

מגישים:

כריסטיאן שקור, 208157826, [christian.s@campus.technion.ac.il](mailto:christian.s@campus.technion.ac.il)

לארין עטאללה, 208653543, [lareine.at@campus.technion.ac.il](mailto:lareine.at@campus.technion.ac.il)

## שאלה 1:

כל ספרה מוכפלת במשקל שלה, וכל הספרות של התוצאות מחוברות יחדיו. אם התוצאה של הכפלת הספרה במשקלה היא בת שתי ספרות, מסוכמת כל אחת משתי הספרות של התוצאה (כלומר המספר הדו-ספרתי מופרד לשני מספרים בעלי ספרה אחת, המחוברות לכל יתר הספרות). את הסכום המתקבל יש להשלים לכפולה הקרובה של 10 (כלפי מעלה).

```

generate_q1.m x checker_q1.m x q3_repeation_code.m x q5_bchmunerr.m x +
1 clear all;
2 close all;
3 clc;
4
5 id = '208653543';
6 Sum = 0;
7
8 %%calculations
9 for i=1:8
10 Sum = Sum + mod(str2num(id(i)) * (2-(mod(i,2))), 10) + idivide(uint32(str2num(id(i)) * (2-(mod(i,2)))),10);
11 end
12
13 checkDigit = mod(10 - mod(Sum,10), 10);

generate_q1.m x checker_q1.m x q3_repeation_code.m x q5_bchmunerr.m x +
1 clear all;
2 close all;
3 clc;
4
5 validate_id('208653543')
6
7 %%check calculations
8 function checkDigit = get_check_digit(id)
9 Sum = 0;
10 for i = 1:8
11 Sum = Sum + mod(str2num(id(i)) * (2-(mod(i,2))), 10) + idivide(uint32(str2num(id(i)) * (2-(mod(i,2)))),10);
12 end
13 checkDigit = mod(10-mod(Sum,10), 10);
14 end
15
16 %%check if equal
17 function isValid = validate_id(id)
18 isValid = get_check_digit(id) == str2num(id(9));
19 end

```

## שאלה 2:

- ספר הקוד זה כל המילים האפשריים מעל האותיות של הקוד בגודל קבוע כלשהו.
- קוד לינארי C הוא קוד מעל שדה סופי F, באורך n חיובי סגור לחיבור וכפל בסקלר  $F: \alpha c_1 + \beta c_2 \in C, \forall c_1, c_2 \in C, \forall \alpha, \beta \in F$ .
- מסקנה: מילת הקוד (0,...,0,0,0) צריכה להיות מוכלת בקוד

- מרחק hamming הוא מרחק המוגדר בין 2 מילות קוד עבור קוד C מעל שדה סופי F עם אורך קוד n. כאשר לכל שני מילות קוד  $x, y \in F_n$  מוגדר את מרחק Hamming ביניהם בתור מספר הקואורדינטות שבהם הם שונים:

$$d_H(x, y) = |\{1 \leq i \leq n : x_i \neq y_i\}|,$$

ולכל מילה  $x = (x_1, x_2, \dots, x_n) \in F_n$  נגדיר את משקל Hamming בתור

$$w_H(x) = d_H(x, 0).$$

- לכל קוד C נגדיר את **מרחק הקוד** בתור:

$$d(C) = \min_{c_1, c_2 \in C, c_1 \neq c_2} d_H(c_1, c_2)$$

קוד עם מרחק d יכול לתקן בוודאות  $(d-1)/2$  שגיאות, ו- $d-1$  מחיקות.

- מרחקי קוד חזרות הוא  $n = d(C_{rep})$ . קוד חזרות הוא קוד לינארי כי הוא סגור לחיבור ולכפל בסקלר.

$$G = (1, 1, 1, \dots, 1) 1 \times n$$

מטריצה מייצרת:

ולכן מימד הקוד הוא 1.

- מרחק קוד זוגיות באורך n הוא  $d(C_{par}) = 2$  והוא לינארי כי הוא סגור לחיבור וכפל בסקלר כי בכל חיבור של 2 קודים זוגיים נקבל קוד זוגי וגם כך בכל הכפלה בסקלר. מימד הקוד הוא n-1

### שאלה 3:

קוד חזרות הוא בעל קצב של  $R = 1/n$  כאשר n הוא אורך הקוד. המפענח ינחש את המספר שמופיע הכי הרבה פעמים ב-n החזרות, ונוכל לתקן עד  $1 - \lfloor n/2 \rfloor$  שגיאות.

נמיר את LAREINE לקוד בינארי:

01001100

01000001

01010010

01000101

01001001

01001110

01000101

השם LAREINE מקודד עם חזרות:

000 111 000 000 111 111 000 000

000 111 000 000 000 000 000 111

000 111 000 111 000 000 111 000

000 111 000 000 000 111 000 111

000 111 000 000 111 000 000 111

000 111 000 000 111 111 111 000

000 111 000 000 000 111 000 111

נמיר את CHRIS לקוד בינארי:

01000011

01001000

01010010

01001001

01010011

השם CHRIS מקודד עם חזרות:

000 111 000 000 000 000 111 111

000 111 000 000 111 000 000 000

000 111 000 111 000 000 111 000

000 111 000 000 111 000 000 111

000 111 000 111 000 000 111 111



```
1 clear all;
2 close all;
3 clc;
4
5 NameinAscii = double('LAREINE')
6 result = [];
7 for i=1:length(NameinAscii)
8     NameinBinary = [zeros(1), de2bi(NameinAscii(i), 'left-msb')];
9     result = [result, repelem(NameinBinary,3)];
10 end
11 result
```

#### **שאלה 4:**

נשתמש המסמך העזר ונחשב את האנרגיה לביט. עבור שידור של  $n$  ביטים במקרה של הערוץ

הגאומי הנתון סך האנרגיה תהיה  $E = n \cdot A^2$ . את האנרגיה הכוללת נוכל לחשב גם כמכפלת מספר הביטים של האינפורמציה באנרגיה לביט

אינפורמציה, כלומר:  $k \cdot E$ . נשווה בין הביטויים ונקבל את האנרגיה לביט אינפורמציה

$$k \cdot E = n \cdot A^2 \rightarrow E = (n/k) \cdot A^2 = (A^2)/R$$

על פי חומר ההכנה:  $N_0 = 2\sigma^2$ , לכן:

פענוח Hard Decision הוא פענוח בו ממירים תחילה את קוד המוצא לערכים בדידים, ורק לאחר מכן מפעילים את המפענח.

פענוח Soft Decision הוא פענוח התלוי באות המוצא הרציף של הערוץ הגאומי ללא ביצוע שינוי באות.

פענוח Soft Decision יהיה בעל הסתברות שגיאה נמוכה יותר, מכיוון שבפענוח Hard Decision ביצוע הדעה-מודולציה גורר איבוד מידע. לכן בפענוח Hard Decision בהכרח נקבל שגיאה גבוהה יותר מאשר בביצוע פענוח ישיר ללא הדעה-מודולציה.

#### **שאלה 5:**

**Bsc:**

ערכי הקבלה של הפונקציה הם:

- סיגנל הקלט הבינארי (data)

- פרמטר ההסתברות של הערוץ הבינארי הסימטרי (probability)

ערכי ההחזרה הם:

- וקטור הפלט לאחר הוספת שגיאות המעבר בערוץ הבינארי הסימטרי (ndata)

- ווקטור בינארי שמייצג את השגיאות שהתרחשו לאחר המעבר בערוץ. הערך 1- מייצג שגיאה

שהתרחשה בביט במיקום התואם, ו-0 מייצג שלא התרחשה בו שגיאה (err)

הפונקציה מדמה מעבר של קלט בינארי בערוץ בינארי סימטרי למעשה.

### Qfunc:

ערך הקבלה הוא סקלר, וקטור או מטריצה.

ערך ההחזרה הוא בהתאמה לערך הקבלה – הפונקציה מחזירה סקלר, וקטור או מטריצה כאשר תחום ערכי המוצא הוא בין 0 ל-1.

הפונקציה מחשבת עבור כל איבר בקלט את ערך פונקציית ה-Q שלו, שהיא בעצם שווה ל- f-1 כאשר f הוא הפילוג הנורמלי.

### Bchnumerr:

מחזירה את כל הקומבינציות האפשריות של אורך ההודעה ואץ מספר השגיאות שאפשר לתקן, כאשר ערך הקבלה הוא קוד BCH באורך N.

## Description

$T = \text{bchnumerr}(N)$  returns all the possible combinations of message length, K, and number of correctable errors, T, for a BCH code of codeword length, N.

generate_q1.m	checker_q1.m	q3_repetition_code.m	q5_bchmunerr.m	+	ans
1	clear all;				3x3 double
2	close all;				
3	clc;				
4					
5	N = 15;				
6	bchnumerr(N);				
7					

	1	2	3
1	15	11	1
2	15	7	2
3	15	5	3
4			
5			