



Факультет программной инженерной и компьютерной техники
Информатика

Лабораторная работа №2
Вариант 83

Преподаватель: Малышева Татьяна Алексеевна
Выполнил: Альхимович Арсений Дмитриевич
Р3110

Санкт-Петербург, 2023

Условие	2
Основное задание.....	3
задание 1 - №10	3
задание 2 - №39	3
задание 3 - №67	4
задание 4 - №79	4
задание 5 - №82	5
задание 6	5
Дополнительное задание	6
Код:.....	6
Вывод программы.....	6
Вывод	7
Список литературы.....	7

Условие

1. Определить свой вариант задания с помощью номера в ISU (он же номер студенческого билета). Вариантом является комбинация 3-й и 5-й цифр. Т.е. если номер в ISU = 123456, то вариант = 35.
2. На основании номера варианта задания выбрать набор из 4 полученных сообщений в виде последовательности 7-символьного кода.
3. Построить схему декодирования классического кода Хэмминга (7;4), которую представить в отчёте в виде изображения.
4. Показать, исходя из выбранных вариантов сообщений (по 4 у каждого – часть №1 в варианте), имеются ли в принятом сообщении ошибки, и если имеются, то какие. Подробно прокомментировать и записать правильное сообщение.
5. На основании номера варианта задания выбрать 1 полученное сообщение в виде последовательности 11-символьного кода.
6. Построить схему декодирования классического кода Хэмминга (15;11), которую представить в отчёте в виде изображения.
7. Показать, исходя из выбранного варианта сообщений (по 1 у каждого – часть №2 в варианте), имеются ли в принятом сообщении ошибки, и если имеются, то какие. Подробно прокомментировать и записать правильное сообщение.

8. Сложить номера всех 5 вариантов заданий. Умножить полученное число на 4. Принять данное число как число информационных разрядов в передаваемом сообщении. Вычислить для данного числа минимальное число проверочных разрядов и коэффициент избыточности.

9. Дополнительное задание №1 (позволяет набрать от 86 до 100 процентов от максимального числа баллов БаРС за данную лабораторную). Написать программу на любом языке программирования, которая на вход получает набор из 7 цифр «0» и «1», записанных подряд, анализирует это сообщение на основе классического кода Хэмминга (7,4), а затем выдает правильное сообщение (только информационные биты) и указывает бит с ошибкой при его наличии.

Основное задание

задание 1 - №10

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	0	1	0	0	0	0

$$r_1 = i_1 \oplus i_2 \oplus i_4 = 1 \oplus 0 \oplus 0 = 1 - \text{true}$$

$$r_2 = i_1 \oplus i_3 \oplus i_4 = 1 \oplus 0 \oplus 0 = 1 - \text{false}$$

$$r_3 = i_2 \oplus i_3 \oplus i_4 = 0 \oplus 0 \oplus 0 = 0 - \text{true}$$

	1	2	3	4	5	6	7
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	x	-	x	-	x	-	x
2	-	x	x	-	-	x	x
4	-	-	-	x	x	x	x

$2^1 = 2 \Rightarrow$ ошибка в символе r_2 , правильное сообщение 1110000

задание 2 - №39

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	1	0	0	0	1	0

$$r_1 = i_1 \oplus i_2 \oplus i_4 = 0 \oplus 0 \oplus 0 = 0 - \text{false}$$

$$r_2 = i_1 \oplus i_3 \oplus i_4 = 0 \oplus 1 \oplus 0 = 1 - \text{true}$$

$$r_3 = i_2 \oplus i_3 \oplus i_4 = 0 \oplus 1 \oplus 0 = 1 - \text{false}$$

	1	2	3	4	5	6	7
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	x	-	x	-	x	-	x

2	-	x	x	-	-	x	x
4	-	-	-	x	x	x	x

$2^0 + 2^2 = 5 \Rightarrow$ ошибка в символе i_2 , правильное сообщение 1100110.

задание 3 - №67

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	1	0	0	1	0	0

$$r_1 = i_1 \oplus i_2 \oplus i_4 = 0 \oplus 1 \oplus 0 = 1 - \text{true}$$

$$r_2 = i_1 \oplus i_3 \oplus i_4 = 0 \oplus 0 \oplus 0 = 0 - \text{false}$$

$$r_3 = i_2 \oplus i_3 \oplus i_4 = 1 \oplus 0 \oplus 0 = 1 - \text{false}$$

	1	2	3	4	5	6	7
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	x	-	x	-	x	-	x
2	-	x	x	-	-	x	x
4	-	-	-	x	x	x	x

$2^1 + 2^2 = 6 \Rightarrow$ ошибка в символе i_3 , правильное сообщение 1100110.

задание 4 - №79

r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	0	0	1	1	0	1

$$r_1 = i_1 \oplus i_2 \oplus i_4 = 0 \oplus 1 \oplus 1 = 0 - \text{false}$$

$$r_2 = i_1 \oplus i_3 \oplus i_4 = 0 \oplus 0 \oplus 1 = 1 - \text{false}$$

$$r_3 = i_2 \oplus i_3 \oplus i_4 = 1 \oplus 0 \oplus 1 = 0 - \text{false}$$

	1	2	3	4	5	6	7
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4
1	x	-	x	-	x	-	x
2	-	x	x	-	-	x	x
4	-	-	-	x	x	x	x

$2^0 + 2^1 + 2^2 = 7 \Rightarrow$ ошибка в символе i_4 , правильное сообщение 1001100.

задание 5 - №82

r_1	r_2	i_1	r_3	i_2	i_3	i_4	r_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}
0	0	1	0	1	0	1	0	0	1	1	0	1	0	1

$$r_1 = i_1 \oplus i_2 \oplus i_4 \oplus i_5 \oplus i_7 \oplus i_9 \oplus i_{11} = 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 = 0 - \text{true}$$

$$r_2 = i_1 \oplus i_3 \oplus i_4 \oplus i_6 \oplus i_7 \oplus i_{10} \oplus i_{11} = 1 \oplus 0 \oplus 1 \oplus 1 \oplus 1 \oplus 0 \oplus 1 = 1 - \text{false}$$

$$r_3 = i_2 \oplus i_3 \oplus i_4 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11} = 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 0 - \text{true}$$

$$r_4 = i_5 \oplus i_6 \oplus i_7 \oplus i_8 \oplus i_9 \oplus i_{10} \oplus i_{11} = 0 \oplus 1 \oplus 1 \oplus 0 \oplus 1 \oplus 0 \oplus 1 = 0 - \text{true}$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2^x	r_1	r_2	i_1	r_3	i_2	i_3	i_4	r_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}
1	x	-	x	-	x	-	x	-	x	-	x	-	x	-	x
2	-	x	x	-	-	x	x	-	-	x	x	-	-	x	x
4	-	-	-	x	x	x	x	-	-	-	-	x	x	x	x
8	-	-	-	-	-	-	-	x	x	x	x	x	x	x	x

$2^1 = 2 \Rightarrow$ ошибка в символе r_2 , правильное сообщение 011010100110101.

задание 6

Информационных разрядов в передаваемом сообщении –
 $(67+10+39+79+82) \cdot 4 = 1108$.

Пусть r – количество проверочных разрядов. Количество информационных разрядов находится по формуле $2^r - r - 1$. Найдем такое ближайшее r , что будет выполняться: $2^r - r - 1 \geq 1108$. (при $r=10$, $2^r - r - 1 = 1013$)

Найдем коэффициент избыточности $k = r / (r + i) = 10 / (10 + 1108) \approx 0,00894454$

Дополнительное задание

Код:

```
def main(number):
    powersCount = 1 #количество степеней двойки

    lenghtForPowers = len(number)
    twoInPowers = [1] #Единица есть всегда тк 0 степень
    falseNum = 0

    #получаем сколько степеней двойки
    while lenghtForPowers>2:
        twoInPowers.append(2**(powersCount))
        powersCount+=1
        lenghtForPowers /= 2

    for r in twoInPowers:
        summIndex = 0 #переменная для подсчета сумм по мод 2
        for step in range(r, len(number)+1, 2*r): #шаг и начало зависят от степени
            if step == r: #убираем саму r (- тк в след ходу она +)
                summIndex -= int(number[r-1])
            for i in range(r):
                if step-1+i<=len(number): #проверка выхода за границу
                    summIndex += int(number[step-1+i])
            if summIndex%2 != int(number[r-1]): # подсчет сумм по мод 2
                falseNum += r
        if falseNum != 0: #проверка есть ли ошибка
            number = number[:falseNum-1] + str(abs(int(number[falseNum-1])-1)) + number[falseNum:]
        else:
            falseNum = "ошибки нету"
    print(f"Правильное сообщение: {number}, индекс ошибки: {falseNum}")

#примеры с лабы
main("1010000")
main("1100010")
main("1100100")
main("1001101")
main("001010100110101")
#пример сообщения без ошибки
main("1110000")
```

Рисунок 1. Программа на Python (дополнительное задание)

Вывод программы: Правильное сообщение: 1110000, индекс ошибки: 2

Правильное сообщение: 1100110, индекс ошибки: 5

Правильное сообщение: 1100110, индекс ошибки: 6

Правильное сообщение: 1001100, индекс ошибки: 7

Правильное сообщение: 011010100110101, индекс ошибки: 2

Правильное сообщение: 1110000, индекс ошибки: ошибки нету.

Вывод

При выполнении данной работы научился работать с кодом Хэмминга, находить ошибки в переданных сообщениях и исправлять их. Поработал с циклами Python. Данные знания помогут мне в освоении будущей профессии и в освоении моей деятельности.

Список литературы

- Статья хабр "Код Хэмминга. Пример работы алгоритма" <https://habr.com/ru/articles/140611/>
- Видео-урок на ютуб: https://www.youtube.com/watch?v=ehuNcmE8S84&ab_channel=Artemy