Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Сибирский государственный университет телекоммуникаций и информатики»  
(СибГУТИ)

Институт информатики и вычислительной техники  
Кафедра прикладной математики и кибернетики

**Практическая работа №4  
 по дисциплине «Теория информации»  
на тему «Определение параметров линейного кода»**

Выполнил:  
студент гр.ИП-014

Обухов А.И.

Проверила:  
Старший преподаватель каф. ПМиК  
Дементьева Кристина Игоревна

Новосибирск 2024 г.

**Цель работы:** Изучение свойств линейного корректирующего кода

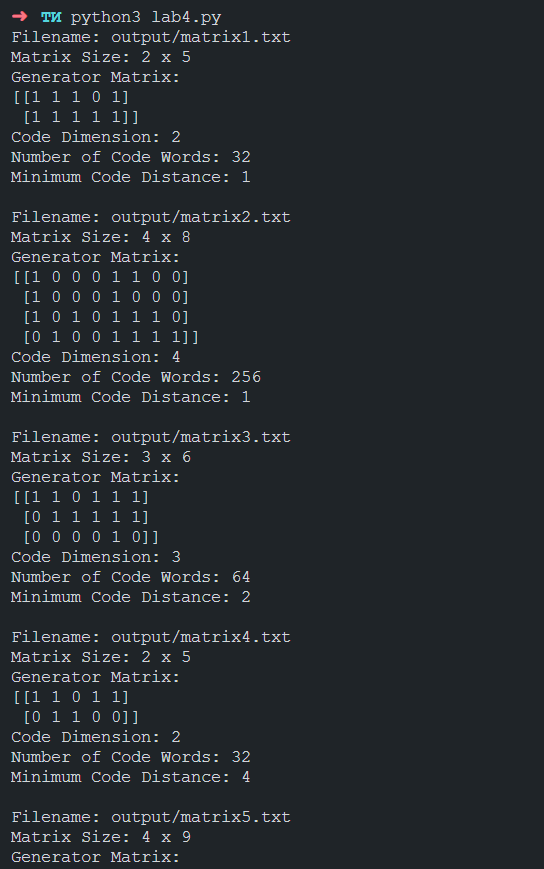
**Язык программирования:** С, С++, С#, Python

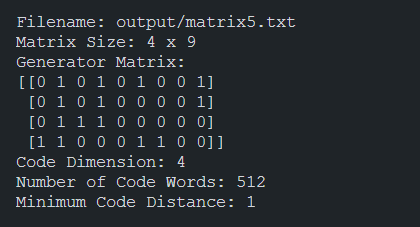
**Результат:** программа, тестовые примеры, отчет.

**Задание:**

1. Порождающая матрица записана в текстовом файле. Файл имеет следующий формат: в первой строке через пробел записаны два натуральных числа n (количество строк матрицы) и m (количество столбцов), в следующих n строках записаны через пробел по m нулей и единиц. Файл можно генерировать случайно.  
   Пример файла  
   3 5  
   1 0 1 1 1  
   0 1 0 1 0  
   0 0 1 1 1
2. По заданной порождающей матрице определить характеристики линейного кода: размерность кода, количество кодовых слов, минимальное кодовое расстояние. Использовать 5 различных файлов.
3. Оформить отчет с результатами , загрузить отчет в электронную среду. По желанию в отчет можно включить описание программной реализации. **В отчет не нужно включать содержимое этого файла.**

**Скриншоты работы программы**





**Анализ результатов работы программы**

1. **Свойства линейного корректирующего кода**:
   * Ключевыми характеристиками кода являются размерность, количество кодовых слов и минимальное кодовое расстояние, которые влияют на эффективность исправления ошибок.
   * Линейный кодирующий код позволяет обнаруживать и исправлять ошибки при передаче информации.
2. **Зависимость характеристик кода от порождающей матрицы**:
   * Минимальное кодовое расстояние определяется внутренними свойствами матрицы, такими как количество единиц в каждой строке и их расположение.
   * Размерность кода зависит от числа строк в порождающей матрице.
   * Количество кодовых слов зависит от количества столбцов в порождающей матрице.
3. **Влияние размерности и минимального кодового расстояния на надежность кода**:
   * Большая размерность кода приводит к возможности создания большего количества кодовых комбинаций, что обычно увеличивает его возможности для исправления ошибок.
   * Большее минимальное кодовое расстояние обеспечивает лучшую способность кода обнаруживать и исправлять ошибки при передаче данных.

**Листинг программы**

import numpy as np

from typing import Tuple

def generate\_generator\_matrix(filename: str, n: int, m: int) -> None:

    generator\_matrix = np.random.randint(0, 2, size=(n, m))

    with open(filename, 'w') as file:

        file.write(f"{n} {m}\n")

        for row in generator\_matrix:

            file.write(' '.join(map(str, row)) + '\n')

def read\_generator\_matrix(filename: str) -> Tuple[np.ndarray, int, int]:

    with open(filename, 'r') as file:

        lines = file.readlines()

        n, m = map(int, lines[0].split())

        generator\_matrix = np.array([list(map(int, line.split())) for line in lines[1:]])

    return generator\_matrix, n, m

def linear\_code\_characteristics(matrix: np.ndarray) -> Tuple[int, int, int]:

    n, m = matrix.shape

    code\_dimension = n

    code\_words = 2 \*\* m

    min\_distance = float("inf")

    for code in range(n):

        for i in range(code + 1, n):

            distance = np.sum(matrix[code] != matrix[i])

            min\_distance = min(min\_distance, distance)

    return code\_dimension, code\_words, min\_distance

def main() -> None:

    num\_of\_matrices = 5

    for i in range(1, num\_of\_matrices + 1):

        filename = f"output/matrix{i}.txt"

        n = np.random.randint(2, 5)  # ROWS

        m = np.random.randint(2, 10)  # COLS

        generate\_generator\_matrix(filename, n, m)

    for i in range(1, num\_of\_matrices + 1):

        filename = f"output/matrix{i}.txt"

        generator\_matrix, n, m = read\_generator\_matrix(filename)

        print(f"Filename: {filename}")

        print(f"Matrix Size: {n} x {m}")

        print("Generator Matrix:")

        print(generator\_matrix)

        code\_dimension, code\_words, min\_distance = linear\_code\_characteristics(generator\_matrix)

        print(f"Code Dimension: {code\_dimension}")

        print(f"Number of Code Words: {code\_words}")

        print(f"Minimum Code Distance: {min\_distance}")

        print()

if \_\_name\_\_ == "\_\_main\_\_":

    main()