

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) Разработать программу, в которой дан целочисленный список A размера 10. Программа выводит порядковый номер последнего из тех его элементов A_K , которые удовлетворяют двойному неравенству $A_1 < A_K < A_{10}$. Если таких элементов нет, то выводит 0.
- 2) Разработать программу, в которой дано число R и список размера N . Программа находит два различных элемента списка, сумма которых наиболее близка к числу R , и выводит 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).
- 3) Разработать программу, в которой дан список размера N , все элементы которого, кроме первого, упорядочены по возрастанию. Программа делает список упорядоченным, перемещает первый элемент на новую позицию.

Текст программы:

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.

```
from random import randint # библиотека дает возможность генерировать случайные числа
a = [randint(1, 25) for i in range(10)]
num = 0
```

```

print(a)
for j in range(9):
    if a[0] < a[j] < a[9]:
        num = a[j]

print(num)  # вывод данных на консоль

```

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).

```

from random import randint
r = int(input('Введите число R:')) # ввод данных с клавиатуры
n = int(input('Введите размер списка:')) # ввод данных с клавиатуры
a = [randint(1, 100) for i in range(n)] # заполнение списка
print('Список:', a) # вывод данных на консоль
min = abs(r - (a[0] + a[1])) # Минимальное обозначение суммы элементов
i_m = 1
for i in range(n): # вычисление суммы и нахождение элементов близких к R
    d = abs(r - (a[i-1] + a[i]))
    if min > d:
        min = d
        i_m = i
if a.index(a[i_m-1]) < a.index(a[i_m]): # вывод элементов в порядке возрастания
    print('Элементы, сумма которых близка к R:', a[i_m-1], (a[i_m])) # вывод данных на консоль
else:
    print('Элементы, сумма которых близка к R:', a[i_m], (a[i_m-1])) # вывод данных на консоль

```

3)

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

```
from random import randint # библиотека дает возможность генерировать
случайные числа
print('Введите размер списка:')
a = [randint(1, 100) for i in range(int(input('')))]
print('Список:', a) # вывод данных на консоль
a.sort() # сортирует все элементы по возрастанию
print('Упорядоченный список:', a) # вывод данных на консоль
a += [a.pop(0)] # перемещает первый элемент на новую позицию
print('Упорядоченный список с перемещённым элементом:', a) # вывод
данных на консоль
```

Протокол работы программы:

1)

[4, 8, 2, 21, 4, 4, 2, 15, 22, 18]
15

Process finished with exit code 0

2)

Введите число R:65
Введите размер списка:7
Список: [94, 89, 64, 4, 24, 40, 45]
Элементы, сумма которых близка к R: 24 40

Process finished with exit code 0

3)

Введите размер списка:
10
Список: [51, 52, 83, 33, 15, 48, 75, 12, 91, 56]
Упорядоченный список: [12, 15, 33, 48, 51, 52, 56, 75, 83, 91]
Упорядоченный список с перемещённым элементом: [15, 33, 48, 51, 52, 56, 75, 83, 91, 12]

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия я закрепила усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрела навыки составления программ со списками в IDE PyCharm Community.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.