Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

**Факультет программной инженерии и компьютерной техники**

**Информатика**

Лабораторная работа №2

«Синтез помехоустойчивого кода»

Выполнил: Герасимов Артём Кириллович

Группа: P3108

Вариант: 95

Преподаватель: Малышева Татьяна Алексеевна

Санкт-Петербург

2021

Оглавление

[Задание 2](#_Toc85392396)

[Этапы вычисления 3](#_Toc85392397)

[Задание 1 3](#_Toc85392398)

[Задание 2 3](#_Toc85392399)

[Задание 3 4](#_Toc85392400)

[Задание 4 4](#_Toc85392401)

[Задание 5 5](#_Toc85392402)

[Задание 6 5](#_Toc85392403)

[Вывод 6](#_Toc85392404)

[Список литературы 6](#_Toc85392405)

Задание

Определить свой вариант задания с помощью номера ИСУ:

Мой номер 339052, следовательно мой вариант 95:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | 1 | | | | 2 |
| 95 | 79 | 9 | 51 | 91 | 93 |

Выполнить следующие задания:

1. Построить схему декодирования классического кода Хэмминга (7;4).
2. Показать имеются ли в принятом сообщении (всего 4 варианта) ошибки, и если имеются, то какие. Прокомментировать и записать правильное сообщение.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ALT | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|  |  |  |  |  |  |  |
| 79 | 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 9 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 51 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 91 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |

**Таблица 1 – Задание №1 из варианта 95**

1. Построить схему декодирования классического кода Хэмминга (15;11).
2. Показать имеются ли в принятом сообщении (всего 1 вариант) ошибки, и если имеются, то какие. Прокомментировать и записать правильное сообщение.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ALT | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 93 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

**Таблица 2 – Задание №2 из варианта 95**

1. 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.
2. (*Необязательно*) Написать программу на любом языке программирования, которая на вход из командной строки 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.

Этапы вычисления

Задание 1

Построим схему декодирования классического кода Хэмминга (7;4):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 |  |
|  |  |  |  |  |  |  |  |  |
| 1 | X |  | X |  | X |  | X |  |
| 2 |  | X | X |  |  | X | X |  |
| 4 |  |  |  | X | X | X | X |  |

**Таблица 3 – Схема декодирования классического кода Хэмминга (7;4)**

Задание 2

**Вариант 79:**

Возьмем значения из **таблицы 1:**

Следовательно, . Подставив значения в **таблицу 3**, понимаем, что ошибка в , таким образом правильное сообщение: **1001100.**

**Вариант 9:**

Возьмем значения из **таблицы 1:**

Следовательно, . Подставив значения в **таблицу 3**, понимаем, что ошибка в , таким образом правильное сообщение: **1001100.**

**Вариант 51:**

Возьмем значения из **таблицы 1:**

Следовательно, . Подставив значения в **таблицу 3**, понимаем, что ошибка в , таким образом правильное сообщение: **1000011.**

**Вариант 91:**

Возьмем значения из **таблицы 1:**

Следовательно, . Подставив значения в **таблицу 3**, понимаем, что ошибка в , таким образом правильное сообщение: **0111100.**

Задание 3

Построим схему декодирования классического кода Хэмминга (15;11):

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| 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 |  |

**Таблица 4 – Таблица кода Хэмминга (15;11)**

Задание 4

Возьмем значения из **таблицы 2:**

Следовательно, . Подставив значения в **таблицу 4** , понимаем, что ошибка в , таким образом правильное сообщение: **001010110010101.**

Задание 5

Сложим номера всех 5 вариантов и умножим на 4:

Минимальное число контрольных разрядов определяется по формуле:

, где ( - число информационных разрядов)

Методом подбора найдем значение , которое удовлетворяет формуле:

Следовательно, минимальное число контрольных разрядов

Коэффициент избыточности вычисляется по формуле:

Подставим значения и найдем его:

Задание 6

while True:  
 digits = input('Введите набор из 7 цифр «0» и «1»: ')  
  
 if len(digits) == 7:  
 for n in range(len(digits)):  
 if digits[n] in '01' and digits[n+1] in '01':  
 r1 = int(digits[0]); r2 = int(digits[1]); r3 = int(digits[3])  
 i1 = int(digits[2]); i2 = int(digits[4]); i3 = int(digits[5]); i4 = int(digits[6])  
 s1 = r1 ^ i1 ^ i2 ^ i4  
 s2 = r2 ^ i1 ^ i3 ^ i4  
 s3 = r3 ^ i2 ^ i3 ^ i4  
  
 print('\nS1 = ', str(r1), '⊕', str(i1), '⊕', str(i2), '⊕', str(i4), '=', str(s1),  
 '\nS2 = ', str(r2), '⊕', str(i1), '⊕', str(i3), '⊕', str(i4), '=', str(s2),  
 '\nS3 = ', str(r3), '⊕', str(i2), '⊕', str(i3), '⊕', str(i4), '=', str(s3), '\n')  
  
 error = int(str(s3) + str(s2) + str(s1), 2)  
 if error == 0:  
 print('Ошибок нет')  
 else:  
 print('Ошибка в бите', error)  
 correctDigits = []  
 for i in range(len(digits)):  
 if i == error - 1:  
 if digits[i] == "0":  
 correctDigits.append(1)  
 if digits[i] == "1":  
 correctDigits.append(0)  
 else:  
 correctDigits.append(digits[i])  
  
 print('\nПравильное сообщение: ', end='')  
 for i in correctDigits:  
 print(i, end='')  
 print() # чтобы отделить от Process finished with exit code  
 break  
 else:  
 print('Ошибка: неверные символы (можно только 0 и 1)')  
 break  
 else:  
 print('Ошибка: введены не 7 цифр и/или неверные символы')  
  
 exit = input('Чтобы завершить работу программы, введите «exit», чтобы продолжить - любой символ: ')  
 if exit == 'exit':  
 break

Вывод

Во время выполнения данной лабораторной работы я понял принцип работы кода Хэмминга, научился строить схемы Хэмминга (7;4) и (15;11), определять ошибочные биты и исправлять их. Также я написал программу для определения ошибочных битов и исправления их с помощью кода Хэмминга.



**Рисунок 1 - Американский математик Ричард Уэсли Хэмминг**

Список литературы

1. [tltshnik](https://habr.com/ru/users/tltshnik/) Код Хэмминга. Пример работы алгоритма, 2012г. Режим доступа: <https://habr.com/ru/post/140611/>
2. Орлов С.А., Цилькер Б.Я. Организация ЭВМ и систем: Учебник для вузов, 2-е изд. - СПб.:Питер,2011. Режим доступа: <https://nsu.ru/xmlui/handle/nsu/9052>