**Петрозаводский государственный университет**

**Институт математики и информационных технологий**

**Кафедра Информатики и Математического Обеспечения**

**Информационный системы и технологии**

**Отчет о лабораторной работе №2 по курсу «Информационная безопасность и защита информации»**

**Тема:**

**Защитное кодирование по методу Хэмминга**

**Выполнил:** Шарипов Отабек 2 курса г22206

**Проверил: Соколов Владислав Евгеньевич****-** старший преподаватель.

**Петрозаводск 2022г**

**Содержание**

1. Краткое описание метода

2. Функциональные части

3. Фрагменты программной реализации

4. Пример

Заключение

1. **Краткое описание метода**

# • Одним из методов, позволяющих фиксировать и исправлять единичные ошибки в блоке двоичных данных, является метод Хэмминга. При использовании этого метода вся последовательность данных разбивается на блоки фиксированной длины n. При кодировании длина фиксированного блока увеличивается до такой длины N, чтобы дополнительные биты позволяли сформировать число

# от нуля до N – для фиксации места ошибки (позиции ошибочного бита блока).

# • Метод требует формирования матрицы контрольного суммирования, каждый столбец которой представляет собой двоичное значение порядкового номера столбца (число столбцов равно N). Эта матрица будет умножаться (по модулю 2) на передаваемый вектор.

**2.Функциональные части**

# • hamming(sequence, length) - функция для закодирования битовой последовательности;

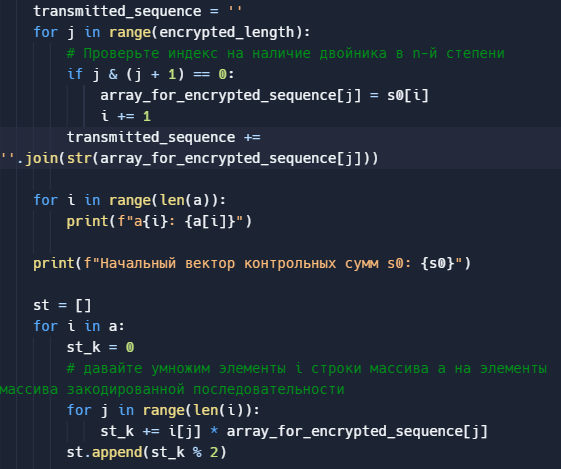
# • calculate(length) - функция для расчета количества контрольных бит;

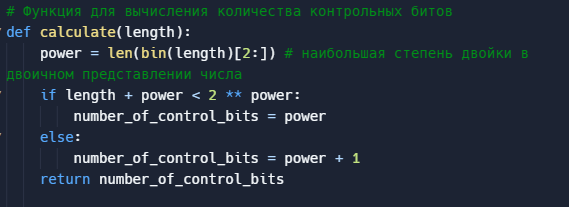
# • check(sequence) - функция для проверки передаваемой битовой последовательности;

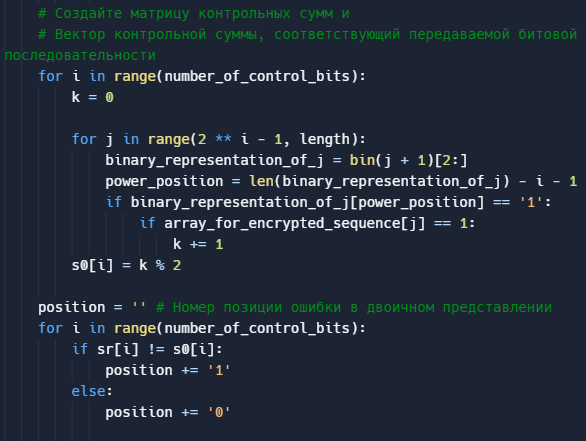
# • main() - функция для вспомогательных действий.

**3 .Фрагменты программной реализации**

## • Фрагмент функции для закодирования збитовой последовательности



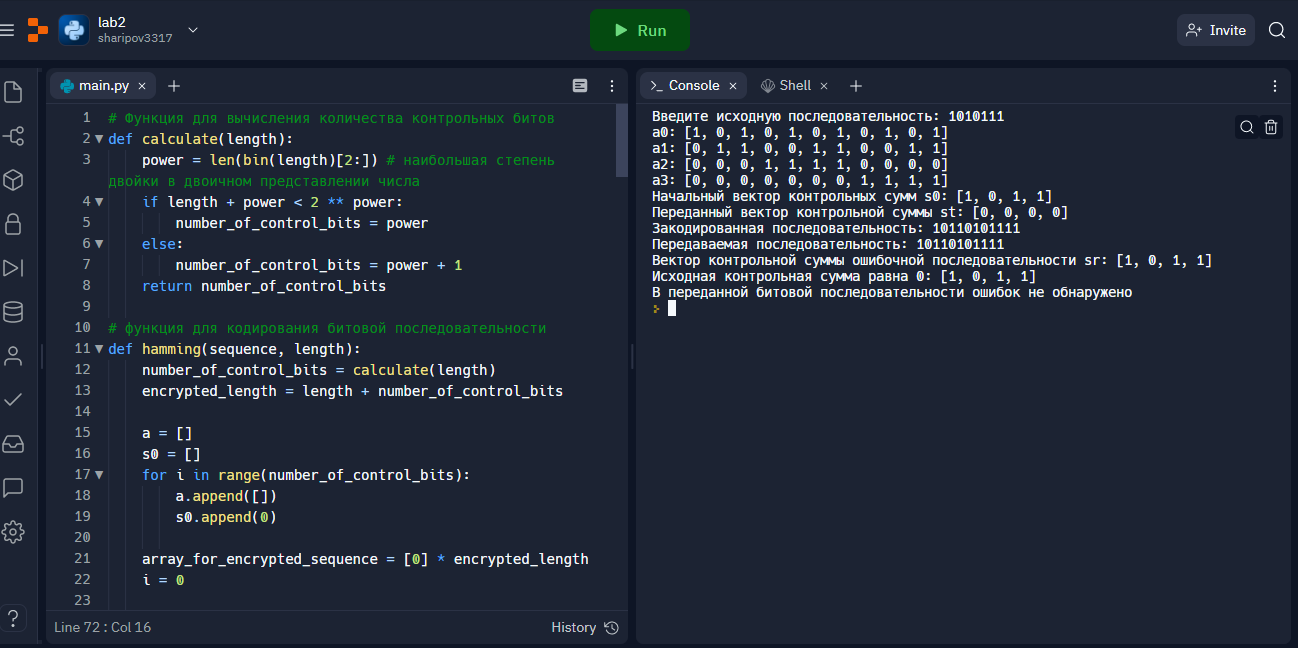
**• Функция расчета кол-ва контрольных бит:**

**• Фрагмент функции проверки передаваемой битовой последовательности:**

# **4 . Первом Пример**

# • Пусть имеется входная последовательность: 1010111

# • В первом случае, не будем допускать ошибку в передаваемой последовательности. Программа должна сообщить нам об этом:

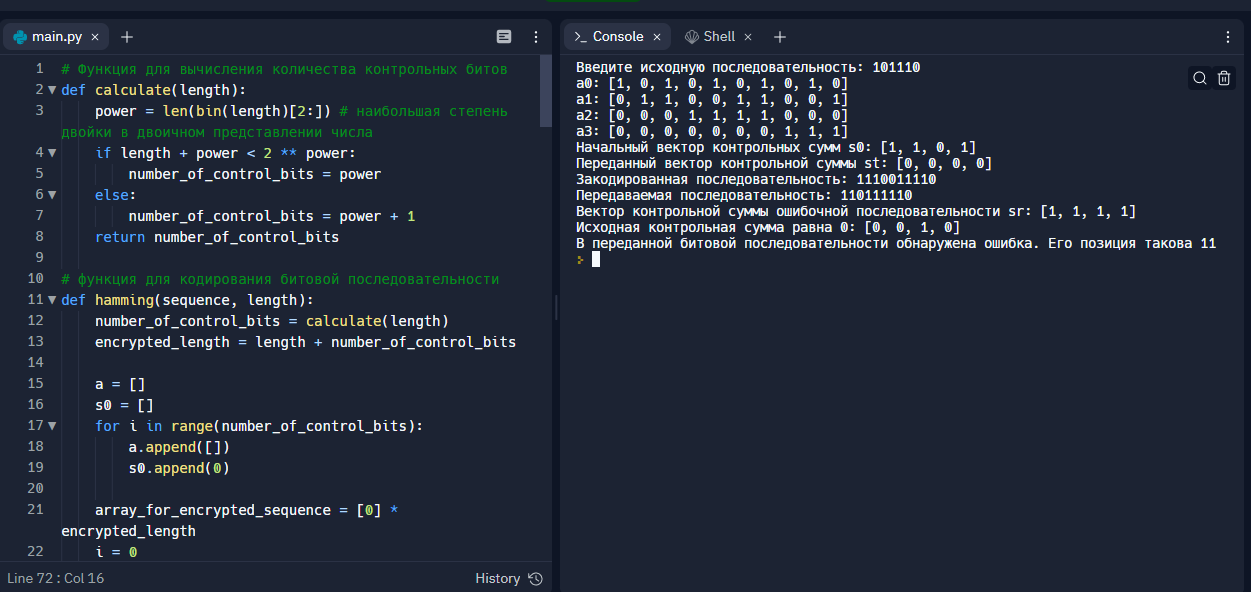
****

# • Втором Пример

# • Пусть имеется входная последовательность: 101110

# • В втором случае, допускается ошибка в передаваемой последовательности.

# Программа должна обнаружить ее и напечатать позицию(11)



**Заключение**

# Предложенные Хэммингом регулярные методы построения кодов, корректирующих ошибки, имеют фундаментальное значение.

# Они демонстрируют инженерам практическую возможность достижения пределов, на которую указывали законы теории информации.

# Эти коды нашли практическое применение при создании компьютерных систем. Работа Хэмминга привела к решению проблемы более плотной упаковки для конечных полей.

# Он ввел в научный обиход важнейшие понятия теории кодирования.

# Работа Хэмминга сыграла ключевую роль в последующем развитии теории кодирования и стимулировала обширные исследования, выполненные в последующие годы.

# Коды Хэмминга позволяют не только обнаружить наличие ошибки, но и место ее нахождения и, следовательно, дают возможность ее исправить.