

## A5 Ataskaita

### 1. Realizuotos užduoties dalys

Realizuotos visos užduoties dalys.

### 2. Programos paleidimas

Programa galima paleisti su A5.jar, nereikalauja jokių specialių papildomų priemonių.

### 3. Programos failai

src/main/java/com/a5coding/calculations/Channel.java	Realizuotas vektoriaus siuntimas kanalu
src/main/java/com/a5coding/calculations/Decoder.java	Realizuotas vektoriaus dekodavimas
src/main/java/com/a5coding/calculations/Encoder.java	Realizuotas vektoriaus kodavimas
src/main/java/com/a5coding/data/DataHandler.java	Abstrakti klasė, kurią paveldi TextHandler ir ImageHandler klasės
src/main/java/com/a5coding/data/TextHandler.java	Atliekamas teksto siuntimas, kodavimas ir dekodavimas
src/main/java/com/a5coding/data/ImageHandler.java	Atliekamas .bmp formato paveiksluko siuntimas, kodavimas ir dekodavimas
src/main/java/com/a5coding/data/image/ImageData.java	Klasė skirta paveiksluko duomenims
src/main/java/com/a5coding/data/image/Header.java	Klasė skirta paveiksluko atrastės duomenims
src/main/java/com/a5coding/io/Main.java	Pagrindinis programos failas
src/main/java/com/a5coding/io/Validator.java	Klasė skirta duomenų validavimui
src/main/java/com/a5coding/io/Converter.java	Klasė skirta duomenų konvertavimui
src/main/resources/com/a5coding/mainScene.fxml	UI pradinis (titulinis) programos failas, skirtas

	gauti pradinės programos reikšmes
src/main/resources/com/a5coding/encodeScene.fxml	UI failas skirtas siunčiamo vektoriaus įvesčiai ir užkoduoto vektoriaus atvaizdavimui
src/main/resources/com/a5coding/decodeScene.fxml	UI failas skirtas vektoriaus koregavimui ir dekodavimui
src/main/resources/com/a5coding/textScene.fxml	UI failas skirtas teksto siuntimui, kodavimui, dekodavimui
src/main/resources/com/a5coding/imageScene.fxml	UI failas skirtas paveiksluko siuntimui, kodavimui, dekodavimui
src/main/java/com/a5coding/controllers/MainController.java	Controller klasė mainScene.fxml
src/main/java/com/a5coding/controllers/EncodeController.java	Controller klasė encodeScene.fxml
src/main/java/com/a5coding/controllers/DecodeController.java	Controller klasė decodeScene.fxml
src/main/java/com/a5coding/controllers/TextController.java	Controller klasė textScene.fxml
src/main/java/com/a5coding/controllers/ImageController.java	Controller klasė imageScene.fxml

#### 4. Vartotojo sąsajos aprašymas

- Pradiniame lange reikia suvesti  $m - RM(1,m)$  parametro ir  $p$  – klaidos tikimybės reikšmes. Galimos reikšmės:  $m \geq 1, 0 \leq p \leq 1$ . Klaidos tikimybę galima įvesti ir su tašku ir su kableliu.
- Taip pat reikia pasirinkti siunčiamų duomenų tipą ir spausti mygtuką „Next“.

Reed Muller's code

**Decoding Reed-Muller's code RM(1, M)  
with Fast Hadamard Transform**

m:

p:

Input type: ☒ Vector  
☐ Text  
☐ Image

Next

- Pasirinkus vektorių kaip duomenų tipą, reikia suvesti nurodyto dydžio pranešimą sudarytą tik iš 0 ir 1, nes programos baigtinis kūnas  $q=2$ .
- Paspaudus mygtuką „Encode“ parodomas užkoduotas pranešimas, kurį toliau galima siųsti paspaudus mygtuką „Send“.

Reed Muller's code

Enter message of size 5:

Message:

Encode

Encoded message:

Send

- Gautą pranešimą galima redaguoti paspaudus mygtuką „Edit“ prieš tai jau atlikus visus norimus pakeitimus. Klaidos ir jų pozicijos taip pat atsinaujina pagal redaguotą pranešimą.
- Paspaudus mygtuką „Decode“ gaunamas dekodotas pranešimas. Jeigu buvo dekodota teisingai pranešimas yra žalios spalvos, jeigu ne – raudonos.
- Paspaudus mygtuką „Home“ yra grįžtama į pradinį langą.

Reed Muller's code

Received message:  
1101111101111000

Errors: 5  
Positions: [2, 8, 13, 14, 15]

Edit Decode

Decoded message:  
10000

Home

Reed Muller's code

Received message:  
1100011011011000

Errors: 8  
Positions: [2, 3, 4, 7, 10, 13, 14, 15]

Edit Decode

Decoded message:  
10100

Home

- Pasirinkus tekstinį duomenų tipą, reikia suvesti norimą tekstinį pranešimą į laukelį. Galima rašyti ir keliose eilutėse.
- Paspaudus mygtuką „Send“ galima pamatyti koks pranešimas buvo gautas nenaudojant ir naudojant kodavimą.
- Paspaudus mygtuką „Home“ yra grįžtama į pradinį langą.

Reed Muller's code

Enter text:  
jonas joja ir dainuoja lietuva salis  
gimtoji

Send

Received without encoding:  
jonas zona"!t(Daine/jq ,ytp"uElic  
Gl,j'i

Received with encoding:  
jonas joja ir dainuoja lietuva salis  
giltoji

Home

- Pasirinkus paveiksliuką kaip duomenų tipą reikia spausti mygtuką „Select image“ ir atsidariusiame lange pasirinkti .bmp formato failą. Pasirinktas paveiksliukas yra parodomas ekrane.
- Paspaudus mygtuką „Send“ galima pamatyti koks paveiksliukas buvo gautas nenaudojant ir naudojant kodavimą.
- Paspaudus mygtuką „Home“ yra grįžtama į pradinį langą.



## 5. Programiniai sprendimai

- Vektoriaus siuntimas kanalu – siunčiant vektorių, kiekvienam jo elementui yra sugeneruojamas skaičius iš intervalo  $[0, 1]$ , jei šis skaičius mažesnis už klaidos tikimybę, elementas yra iškraipomas t.y 1 yra pakeičiamas į 0 ir atvirkščiai.
- Veiksmai su tekstu - tekstas yra paverčiamas į bitų seką, o tada suskaidomas į reikiamo ilgio vektorius. Jeigu šios sekos neįmanoma suskaidyti į vienodo ilgio vektorius t.y lieka paskutinis nepilnas vektorius, prie jo galo yra prideda tiek nulių kiek reikia, kad būtų gaunamas reikiamo ilgio vektorius. Gauti suskaidyti veiktoriai yra siunčiami, koduojami, dekoduojami po vieną ir vienas po kito. Jeigu prie paskutinio vektoriaus skaidymo metu buvo prideda papildomų nulių, vektorius dekodavimo metu yra sutrumpinamas tiek elementų kiek nulių buvo pridėta. Galiausiai visi vektoriai vėl yra sujungiami į vieną vektorių. Šis vektorius (bitų seka) yra paverčiamas tekstu.
- Vieksmai su paveiksluku – visu pirma yra išsaugomos paveiksluko antraštės reikšmės: ilgis ir plotis. Tada paveikslukas yra paverčiamas į bitų seką t.y paveiksluko RGB reikšmės yra išsaugomos binariniu formatu. Ši bitų seka yra suskaitodoma į reikiamo ilgio vektorius, kurie koduojami, siunčiami ir dekoduojami taip pat kaip ir veiksmuose su tekstu. Paveikslukas yra rekonstruojamas bitų seką konvertuojant atgal į RGB reikšmes. *Pastaba:* Kadangi programos veikimas gali užtrukti ilgai geriausia yra rinktis mažo dydžio ir mažai spalvotus (paprastesnės grafikos) paveikslukus.

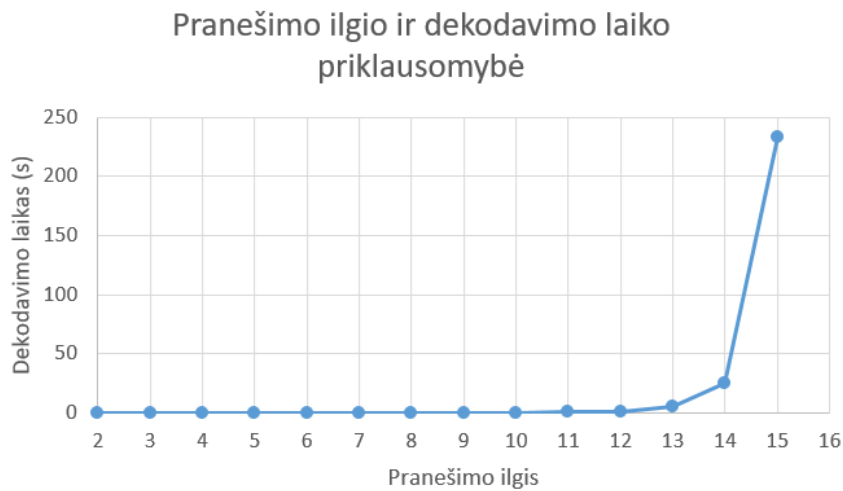
## 6. Eksperimentai

Atliktas eksperimentas, kurio pirma dalis buvo paanalizuoti kaip dekodavimo laikas priklauso nuo užkoduoto pranešimo ilgio (arba kitaip  $m + 1$ , kur  $m = RM(1, m)$  parametras). Klaidos tikimybė buvo palaikoma pastovi  $p = 0.3$ . Kiekvienam pranešimo ilgiui dekodavimo laikas buvo matuojamas 3 kartus, o tada buvo išvedamas vidurkis. Eksperimento duomenys yra pavaizduoti lentelėje 1, o pranešimo ilgio ir dekodavimo laiko priklausomybė – grafike 1.

Antra eksperimento dalis buvo stebėti ne tik vykdymo laiką, bet ir kiek kartų iš 3 bandymų pranešimas buvo teisingai dekodotas. Eksperimento duomenys yra pavaizduoti lentelėje 1 ir grafike 2.

Klaidos tikimybė	Pranešimo ilgis	Dekodavimo laikas (s)			Dekodavimo laiko vidurkis (s)	Teisingų dekodavimų skaičius
0.3	2	0.001	0.001	0.001	0.001	1
0.3	3	0.001	0.001	0.001	0.001	1
0.3	4	0.001	0.001	0.001	0.001	0
0.3	5	0.001	0.001	0.001	0.001	2
0.3	6	0.002	0.002	0.001	0.002	2
0.3	7	0.004	0.004	0.003	0.004	1
0.3	8	0.008	0.006	0.008	0.007	3
0.3	9	0.015	0.014	0.015	0.015	3
0.3	10	0.045	0.043	0.041	0.043	3
0.3	11	0.136	0.137	0.129	0.134	3
0.3	12	0.767	0.768	0.782	0.772	3
0.3	13	4.473	4.473	4.932	4.626	3
0.3	14	23.01	26.02	25.65	24.89	3
0.3	15	225.4	243.7	231.7	233.6	3

Lentelė 1. Eksperimento duomenys



Grafikas 1. Pranešimo ilgio ir dekodavimo laiko priklausomybė

Iš eksperimento duomenų galime matyti, kad nedidelio ilgio pranešimai yra dekoduojami greitai, tačiau dekodavimo laikas pradeda sparčiai didėti didėjant pranešimo ilgiui.

Rydo Miulero kodo  $RM(1,m)$ , kur  $m \geq 2$  ilgis yra  $2^m$ , pavyzdžiui, ilgio 5 pranešimą ( $m = 4$ ) užkoduosime ilgio 16 kodu.  $2^m$  yra eksponentinė funkcija, ką primena ir eksperimento grafikas su didesniais pranešimo ilgiais. Kadangi kodo ilgis didėjant  $m$  labai greitai didėja, taip pat didėja ir dekodavimo laikas.



Grafikas 2. Teisingas dekodavimas

Iš grafiko matome, kad su klaidos tikimybe  $p = 0.3$  didėjant pranešimo ilgiui taip pat didėja ir teisingų dekodavimų skaičius. Tai rodo, kad kuo ilgesnis pranešimas tuo algoritmas veikia efektyviau, tačiau kaip jau matėme grafike 1, ilgesni pranešimai reikalauja didesnių laiko resursų, jų dekodavimo laikas yra didesnis.

## 7. Literatūra

- D.G.Hoffman, D.A.Leonard, C.C.Lindner, K.T.Phelps, C.A.Rodger, J.R.Wall. Coding Theory: The Essentials. Dekker, New York, 1991.  
<https://klevas.mif.vu.lt/~skersys/doc/ktkt/literatura13.pdf>
- Gintaras Skersys "Klaidas taisančių kodų teorija"
- Weisstein, Eric W. "Hadamard Matrix." Text. Wolfram Research, Inc.  
<https://mathworld.wolfram.com/>