

Практическое занятие №6

Тема: составление программ со списками в IDE PyCharmCommunity.

Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со списками в IDE PyCharmCommunity.

Постановка задачи №1: Дан целочисленный список размера 10. Вывести все содержащиеся в данном списке четные числа в порядке убывания их индексов, а также их количество K.

Тип алгоритма: циклический

Текст программы:

Дан целочисленный список размера 10. Вывести все содержащиеся в данном списке

четные числа в порядке убывания их индексов, а также их количество K.

#Вариант 8

```
from random import randint
```

```
my_list = [randint(-100, 100) for i in range(10)] #создание целочисленного списка случайных чисел
```

```
k = 0
```

```
for el in reversed(my_list): #вывод всех чётных чисел
```

```
    if el % 2 == 0:
```

```
        print(el)
```

```
        k += 1
```

```
print(k)
```

Протокол работы программы:

-70

94

-44

-60

12

-92

6

Process finished with exit code 0

Постановка задачи №2: Дан список размера N. Найти количество участков, на которых его элементы монотонно возрастают.

Тип алгоритма: циклический

Текст программы:

#Дан список размера N. Найти количество участков, на которых его элементы

#монотонно возрастают.

#Вариант 8

```
from random import randint
```

```
n = int(input('Введите n: ')) #ввод исходных данных
```

```
my_list = [randint(-100, 100) for i in range(n)] #создание списка случайных чисел с размером n
```

```
print(my_list)
```

```
buff = "
```

```

for i in range(1, len(my_list)):

    if my_list[i] > my_list[i-1]: #сравнение числе в списке

        buff += '+'

    elif my_list[i] < my_list[i-1]:

        buff += '-'

k = 0

for el in buff.split('-'): #нахождение монотонно возрастающих участков

    if '+' in el:

        k += 1

print(k)

```

Протокол работы программы:

Введите n: 5

[97, 99, -74, -21, -98]

2

Process finished with exit code 0

Постановка задачи №3: Дан список размера N. Заменить каждый элемент списка на среднее арифметическое этого элемента и его соседей.

Тип алгоритма: циклический

Текст программы:

```

#Дан список размера N. Заменить каждый элемент списка на среднее
арифметическое
#этого элемента и его соседей.
#Вариант 8

from random import randint

n = int(input('Введите n: ')) #ввод исходных данных
my_list = [randint(-100, 100) for i in range(n)] #создание списка с
случайными числами размером n
my_list_copy = my_list.copy() #клонирование списка

```

```

for i in range(len(my_list)):
    if i == 0:
        my_list_copy[i] = (my_list[i] + my_list[i + 1]) / 2
    elif i == len(my_list) - 1:
        my_list_copy[i] = (my_list[i-1] + my_list[i]) / 2
#нахождение среднего арифметического
    else:
        my_list_copy[i] = (my_list[i-1] + my_list[i] + my_list[i+1]) /
3
print(my_list) #вывод конечных данных
print(my_list_copy)

```

Протокол работы программы:

Введите n: 6

[-48, -47, 100, 8, 81, 31]

[-47.5, 1.6666666666666667, 20.333333333333332, 63.0, 40.0, 56.0]

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ со списками в IDE PyCharm Community. Выполнена разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.