

Тема: Составление программ со списками в IDE PyCharm Community.

Цель: Закрепить усвоенные знания ,понятия , алгоритмы ,основные принципы составления программ, приобрести навыки составления программ с функциями в IDE PyCharm Community.

Постановка задачи №1: Дан целочисленный список A размера 10. Вывести порядковый номер последнего из тех его элементов A_k , которые удовлетворяют двойному неравенству $A_1 < A_k < A_{10}$. Если таких элементов нет, то вывести 0.

```
# Дан целочисленный список A размера 10.
# Вывести порядковый номер последнего из тех его элементов  $A_k$ ,
# которые удовлетворяют двойному неравенству  $A_1 < A_k < A_{10}$ .
# Если таких элементов нет, то вывести 0.

def find_last_element_index(A):
    for i in range(len(A)-1, 0, -1):
        if A[0] < A[i] < A[-1]:
            return i
    return 0

A = [5, 2, 8, 4, 6, 10, 12, 3, 7, 9] # Пример целочисленного
списка A
result = find_last_element_index(A)
print("Порядковый номер последнего элемента, удовлетворяющего
условию:", result)
```

Протокол работы:

Порядковый номер последнего элемента, удовлетворяющего условию: 8

Постановка задачи №2: Дано число R и список размера N. Найти два различных элемента списка, сумма которых наиболее близка к числу R, и вывести эти элементы в порядке возрастания их индексов (определение наиболее близких чисел - то есть такой элемент A_k , для которого величина $|A_k - R|$ является минимальной).

```
# Дано число R и список размера N. Найти два различных элемента
списка, сумма которых наиболее близка к числу R,
# и вывести эти элементы в порядке возрастания их индексов
(определение наиболее близких чисел - то есть такой элемент  $A_k$ ,
# для которого величина  $|A_k - R|$  является минимальной) .
```

```
def find_closest_sum_pair(arr, R):
    closest_sum = float('inf')
    pair = ()
    for i in range(len(arr)):
        for j in range(i+1, len(arr)):
            current_sum = arr[i] + arr[j]
            if abs(current_sum - R) < abs(closest_sum - R):
                closest_sum = current_sum
                pair = (i, j)
    if pair:
        return [arr[pair[0]], arr[pair[1]]]
    else:
        return "Нет двух различных элементов, сумма
которых близка к числу R."

arr = [4, 6, 8, 2, 3, 5, 7, 9] # Пример целочисленного списка arr
R = 10 # Пример числа R
result = find_closest_sum_pair(arr, R)
print("Два различных элемента списка, сумма которых наиболее
близка к числу R:", result)
```

Протокол работы:

Два различных элемента списка, сумма которых наиболее близка к числу R: [4, 6]

Постановка задачи №3: Дан список размера N, все элементы которого, кроме первого, упорядочены по возрастанию. Сделать список упорядоченным, переместив первый элемент на новую позицию.

```
# Дан список размера N, все элементы которого, кроме первого,
упорядочены по возрастанию.
# Сделать список упорядоченным, переместив первый элемент на новую
позицию.

def reorder_list(arr):
    if len(arr) < 2:
        return arr

    first_element = arr[0]
    arr = arr[1:]
    pos = 0
    for i in range(len(arr)):
        if arr[i] < first_element:
            pos = i + 1
    arr.insert(pos, first_element)

    return arr
```

```
# Пример
arr = [3, 5, 7, 6, 8, 9]
result = reorder_list(arr)
print(result) # Вывод: [3, 5, 6, 7, 8, 9]
```

Протокол работы:

[3, 5, 7, 6, 8, 9]

Общий вывод: Я закрепил усвоенные знания ,понятия , алгоритмы ,основные принципы составления программ и приобрел навыки составления программ с функциями в IDE PyCharm Community.