

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ

«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ ЭЛЕКТРОННО-ИНФОРМАЦИОННЫХ СИСТЕМ

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №1

Специальность ПО-12

Выполнил

С. Э. Михновец,

студент группы ПО-12

Проверила

А. Д. Кулик,

преп. стаж. кафедры ИИТ,

«___ к _____ 2026 г.

Брест 2026

Цель работы: научиться разрабатывать простейшие программы на языке программирования Python.

Задание 1. Для последовательности из N целых чисел реализовать обработку:

Для данной последовательности, выводит значение равны, если все элементы последовательности равны и не равны, в противном случае.

Выполнение:

```
"""Модуль для работы со списками: создание, сравнение, добавление и удаление элементов."""
```

```
def createlist() -> list[int]:
```

```
    """
```

```
    Позволяет создать список и заполнить его целыми числами.
```

```
    :return: список целых чисел
```

```
    """
```

```
    objects = []
```

```
    try:
```

```
        n = int(input("Число объектов в списке: "))
```

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

```
            element = int(input(f"Введите элемент {i+1}: "))
```

```
            objects.append(element)
```

```
            print(f"Элемент {element} был добавлен в список.")
```

```
            print(f"Итоговый список: {objects}")
```

```
    except ValueError:
```

```
        print("Ошибка: нужно вводить только целые числа!")
```

```
    return objects
```

```
def isequal(objects: list[int]) -> str:
```

```
"""
Проверяет, равны ли элементы списка.

:param objects: Список целочисленных элементов
:return: Стока с результатом сравнения
"""

if len(objects) < 2:
    print("Нужно минимум 2 элемента для сравнения!")
    return "меньше двух."

if len(set(objects)) == 1:
    print("Все элементы в списке равны.")
    return "равны"

print("В списке есть разные элементы.")
return "неравны"

def add_item(objects: list[int]):
    """Добавляет элемент в список."""
    print("Вы выбрали: добавление")
    try:
        element = int(input("Введите элемент: "))
        objects.append(element)
        print(f"Элемент {element} был добавлен в список.")
    except ValueError:
        print("Ошибка: нужно вводить только целые числа!")

def remove_item(objects: list[int]):
    """Удаляет элемент из списка по индексу."""
    print("Вы выбрали: удаление")
```

```
try:
    idx = int(input("Введите номер элемента: ")) - 1

    try:
        removed = objects.pop(idx)

        print(f"Элемент {removed} под номером {idx + 1} был удален.")

    except IndexError:

        print("Индекс вышел за пределы списка!")

    except ValueError:

        print("Ошибка: нужно вводить только целые числа!")

def main():

    """Основная функция управления программой."""

    objects: list[int] = []

    while True:

        print("\n--- Хелло! ---")

        print("1. Показать список\n2. Создать список\n3. Добавить элемент")
        print("4. Удалить элемент\n5. Сравнить элементы\n6. Выход")

        choice = input("\nВведите выбор меню: ")

        if choice == "1":

            print(f"Список: {objects}")

        elif choice == "2":

            objects = createlist()

        elif choice == "3":

            add_item(objects)

        elif choice == "4":

            remove_item(objects)

        elif choice == "5":

            result = isequal(objects)
```

```
print(f"Элементы(ов) {result}.")  
  
elif choice == "6":  
  
    print("Гу ба-и!")  
  
    break  
  
else:  
  
    print("Ошибка! Введите число от 1 до 6.")
```

```
if __name__ == "__main__":  
  
    main()
```

Рисунки с результатами работы программы

- 2. Создать список
- 3. Добавить элемент
- 4. Удалить элемент
- 5. Сравнить элементы
- 6. Выход

```
Введите выбор меню: 5  
Все элементы в списке равны.  
Элементы(ов) равны.
```

Задание 2. Дан список целых чисел `nums` и целое целевое число, вернуть индексы двух

чисел таким образом, чтобы они в сумме давали целевое число.

Input: `nums = [2,7,11,15], target = 9`

Output: `[0,1]`

Выполнение:

```
"""Модуль для решения задачи Two Sum."""
```

```
def twosum(target: int, nums: list[int]) -> list[int]:
```

```
    """
```

Ищет два числа в списке, сумма которых равна `target`.

```

:param target: Искомая сумма

:param nums: Список чисел

:return: Индексы двух чисел

"""

pmap = {}

for index, num in enumerate(nums):

    diff = target - num

    if diff in pmap:

        return [pmap[diff], index]

    pmap[num] = index

return []


def setlist() -> list[int]:

    """Создает список чисел через ввод пользователя."""

    objects = []

    try:

        n = int(input("Число объектов в списке: "))

        for i in range(n):

            element = int(input(f"Введите элемент {i+1}: "))

            objects.append(element)

            print(f"Элемент {element} был добавлен в список.")

        print(f"Итоговый список: {objects}")

    except ValueError:

        print("Ошибка: нужно вводить только целые числа!")

    return objects


def settarget() -> int:

    """Запрашивает целевое число."""

    try:

        return int(input("Целевое число: "))

    
```

```
except ValueError:

    print("Ошибка: нужно вводить только целые числа!")

    return 0


def main():

    """Запуск алгоритма Two Sum."""

    nums = setlist()

    target = settarget()

    result = twosum(target, nums)

    print(f"Результат: {result}")

if __name__ == "__main__":

    main()
```

Рисунки с результатами работы программы

```
PS D:\SPP> & C:/Users/Admin/AppData/Local/Programs/Python/Python311/python.exe d:/SPP/repo/spp_po12v15/reports/Mihnovec_S_E/Lab_1/src/Lab1_2.py
Итоговый список: [2, 7]
Введите элемент 3: 11
Элемент 11 был добавлен в список.
Итоговый список: [2, 7, 11]
Введите элемент 4: 15
Элемент 15 был добавлен в список.
Итоговый список: [2, 7, 11, 15]
Целевое число: 9
Результат: [0, 1]
PS D:\SPP>
```

Вывод: научился разрабатывать простейшие программы на Python.