МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ   
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”



ДОСЛІДЖЕННЯ КОДУ З ОДНІЄЮ ПЕРЕВІРКОЮ НА ПАРНІСТЬ

**Звіт до лабораторної роботи №4**

**з курсу “Теорія інформації та кодування”**

Варіант 36

Виконав:  
ст. гр. ІР-21

Касараба Володимир  
  
Прийняв:  
Стахів Р. І.

Львів – 2021

**Мета роботи:** вивчення характеристики і принципу побудови двійкового коду з однією перевіркою на парність.

**Завдання**

1. За вказівкою викладача з таблиці 1, згідно з номером списку в журналі групи, вибрати десяткову комбінацію, яку необхідно перевести в комбінацію простого двійкового коду. Одержані двійкові комбінації перетворити в комбінацію коду з однією перевіркою на парність.

|  |  |
| --- | --- |
| Варіант | Десяткова комбінація  *N10* |
| 36 | 798 |

2. Записати основні характеристики для одержаних кодів з однією перевіркою на парність.

3. Розробити алгоритм програми кодування кодом з однією перевіркою на парність і декодування з виявленням можливих помилок (порушенням парності).

**Послідовність виконання роботи**

1. Задану десяткову (табл. 1) комбінацію перетворити в комбінацію простого двійкового коду.

2. Для розроблених двох алгоритмів програм кодування і декодування написати, набрати і відлагодити програми на мові Python

3. Провести кодування і декодування двійкових кодів (з попереднього одержаного завдання).

4. Роздрукувати одержані результати і тексти програм перетворення.

5. Зробити висновки щодо виконаної роботи.

**Виконання**

Задана десяткова комбінація: 798

Комбінація двійкового коду: 1100011110

**Основні характеристики для одержаних кодів з однією перевіркою на парність:**

Порівняння переданого контрольного розряду з обчисленнями виконується з допомогою елемента „**ВИКЛЮЧНЕ АБО**” (mod 2). Якщо вони відрізняються, то виробляється сигнал помилки S = 1.

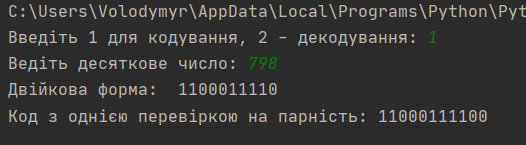
Для кодів з бітом перевірки на парність передавач і приймач ідентичні.

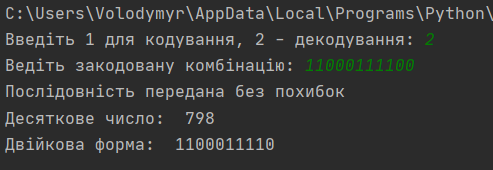
Програма кодування/декодування на мові Python:

def get\_code\_with\_parity\_check(binary):  
 numbers = list(binary)  
 numbers = map(int, numbers)  
 parity\_symbol = sum(numbers) % 2  
 coded = binary + str(parity\_symbol)  
 return 'Код з однією перевіркою на парність: ' + str(coded)  
  
  
def check\_code\_by\_parity(coded):  
 parity\_symbol = coded[-1]  
 binary = coded[:-1]  
 numbers = list(binary)  
 numbers = map(int, numbers)  
 if sum(numbers) % 2 == int(parity\_symbol):  
 return 'Послідовність передана без похибок'  
 else:  
 return 'Присутня похибка'  
  
  
choice = input('Введіть 1 для кодування, 2 - декодування: ')  
  
if choice == "1":  
 dec = int(input('Ведіть десяткове число: '))  
 binary = str(bin(dec))[2:]  
 print('Двійкова форма: ', binary)  
 print(get\_code\_with\_parity\_check(binary))  
  
if choice == "2":  
 coded = input('Ведіть закодовану комбінацію: ')  
 print(check\_code\_by\_parity(coded))  
 print('Десяткове число: ', int('0b'+coded[:-1], 2))

print('Двійкова форма: ', coded[:-1])

**Результат виконання програми**





**Висновок:**

На цій лабораторній роботі я вивчив характеристики і принцип побудови двійкового коду з однією перевіркою на парність та написав програму мовою Python для кодування та декодування. Порівняння переданого контрольного розряду з обчисленнями я виконав з допомогою «виключного або» (mod 2).

Згідно варіанту, для кодування я використав десяткове число 798, двійква форма якого 11000111110. Послідовність кодування з однією перевіркою на парність: 110001111100.

Як бачимо, з результів виконання програми, програма працює коректно, оскільки подане на кодування десяткове число 798 і відповідна йому двійкова форма збігаються з результатами декодування.