

Тема: составление программы со списками в IDE PyCharm Community.

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

Постановка задачи: Дан целочисленный список размера 10. Вывести все содержимое в данном списке четные числа в порядке убывания их индексов, а так же их количество K.

Текст программы:

```
import random

lst = [random.randint(-100, 100) for _ in range(10)]

print("Исходный список:")

print(lst)

new_lst = [num for i, num in reversed(list(enumerate(lst))) if num % 2 == 0]

print("Новый список с четными числами (в порядке убывания их индексов):")

print(new_lst)

K = len(new_lst)

print("Количество четных чисел: ", K)
```

Постановка задачи: Дан список размера N. Найти количество участков, на которых его элементы монотонно возрастают.

Текст программы:

```
import random

def validate_input(prompt):

    while True:

        try:

            value = int(input(prompt))

            return value

        except ValueError:

            print("Некорректный ввод. Пожалуйста, введите целое число.")

N = validate_input("Введите размер списка: ")

lst = []

for _ in range(N):

    lst.append(random.randint(1, 100))
```

```
count = 0
```

```
try:
```

```
    for i in range(1, len(lst)):
```

```
        if lst[i] > lst[i - 1]:
```

```
            count += 1
```

```
except IndexError:
```

```
    print("Возникла ошибка индекса списка")
```

```
# Вывести результаты
```

```
print("Сгенерированный список:", lst)
```

```
print("Количество участков с монотонно возрастающими элементами:", count)
```

Постановка задачи: Дан список размера N. Заменить каждый элемент списка на среднее арифметическое этого элемента и его соседей.

Текст программы:

```
try:
    N = int(input("Введите длину списка: "))
    if N <= 0:
        raise ValueError("Длина списка должна быть положительным числом")
except ValueError as e:
    print("Ошибка:", e)
else:
    numbers = []
    for i in range(N):
        try:
            num = float(input("Введите число для элемента {}: ".format(i+1)))
            numbers.append(num)
        except ValueError:
            print("Ошибка ввода! Пожалуйста, введите число.")

    print("Исходный список:", numbers)

def calculate_average(lst, i):
    total = lst[i]
    count = 1
    if i > 0:
        total += lst[i-1]
        count += 1
    if i < len(lst)-1:
        total += lst[i+1]
        count += 1
```

```
return total / count

for i in range(len(numbers)):
    numbers[i] = calculate_average(numbers, i)

print("Измененный список:", numbers)
```