

Лабораторная работа №3

Цель работы: изучить понятия итератора и генератора в Python, а также их преимущества; ознакомиться с примерами их пользования.

Ход работы:

1. Постановка задачи: Необходимо создать функцию, которая принимает список из n целых чисел и возвращает новый список с удвоенными элементами.
2. Реализация:
 - В функции используется генераторное выражение для удвоения каждого элемента исходного списка.
 - Сначала программа запрашивает у пользователя количество элементов в списке и сами элементы.
 - Далее создается новый список, в котором каждый элемент исходного списка умножен на 2.
 - Результат выводится на экран.
3. Проверка работы:
 - Программа корректно обрабатывает любой список целых чисел и возвращает его удвоенные значения.
 - Программа тестируется с разным количеством и значениями элементов.

Скриншоты программы с работоспособностью программы продемонстрированы на рисунке 1.

```
Введите количество элементов списка: 3
Введите элемент 1: 2
Введите элемент 2: 6
Введите элемент 3: 3
Удвоенные элементы списка: [4, 12, 6]
```

Рис.1.

Вывод:

В результате работы была реализована функция, которая с помощью генераторного выражения эффективно обрабатывает целочисленный список и возвращает новый список, где каждый элемент удвоен. Программа работает корректно, обрабатывает ввод и выводит результат в ожидаемом формате.

Листинг программы:

```
def double_elements(input_list):  
    return [x * 2 for x in input_list]  
n = int(input("Введите количество элементов списка: "))  
input_list = [int(input(f"Введите элемент {i + 1}: ")) for i in range(n)]  
doubled_list = double_elements(input_list)  
print("Удвоенные элементы списка:", doubled_list)
```