertuojant atgal į pixelius nebūtų padaryta klaidų. pixelius pakonvertavus į dvejetainę eilutę jie atitinkamai teksto siuntimui yra taip pat siunčiami, o vėliau naudojanti sukurtu masyvu atverčiami atgal į pixelių masyvą, pagal kurį paveikslėliui nustatomos pixelių reikšmės.

### 3. Atlikti eksperimentai

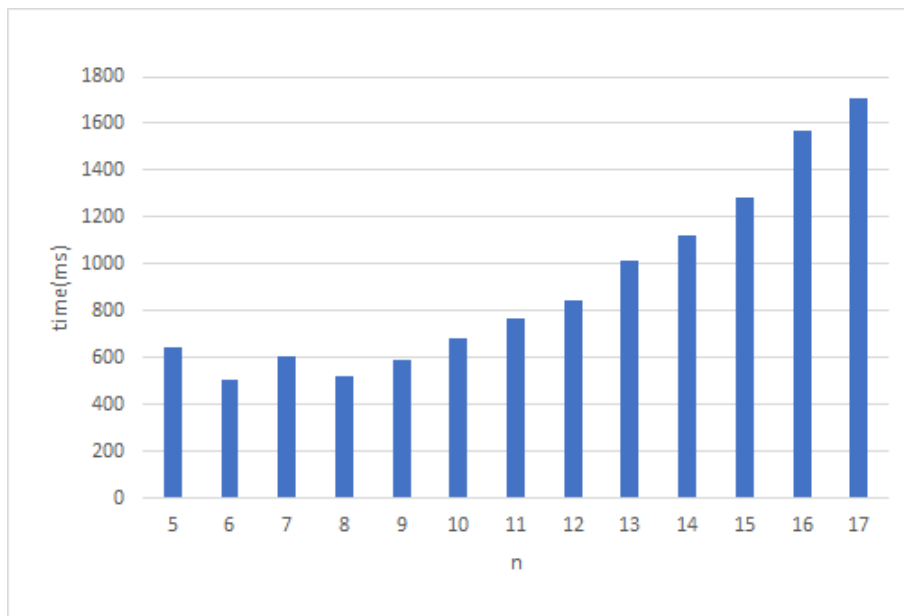
#### 3.1. Žinutės siuntimas keičiant kodo ilgį

Žinutės siuntiams naudojant  $k = 4$ ,  $p = 0.01$  ir keičiant  $n$  reikšmę. Kiekviena kartą žinutė siunčiama 10000 kartų, žiūrima, kiek kartų žinutė nebus pakeista. Žinutės reikšmė - "eksperimentas" tikriname, kad nepasikeitė nei viena raidė. Iš grafiko matome, kad dekodavimo tiksluma žymiai



1 pav. Tikslumas siunčiant kanalu žinutę 10000 kartų keičiant kodo ilgį

nebekinta tuo  $n = 9$ . Atliekant šį eksperimentą taip pat buvo matuojamas laikas, kiek užtrunka išsiųsti šias žinutes 10000 kartų, rezultatai matomi apačioje: Laiko kitimo grafikas,



2 pav. Laikas siunčiant kanalu žinutę 10000 kartų keičiant kodo ilgį

## 4. Naudota literatūra

- <https://klevas.mif.vu.lt/~skersys/19r/ktkt/KTKT.pdf>
- <https://klevas.mif.vu.lt/~skersys/doc/ktkt/literatura21.pdf>
- [https://en.wikipedia.org/wiki/Standard\\_array](https://en.wikipedia.org/wiki/Standard_array)
- [https://en.wikipedia.org/wiki/Parity-check\\_matrix](https://en.wikipedia.org/wiki/Parity-check_matrix)