

Практическое задание №11

Тема: Составление программ для работы с элементами функциональным программированием IDE PyCharm Community.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с элементами функциональным программированием в IDE PyCharm Community.

Постановка задачи №1:

Организовать и вывести последовательность A из n чисел (n - четное). Из последовательности A получить две последовательности B и C: в последовательности B – первая половина элементов A, в C – вторая половина элементов A. Найти произведение соответствующих элементов последовательностей B и C. Найти среднее арифметической полученной последовательности.

Тип алгоритма: циклический

Текст программы:

```
import random
try:
    colwo = int(input('Введите сколько будет чисел в последовательности(Четное количество): '))
    if colwo % 2 == 0:
        list_ = [random.randint(0, 20) for i in range(colwo)]
        print('Исходная последовательность', list_)

        listB = list_[:len(list_)//2]
        listC = list_[len(list_)//2:]
        print(listB, listC)

        A = [x * y for x, y in zip(listB, listC)]
        B = sum(A)/len(A)
        print('Произведение соответствующих элементов последовательностей', A)
        print('Среднее арифметическое полученной последовательности', B)
    except:
        print('Неверные данные')
```

Протокол работы программы:

Введите сколько будет чисел в последовательности(Четное количество): 12

Исходная последовательность [6, 4, 10, 15, 0, 13, 13, 0, 6, 4, 3, 20]

[6, 4, 10, 15, 0, 13] [13, 0, 6, 4, 3, 20]

Произведение соответствующих элементов последовательностей [78, 0, 60, 60, 0, 260]

Среднее арифметическое полученной последовательности 76.33333333333333

Process finished with exit code 0

Постановка задачи №2:

Составить список, в который будут включены только согласные буквы и привести их к верхнему регистру. Список: ['Оттава', 'Москва', 'Пекин', 'Полоцк', 'Версаль', 'Дели', 'Каир']

Тип алгоритма: циклический

Текст программы:

```
def cogl(towns):
    for town in towns:
        for bykva in town:
            if bykva.upper() not in ['А', 'Е', 'Ё', 'И', 'О', 'У', 'Ы', 'Э', 'Ю', 'Я']:
                yield bykva.upper()

towns = ['Оттава', 'Москва', 'Пекин', 'Полоцк', 'Версаль', 'Дели', 'Каир']
print(list(cogl(towns)))
```

Протокол работы программы:

['Т', 'Т', 'В', 'М', 'С', 'К', 'В', 'П', 'К', 'Н', 'П', 'Л', 'Ц', 'К', 'В', 'Р', 'С', 'Л', 'Б', 'Д', 'Л', 'К', 'Р']

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с элементами функционального программирования в IDE PyCharm Community.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.