

Практическое занятие № 6

Тема: Составление программ 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:

1. Дан целочисленный список A размера 10. Вывести порядковый номер последнего из тех его элементов AK, которые удовлетворяют двойному неравенству $A_1 < A_K < A_{10}$. Если таких элементов нет, то вывести 0.
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 AK, для которого величина $|AK - R|$ является минимальной).
3. Дан список размера N, все элементы которого, кроме первого, упорядочены по возрастанию. Сделать список упорядоченным, переместив первый элемент на новую позицию.

Код программы:

```
# Дан целочисленный список A размера 10. Вывести порядковый номер последнего
из
# тех его элементов AK, которые удовлетворяют двойному неравенству  $A_1 < A_K < A_{10}$ .
# Если таких элементов нет, то вывести 0.
A = list(map(int, input("Введите 10 целых чисел через пробел: ").split()))
last_index = 0
try:
    for i in range(10):
        if A[0] < A[i] < A[-1]:
            last_index = i
except IndexError:
    print("Список пуст или содержит недостаточно элементов.")

if last_index != 0:
    print("Порядковый номер последнего из элементов, удовлетворяющих
условию:", last_index + 1)
else:
    print("0")
print("Исходный список", A)
```

Протокол работы программы:

Введите 10 целых чисел через пробел: 1 2 3 4 5 6 7 8 9 10

Порядковый номер последнего из элементов, удовлетворяющих условию: 9

Исходный список [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

Process finished with exit code 0

Код программы:

```
# # 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 AK,
# для
# которого величина |AK - R| является минимальной).
def find_closest_elements(R, lst):
    closest_sum = None
    closest_indices = []
    try:
        for i in range(len(lst)):
            for j in range(i + 1, len(lst)):
                if closest_sum is None or abs(lst[i] + lst[j] - R) <
abs(closest_sum - R):
                    closest_sum = lst[i] + lst[j]
                    closest_indices = [i, j]
        return closest_indices
    except Exception as e:
        print("Ошибка: ", e)

R = int(input("Введите число R: "))
lst = list(map(int, input("Введите список чисел через пробел: ").split()))
closest_indices = find_closest_elements(R, lst)
if closest_indices:
    print("Ближайшие элементы:", lst[closest_indices[0]],
lst[closest_indices[1]])
else:
    print("Список пуст или содержит только одно число.")
print("Исходный список",lst)
```

Протокол работы программы:

Введите число R: 3

Введите список чисел через пробел: 1 2 3 4 5 6

Ближайшие элементы: 1 2

Исходный список [1, 2, 3, 4, 5, 6]

Process finished with exit code 0

Код программы:

```
# Дан список размера N, все элементы которого, кроме первого, упорядочены по
# возрастанию. Сделать список упорядоченным, переместив первый элемент на
# новую
# позицию.
lst = input("Введите элементы списка через пробел: ").split()
print("Исходный список", lst)
if not lst:
    print("Список пуст")
else:
    try:
        lst.sort()
    except Exception as e:
        print("Ошибка при перемещении элемента:", e)
print("Преобразованный список", lst)
```

Протокол работы программы:

Введите элементы списка через пробел: 2 1 3 4 5 6

Исходный список ['2', '1', '3', '4', '5', '6']

Преобразованный список ['1', '2', '3', '4', '5', '6']

Process finished with exit code 0

Вывод: Я закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ и приобрела навыки составления программ со списками в IDE PyCharm Community. А также выложила отчет и код программы на платформу GitHub.