

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
ФАКУЛЬТЕТ ЭЛЕКТРОННО-ИНФОРМАЦИОННЫХ СИСТЕМ
Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №1

Специальность ПО11

Выполнил
С. С. Жватель
студент группы ПО11

Проверил
А. А. Крощенко
ст. преп. кафедры ИИТ,
12.04.2025 г.

Брест 2025

Цель работы: закрепить базовые знания языка программирования Python при решении практических задач

Задание 1. Для последовательности из N целых чисел реализовать обработку: Распределение чисел: количество одноцифровых чисел, двуцифровых, трехцифровых и т. д.

Выполнение:

Код программы:

```
"""Module for counting digit distribution in a sequence of numbers."""
```

```
def count_digits_distribution(sequence):
```

```
    """Count the number of digits in each number of the sequence.
```

```
    Args:
```

```
    Returns:
```

```
        dict: A dictionary mapping the number of digits to their frequency.
```

```
    """
```

```
    distribution = {}
```

```
    for num in sequence:
```

```
        digits = len(str(abs(num)))
```

```
        distribution[digits] = distribution.get(digits, 0) + 1
```

```
    return distribution
```

```
print("Введите количество чисел в последовательности:")
```

```
n = int(input())
```

```
input_sequence = []
```

```
print(f"Введите {n} целых чисел (каждое на новой строке):")
```

```
for _ in range(n):
```

```
    input_num = int(input())
```

```
    input_sequence.append(input_num)
```

```
result = count_digits_distribution(input_sequence)
```

```
for num_digits, count in sorted(result.items()):
```

```
    print(f"Чисел с {num_digits} цифрой(ами): {count}")
```

Спецификация ввода:

<количество элементов>

<1-й элемент>

...

<n-й элемент>

Пример:

Введите количество чисел в последовательности: 3

Введите 3 целых чисел (каждое на новой строке):

10

123

41241

Спецификация вывода:

Чисел с n цифрой(ами): {<Количество чисел>}

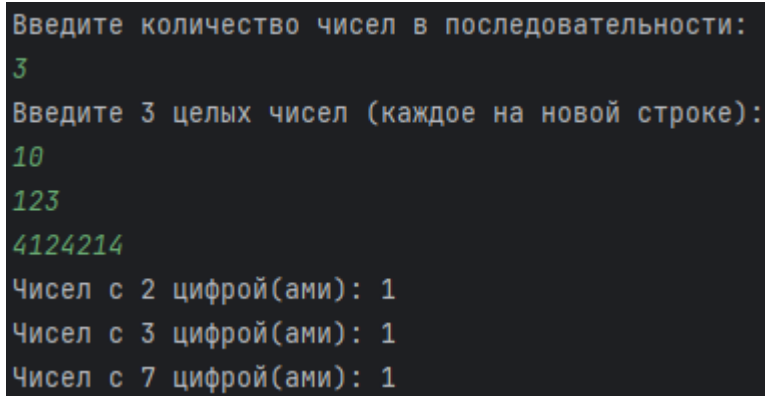
Пример:

Чисел с 2 цифрой(ами): 1

Чисел с 3 цифрой(ами): 1

Чисел с 5 цифрой(ами): 1

Рисунки с результатами работы программы:



```
Введите количество чисел в последовательности:
3
Введите 3 целых чисел (каждое на новой строке):
10
123
4124214
Чисел с 2 цифрой(ами): 1
Чисел с 3 цифрой(ами): 1
Чисел с 7 цифрой(ами): 1
```

Задание 2. Для данного положительного целого числа n напишите функцию, которая возвращает количество установленных битов в его двоичном представлении (также известном как вес Хэмминга).

Выполнение:

Код программы:

```
"""Module for calculating the Hamming weight of positive integers."""

def hamming_weight(n):
    """Calculate the Hamming weight (number of 1s in binary) of a number.

    Args:
        n (int): A positive integer.

    Returns:
        int: The count of 1s in the binary representation of n.
    """
    return bin(n).count("1")

print("Введите количество тестов:")
t = int(input())

print(f"Введите {t} положительных целых чисел (каждое на новой строке):")
for _ in range(t):
    input_n = int(input())
    print(f"Input: n = {input_n}, Output: {hamming_weight(input_n)}")
```

Спецификация ввода:

<количество элементов>

<1-й элемент>

...

<n-й элемент>

Пример:

Введите количество тестов:

2

Введите 2 положительных целых чисел (каждое на новой строке):

10

21

Спецификация вывода:

Input: n = <число>, Output: <количество установленных битов>

Пример:

Input: n = 10, Output: 2

Input: n = 21, Output: 3

Рисунки с результатами работы программы:

```
Введите количество тестов:
2
Введите 2 положительных целых чисел (каждое на новой строке):
10
Input: n = 10, Output: 2
21
Input: n = 21, Output: 3
```

Вывод: закрепил базовые знания Python при решении практических задач