

## 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:

#### Задание №1

Дан список размера N и целые числа K и L ( $1 < K < L < N$ ). Найти сумму всех элементов списка, кроме элементов с номерами от K до L включительно.

#### Задание №2

Дан целочисленный список размера N. Если он является перестановкой, то есть содержит все числа от 1 до N, то вывести 0; в противном случае вывести номер первого недопустимого элемента.

#### Задание №3

Дано множество A из N точек на плоскости и точка B (точки заданы своими координатами x, y). Найти точку из множества A, наиболее близкую к точке B. Расстояние R между точками с координатами (x1, y1) и (x2, y2) 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.

### Текст программы:

#### Задание №1

#Дан список размера N и целые числа K и L ( $1 < K < L < N$ ). Найти сумму всех элементов списка, кроме элементов с номерами от K до L включительно.

```
import random
a = []
h = []
try:
    N = int(input("Введите количество элементов в списке: "))
    K = int(input("Введите L: "))
    L = int(input("Введите K: "))
except ValueError:
    print("Введите число заново")
    N = int(input("Введите количество элементов в списке: "))
    K = int(input("Введите L: "))
    L = int(input("Введите K: "))
for i in range(1, N):
    i += 2
    a.append(i)
    sum(a)
    for q in range(K+1, L-1):
        h.append(q)
```

```
sum(h)
if K > 1 and K < L and L < N:
    p = sum(a)-sum(h)
    print(p)
```

### Протокол работы программы:

Введите количество элементов в списке: 6  
Введите L: 2  
Введите K: 3  
25

Process finished with exit code 0

### Задание №2

# Дан целочисленный список размера N. Если он является перестановкой, то есть  
# содержит все числа от 1 до N, то вывести 0; в противном случае вывести номер  
# первого недопустимого элемента.

try:

```
N = int(input("Введите количество элементов массива: "))
```

except ValueError:

```
    print("Введите число заново")
```

```
    N = int(input("Введите количество элементов массива: "))
```

```
a = [int(i) for i in range(1, N + 1)]
```

for i in range(N):

```
    print("Введите элемент массива ", i + 1, ": ")
```

```
    tmp = int(input())
```

```
    if tmp in a :
```

```
        a.pop(a.index(tmp))
```

```
    else :
```

```
        print ('Первая ошибка появилась в символе номер ' + str(i + 1))
```

```
        break
```

```
if len(a) == 0 :
```

```
    print (str(0))
```

### Протокол работы программы:

Введите элемент массива 1 :

1

Введите элемент массива 2 :

2

Введите элемент массива 3 :

3

Введите элемент массива 4 :

4

Введите элемент массива 5 :

5  
0

Process finished with exit code 0

### Задание №3

```
# Дано множество A из N точек на плоскости и точка B (точки заданы своими
# координатами x, y). Найти точку из множества A, наиболее близкую к точке B.
# Расстояние R между точками с координатами (x1, y1) и (x2, y2) 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.
import math
import random
A = []
B = []
C = []
N = random.randint(1,100)
X1 = random.randint(1,100)
Y1 = random.randint(1,100)
X2 = random.randint(1,100)
Y2 = random.randint(1,100)
for X1 in range(1, N+1):
    A.append(X1)
for Y1 in range(1,N+1):
    B.append(Y1)
R = math.sqrt((X2 - X1)**2) + (Y2 - Y1)**2
C.append(R)
print(min(C))
```

### Протокол работы программы:

8473.0

Process finished with exit code 0

### Вывод:

В процессе выполнения работы, 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.