

Практическое занятие №5

Тема: составление программ со списками в IDE PyCharm Community.

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

Постановление задачи №1: дан целочисленный список размера N, не содержащий одинаковых чисел. Проверить, образуют ли его элементы арифметическую прогрессию. Если образуют, то вывести разность прогрессии, если нет — вывести 0.

Тип алгоритма: алгоритм с функциями.

Код алгоритма №1:

```
def arithmetic(arr):  
    n = len(arr)  
  
    if n < 2:  
        return 0  
  
    difference = arr[1] - arr[0]  
  
    for i in range(2, n):  
        if arr[i] - arr[i-1] != difference:  
            return 0  
  
    return difference  
  
my_list = [2, 4, 6, 8, 10]  
result = arithmetic(my_list)  
  
if result != 0:  
    print(f"Элементы образуют арифметическую прогрессию с разностью  
{result}.")  
else:  
    print("Элементы не образуют арифметическую прогрессию.")
```

Ожидаемый вывод:

Элементы образуют арифметическую прогрессию с разностью 2.

Постановление задачи №2: Дан список A размера N. Сформировать новый список B того же размера, элементы которого определяются следующим образом: $BK = 2 \cdot AK$, если $AK < 5$, $AK/2$ в противном случае.

Тип алгоритма: алгоритм с функциями.

Код алгоритма №2:

```
def spisok(arr):
    arr2 = []
    for i in arr:
        if i < 5:
            b = i * 2
        else:
            b = i / 2
        arr2.append(b)
    return arr2

arr = [2, 3, 5, 1, 7]
arr2 = spisok(arr)
print(f'Изначальный список: {arr}')
print(f'Новый список: {arr2}')
```

Ожидаемый вывод:

Изначальный список: [2, 3, 5, 1, 7]

Новый список: [4, 6, 2.5, 2, 3.5]

Постановление задачи №3: Дано множество A из N точек (точки заданы своими координатами x, y). Среди всех точек этого множества, лежащих в первой или третьей четверти, найти точку, наиболее близкую к началу координат. Если таких точек нет, то вывести точку с нулевыми координатами. Расстояние R между точками с координатами (x_1, y_1) и (x_2, y_2) вычисляется по формуле: $R = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Тип алгоритма: алгоритм с функциями.

Код алгоритма №3:

```
import math

def closest_point(x_list, y_list):
    closest_distance = float('inf')
    closest_point = (0, 0)
```

```
for i in range(len(x_list)):
    if x_list[i] >= 0 and y_list[i] >= 0:
        distance = math.sqrt(x_list[i] ** 2 + y_list[i] ** 2)
        if distance < closest_distance:
            closest_distance = distance
            closest_point = (x_list[i], y_list[i])

if closest_point == (0, 0):
    return "Нет точек в первой или третьей четверти"

return closest_point

x = [0.23, 0, 0.3, 1, 2, 8]
y = [0.243, 1, 0.1, 3, 1, 6]
print(closest_point(x, y))
```

Ожидаемый вывод:

Точки которые ближе всего лежат к началу координат: (0.3, 0.1)

Вывод: я закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ со списками в IDE PyCharm Community.