

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

**Факультет программной инженерии и компьютерной техники**

**Лабораторная работа №2  
"Синтез помехоустойчивого кода"**

**Вариант 81**

Выполнил:

Студент  
группы Р3116

Брагин  
Роман Андреевич

Проверила:

доцент факультета ПИиКТ  
Авксентьева Елена Юрьевна

г. Санкт-Петербург

2023 г.

Оглавление	
Задание 1-2	3
Задание 3-4:	4
Задание 5-6:	5
Дополнительное задание:	6
Заключение	7
Используемая литература	7

Задание 1-2:

Задание 1

Исходный код: 1010100

001	010	011	100	101	110	111			
1	2	3	4	5	6	7			
1	0	1	0	1	1	0			
S1		1	1	1	0	=	1		
S2		0	1	1	0	=	0		ошибка в 001 т.е в 1 бите
S3		0	1	1	0	=	0		
при замене 1 на 0 в 1 бите									
001	010	011	100	101	110	111			
1	2	3	4	5	6	7			
0	0	1	0	1	1	0			
S1		0	1	1	0	=	0		
S2		0	1	1	0	=	0		ошибок нет
S3		0	1	1	0	=	0		

Ответ: в этом коде есть ошибка в 2 контрольном бите, при замене его на 0 получаем правильный код 0010110, при декодировании получаем код 1110

Задание 2:

Изначальный код 1010000

001	010	011	100	101	110	111			
1	2	3	4	5	6	7			
1	0	1	0	0	0	0			
S1		1	1	0	0	=	0		
S2		0	1	0	0	=	1		ошибка в 010 т.е в 2 бите
S3		0	0	0	0	=	0		
при замене 1 на 0 в 1 бите									
001	010	011	100	101	110	111			
1	2	3	4	5	6	7			
1	1	1	0	0	0	0			
S1		1	1	0	0	=	0		
S2		1	1	0	0	=	0		ошибок нет
S3		0	0	0	0	=	0		

Ответ: Ошибка была во 2 контрольном бите, правильно код будет выглядеть так

1110000, при декодировании получаем код 1000

Задание 3-4:

Задание 3

Изначальный код 1001010

001	010	011	100	101	110	111				
1	2	3	4	5	6	7				
1	0	0	1	0	1	0				
S1		1	0	0	0	=	1			
S2		0	0	1	0	=	1		ошибка в 011 т.е в 3бите	
S3		1	0	1	0	=	0			
при замене 1 на 0 в 1 бите										
001	010	011	100	101	110	111				
1	2	3	4	5	6	7				
1	0	1	1	0	1	0				
S1		1	1	0	0	=	0			
S2		0	1	1	0	=	0		ошибок нет	
S3		1	0	1	0	=	0			

Ответ: Ошибка была в 3 контрольном бите, при замене его на обратный ему 0 получаем правильный код: 0101010, при декодировании получаем код 1010

Задание 4:

Исходный код: 0111101

001	010	011	100	101	110	111				
1	2	3	4	5	6	7				
0	1	1	1	1	0	1				
S1		0	1	1	1	=	1			
S2		1	1	0	1	=	1		ошибка в 111 т.е в 7 бите	
S3		1	1	0	1	=	1			
при замене 1 на 0 в 1 бите										
001	010	011	100	101	110	111				
1	2	3	4	5	6	7				
0	1	1	1	1	0	0				
S1		0	1	1	0	=	0			
S2		1	1	0	0	=	0		ошибок нет	
S3		1	1	0	0	=	0			

Ответ: Ошибка была в 7 бите и при замене 1 на 0 получаем правильный код: 0111100, при декодировании получаем код 1100

Задание 5-6:

Задание 5

Исходный код: 001010100000101

0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	1	0	1	0	1	0	0	0	0	0	1	0	1
s1	0	1	1	1	0	0	1	1	1					
s2	0	1	0	1	0	0	0	1	1					
s3	0	1	0	1	0	1	0	1	0					
s4	0	0	0	0	0	1	0	1	0					
ОШИБКА В 0011 Т.Е В 3 БИТЕ														
0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	1	0	1	0	0	0	0	0	1	0	1
s1	0	0	1	1	0	0	1	1	0					
s2	0	0	0	1	0	0	0	1	0					
s3	0	1	0	1	0	1	0	1	0					
s4	0	0	0	0	0	1	0	1	0					

Ответ: Ошибка была в 3 бите, поменяв 1 на 0 получаем правильный код: 000010100000101, декодировав получаем 001010000101

Задание 6

1076

$2^r \geq r + i + 1$ , при  $i = 1076, r = 11$

Коэффициент избыточности  $n = i + r = 1087$

## Дополнительное задание:

```
2 usages
> def s2(n):...
1 usage
> def s1(n):...
# проверка кода
2 usages
> def pr(n):...
y = ''
yt = ''

print('ВВЕДИТЕ КОД НА ПРОВЕРКУ')
a = input()
a = '5' + a
for i in range(len(a)):
    if s2(i) == 1:
        y += str(s1(i))
y = y[::-1]
h = int(y, 2)
u = int(a[h])
u = int(a[h]) + 1
u = u % 2
if y.count('1') % 2 == 0:
    print('НЕТ ОШИБОК')
    print('КОД ')
    print(a[1:])
    print('ДЕКОДИРОВАННОЕ СООБЩЕНИЕ')
    print(pr(a))
else:
    a = a[:h] + str(u) + a[h+1:]
    print('! ОШИБКА !')
    print('ИСПРАВЛЕННЫЙ КОД ')
    print(a[1:])
    print('ДЕКОДИРОВАННОЕ СООБЩЕНИЕ')
    print(pr(a))
    print('ОШИБОЧНЫЙ БИТ', h)
```

```
def s2(n):
    k = 0
    for i in range(0, 100):
        if (2**i == n):
            k += 1
    if k == 1:
        return 1
    else:
        return 0
1 usage
def s1(n):
    if n != 1:
        k = ''
        for i in range(n, len(a), 2*n):
            for j in range(i, i+n):
                k += str(a[j])

    if n == 1:
        k = ''
        for i in range(len(a)):
            if i % 2 != 0:
                k += str(a[i])

    if k.count('1') % 2 == 0:
        return '0'
    else:
        return '1'
# проверка кода
2 usages
def pr(n):
    a = ''
    for i in range(len(n)):
        if s2(i) != 1 and i != 0:
            a += str(n[i])
    return a
```

```
ВВЕДИТЕ КОД НА ПРОВЕРКУ
1010110
! ОШИБКА !
ИСПРАВЛЕННЫЙ КОД
0010110
ДЕКОДИРОВАННОЕ СООБЩЕНИЕ
1110
ОШИБОЧНЫЙ БИТ 1
```

## Заключение

Проделав данную лабораторную работу, я научился работать с помехоустойчивым кодом Хэмминга, и разобрался в его работе.

## Используемая литература

1. Код Хэмминга. Самоконтролирующийся и самокорректирующийся код. - YouTube
2. Код Хэмминга. Пример работы алгоритма / Хабр (habr.com)