Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ульяновский государственный технический университет»

Кафедра «Вычислительная техника»

Дисциплина «Основы информационных технологий»

Лабораторная работа №3

**«Кодирование-декодирование»**

Выполнил студент

группы ИВТбд-11

Каршибоев Я.Ш

Проверил: Валюх В.В.

Ульяновск, 2022

**Цели работы:**

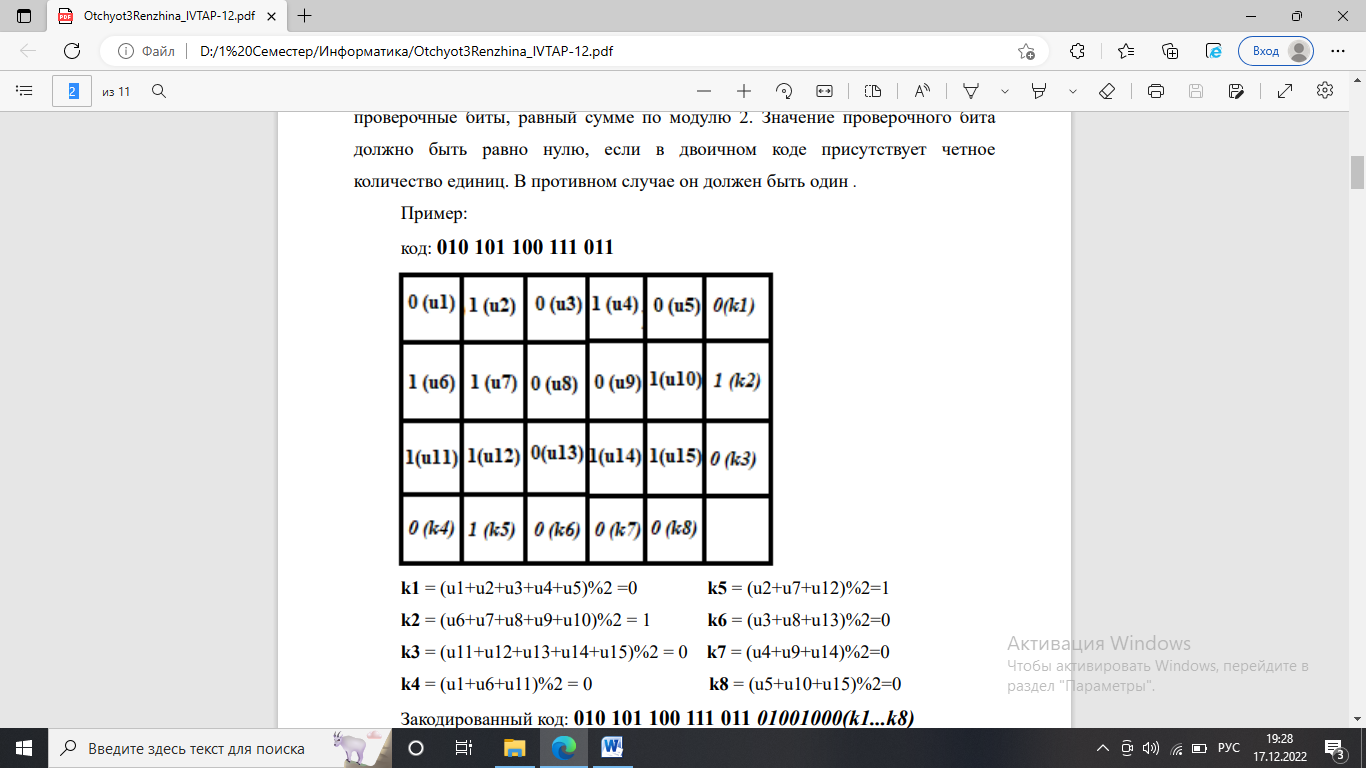
Написать программу, осуществляющая кодирование-декодирование.

**Теория**

**Код с проверкой на четность по вертикали и горизонтали** — это такой метод кодирования, при котором к переданному сообщению прибавляются проверочные биты, равный сумме по модулю 2. Значение проверочного бита должно быть равно нулю, если в двоичном коде присутствует четное количество единиц. В противном случае он должен быть один .

**Пример:**

код: **010 101 100 111 011**



**k1** = (u1+u2+u3+u4+u5)%2 =0  **k5** = (u2+u7+u12)%2=1

**k2** = (u6+u7+u8+u9+u10)%2 = 1  **k6** = (u3+u8+u13)%2=0

**k3** = (u11+u12+u13+u14+u15)%2 = 0  **k7** = (u4+u9+u14)%2=0

**k4** = (u1+u6+u11)%2 = 0 **k8** = (u5+u10+u15)%2=0

Закодированный код: **010 101 100 111 011 01001000(k1...k8)**

Декодирование осуществляется таким образом:

Если сумма по модулю каждой строки и столбца равны 0, то сообщение передано без ошибок, но если сумма равна 1, то значит при кодировании былизменен бит.

Закодированное сообщение: **010 101 101 111 011 01001000**

**(е=000 000 001 000 000 000 000 00)**

**s1**=(k1+u1+u2+u3+u4+u5)%2 =0

**s2**=(k2 +u6+u7+u8+u9+u10)%2 = 1

**s3**=(k3+u11+u12+u13+u14+u15)%2 = 0

**s4**=(k4+u1+u6+u11)%2 = 0

**s5**=(k5+u2+u7+u12)%2=0

**s6**=(k6+u3+u8+u13)%2=0

**s7**=(k7+u4+u9+u14)%2=1

**s8**=(k8+u5+u10+u15)%2=0

Из этого следует что сообщение передано с ошибкой, исправляем находя по строке и столбцу и получаем исправленный код.

Декодированный код: **010 101 100 111 011**

Бывают случаи, когда код, переданный с ошибкой при декодировании не выдает ошибку, и мы получаем код с ошибкой.

**Пошаговый алгоритм выполнения кода**

1. Кодирование: Создаем массивы, для записи в них сообщения и дальнейшей работы с сообщением.

2. Вводим с клавиатуры сообщение и заносим в первый массив, а также в массив, с которым будем работать.

3. Далее по каждой строке и столбцу считаем сумму элементов по модулю 2 и записываем туда, по чему мы считали.

4. Выводим наше закодированное сообщение.

5. **Декодирование**: Создаем также массив для работы с сообщением, и другие переменные, которые нам понадобятся.

6. Заносим наше сообщение в массив.

7. Если сумма по строкам и столбцу не равны 0, то записываем индекс строки и столбца, а также выводим сообщение о том, что в этих местах ошибка.

8. Далее, если элемент массива с индексами той строчки и столбца равен 0,

то меняем на 1 и аналогично для 1.

9. Выводим наше закодированное сообщение.

**Контрольные тесты для программы**

1. **Код переданный без ошибок**

**КОДИРОВАНИЕ**

Введите сообщение из 9 чисел в двоичной системе счисления, цифры через пробел:

**0 1 0 1 0 1 1 0 0**

Матрица:

**0101**

**1010**

**1001**

**011**

Закодированное сообщение: **010101100101011**

**ДЕКОДИРОВАНИЕ**

Введите закодированное сообщение:

**0 1 0 1 0 1 1 0 0 1 0 1 0 1 1**

Матрица:

**0101**

**1010**

**1001**

**011**

Сообщение передано без ошибок

Декодированный код: **010101100**

3. Код передан с одной ошибкой

**КОДИРОВАНИЕ**

Введите сообщение из 9 чисел в двоичной системе счисления, цифры через пробел:

**0 0 1 1 1 1 0 1 0**

Матрица:

**0011**

**1111**

**0101**

**100**

Закодированное сообщение: **001111010111100**

**ДЕКОДИРОВАНИЕ**

Введите закодированное сообщение:

**0 0 1 1 1 0 0 1 0 1 1 1 1 0 0**

Матрица:

**0011**

**1101**

**0101**

**100**

1 ошибка в строке 2

1 ошибка в столбце 3

Исправленный декодированный код:

**001111010**

4. Код передан с двумя ошибками

**КОДИРОВАНИЕ**

Введите сообщение из 9 чисел в двоичной системе счисления, цифры через пробел:

**1 0 0 0 1 0 1 0 1**

Матрица:

**1001**

**0101**

**1010**

**011**

Закодированное сообщение: **100010101110011**

**ДЕКОДИРОВАНИЕ**

Введите закодированное сообщение:

**0 0 0 0 1 0 1 0 0 1 1 0 0 1 1**

Матрица:

**0001**

**0101**

**1000**

**011**

1 ошибка в строке 1

2 ошибка в строке 3

1 ошибка в столбце 1

2 ошибка в столбце 3

Исправленный декодированный код:

**100010101**

5. Код передан с такими ошибками, которую декодер не сможет исправить

**КОДИРОВАНИЕ**

Введите сообщение из 9 чисел в двоичной системе счисления, цифры через пробел:

**1 0 1 0 0 0 1 1 0**

Матрица:

**1010**

**0000**

**1100**

**011**

Закодированное сообщение: **101000110000011**

**ДЕКОДИРОВАНИЕ**

Введите закодированное сообщение:

**1 0 1 0 0 1 1 1 1 0 0 0 0 1 1**

Матрица:

**1010**

**0010**

**1110**

**011**

Исправленный декодированный код:

**101001111**

Это произошло из-за того, что в столбце, сумма оказалась равная 0.

**Теоретические вопросы**

**1.**Какой вид имеют комбинации корректирующего кода Хэмминга для передачи сообщения 1011?

**2**.Передана следующая комбинация в коде Хэмминга 1101001, получена 1001001. Показать процесс обнаружения ошибки.

**3.**Первичный алфавит состоит из трёх знаков с вероятностями *p*1 = 0,2;   
*р*2 = 0,7; *р*3 = 0,1. Для передачи по каналу без помех используются равномерный двоичный код. Частота тактового генератора 500 Гц. Какова пропускная способность канала?

**Ответы на теоретические вопросы**

**1. P0 = D0 + D1 + D3**

**P1 = D0 + D2 + D3**

**P2 = D1 + D2 + D3**

2. Принята комбинация **"1001001"**:

а) составляем синдром ошибки. Для этого из таблицы выписываем номера позиций, охватываемых первой, второй и третьей проверками. Общее число символов в комбинации n = 7. Число контрольных символов определяется из формулы и равно 3, следовательно, трех проверок достаточно;

б) первая проверка охватывает позиции 1, 3, 5, 7

1 A 0 A 0 A 1 =0.

Следовательно, в младшем разряде двоичного числа - синдрома ошибки должен быть записан 0;

в) вторая проверка охватывает позиции 2, 3, 6, 7

0 A 0 A 0 A 1 =1.

Следовательно, в первом разряде синдрома ошибки должна быть записана 1;

г) третья проверка охватывает позиции 4, 5, 6, 7

1 A 0 A 0 A 1 =0.

Следовательно, в старшем (втором) разряде синдрома ошибки должен быть записан 0.Таким образом, синдром ошибки для данного кодового сочетания - 010, что соответствует десятичной цифре 2. Следовательно, ошибочный разряд второй.

3. **p1 = 0,2; р2 = 0,7; р3 = 0,1, частота = 500 Гц**

**N = 3**

**I1 = - 0,2∙log20,2 - 0,7∙log20,7 - 0,1log20,1 = 1,16 бит**

**С = υ\* I1/K^2 = 500 \* 1,16/2 = 290 бит**

**Приложение. Код программы:**

#include <stdio.h>

#include<stdlib.h>

#include<conio.h>

#include<locale>

#include <malloc.h>

int main() {

setlocale(LC\_ALL, "RUS");

const int m = 3, n = 3;

const int m1 = 4, n1 = 4;

int arr[m][n];

int arr1[m1][n1];

printf("КОДИРОВАНИЕ\n");

printf("Введите сообщение из 9 чисел в двоичной системе счисления, цифры через пробел:\n");

for (int i = 0; i < m; i++) {

for (int j = 0; j < n; j++) {

scanf\_s("%d", &arr[i][j]);

arr1[i][j] = arr[i][j];

}

}

for (int i = 0; i < m1; i++) {

int s = 0;

for (int j = 0; j < n; j++) {

s += arr1[i][j];

}

arr1[i][n1 - 1] = s % 2;

}

for (int j = 0; j<n1; j++) {

int s = 0;

for (int i = 0; i<m ; i++) {

s += arr1[i][j];

}

arr1[m1-1][j] = s % 2;

}

arr1[m1 - 1][n1 - 1] = 2;

printf("Матрица:\n");

for (int i = 0; i < m1; i++) {

for (int j = 0; j < n1; j++) {

if (arr1[i][j] != 2) {

printf("%d", arr1[i][j]);

}

}

printf("\n");

}

printf("Закодированное сообщение: ");

for (int i = 0; i < m; i++) {

for (int j = 0; j < n; j++) {

printf("%d", arr1[i][j]);

}

}

for (int i = 0; i < m; i++) {

printf("%d", arr1[i][n1 - 1]);

}

for (int i = 0; i < n; i++) {

printf("%d", arr1[m1-1][i]);

}

printf("\n\n");

printf("ДЕКОДИРОВАНИЕ\n");

int darr[5][3];

int arr2[m1][n1];

int str[4] = { 9,9,9,9 };

int st[4] = {9,9,9,9};

int k = 0;

int l = 0;

printf("Введите закодированное сообщение:\n");

for (int i = 0; i < 5; i++) {

for (int j = 0; j < 3; j++) {

scanf\_s("%d", &darr[i][j]);

}

}

for (int i = 0; i < 3; i++) {

for (int j = 0; j < 3; j++) {

arr2[i][j] = darr[i][j];

}

}

arr2[0][n1 - 1] = darr[3][0];

arr2[1][n1 - 1] = darr[3][1];

arr2[2][n1 - 1] = darr[3][2];

arr2[m1-1][0] = darr[4][0];

arr2[m1 - 1][1] = darr[4][1];

arr2[m1 - 1][2] = darr[4][2];

arr2[m1 - 1][n1 - 1] = 2;

printf("Матрица:\n");

for (int i = 0; i < m1; i++) {

for (int j = 0; j < n1; j++) {

if (arr2[i][j] != 2) {

printf("%d", arr2[i][j]);

}

}

printf("\n");

}

int s;

for (int i = 0; i < m1 - 1; i++) {

s = 0;

for (int j = 0; j < n1; j++) {

s += arr2[i][j];

}

if (s % 2 != 0) {

str[k] = i;

printf("%d ошибка в строке %d\n", k + 1, str[k] + 1);

k++;

}

}

int t;

for (int j = 0; j < n1 - 1; j++) {

t = 0;

for (int i = 0; i < m1; i++) {

t += arr2[i][j];

}

if (t % 2 != 0) {

st[l] = j;

printf("%d ошибка в столбце %d\n", l + 1, st[l] + 1);

l++;

}

}

if (k>0 && l==0){

printf(ошибка в бите %d\n", str[k]+9);

}

else if(l>0 && k==0){

printf(ошибка в бите %d\n", st[l]+14);

}

if (k > 0 && l>0) {

int cm, cn;

for(int i=0;i<k;i++){

cm = str[i];

cn = st[i];

if (arr2[cm][cn] == 1) {

arr2[cm][cn] = 0;

}

else {

arr2[cm][cn] = 1;

}

}

printf("Исправленный декодированный код:\n");

}

else if (k == 0) {

printf("Сообщение передано без ошибок\n");

printf("Декодированный код: ");

}

for (int i = 0; i < m1 - 1; i++) {

for (int j = 0; j < n1 - 1; j++) {

printf("%d", arr2[i][j]);

}

}

}