

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

Постановка задачи №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.

Постановка задачи №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: $R = \sqrt{(x2 - x1)^2 + (y2 - y1)^2}$. Для хранения данных о каждом наборе точек следует использовать по два списка: первый список для хранения абсцисс, второй — для хранения ординат.

Тип алгоритма: циклический с использованием функции.

Текст программы №1.

```
"""Практическая работа №6 Складов Владимир Группа ИС-23."""
```

```
"""Постановка задачи
```

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

```
from random import randint
```

```
"""Импортирование из библиотеки random функции randint."""
```

```
def numbers_random(number_1, number_2, number_3): #В функции 3 аргумента
```

```
    list_1 = []
```

```
    for number in range(0, number_1): #Цикл пока индекс в диапазоне списка(от 0 до number_input).
```

```
        list_1.append(randint(1, 100)) # Добавление в список значения от 1 до 100.
```

```
    print('Изначальный список: ', list_1)
```

```
    del list_1[number_3:number_2+1] # Удаление в списке значений от K до L
```

```
    print('Список после удаления индексов от k до L: ', list_1) # Вывод списка после удаления значений от
k до l
```

```
    sums = f'Сумма списка: {sum(list_1)}'
```

```

return sums          # Возвращение СУММЫ списка (итоговое значение после вычета элементов от
k до L)

```

```

"""Ввод/Вывод данных"""

```

```

try:

```

```

    number_input = int(input('Введите размер списка N: '))

```

```

    number_L = int(input('Введите число L, оно должно быть меньше N: '))

```

```

    number_K = int(input('Введите число K, оно должно быть больше 1 и меньше L: '))

```

```

    if number_L >= number_input or number_L <= number_K: # Вывод ошибки если не соответствует
условию.

```

```

        print('Ошибка')

```

```

    elif number_K <= 1 or number_K >= number_L:

```

```

        print("Ошибка")

```

```

    else:

```

```

        print(numbers_random(number_input, number_L, number_K)) # Вызов функции

```

```

except ValueError:

```

```

    print("Ошибка")

```

Текст программы №2.

```

"""21 Вариант Складов Владимир. Группа: ИС-23."""

```

```

"""

```

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."""

```

import random

```

```

"""Импортирование библиотеки random"""

```

```

try:

```

```

    N = int(input('Введите значение: ')) # Переменной присваивается введенное значение

```

```

    a = [random.randrange(1, N+1) for i in range(N)] # В переменной образуются рандомные
значения от 1 до 5 в списке

```

```

    L = sorted(a) # Сортировка списка a

```

```

except ValueError:

```

```

    print("Ошибка")

```

```

else:

```

```

    for i in range(1, N+1): # Цикл проходится по каждому значению от 1 до вводимого значения

```

```

        if i != L[i-1]: # Проверка условия, если i (индекс) не равен списку по индексу i-1

```

```

            break # То произойдет остановка цикла

```

```

    if i == N: # Проверка условия если i == N

```

```

        print(L)

```

```

        print("0") # Вывод 0

```

```

    else: # Условие иначе

```

```

        number = L[0] # переменная number ссылается на первое значение в списке L

```

```

        id = 0

```

```

        m = 1

```

```

        while id < len(L)+1: # Цикл в котором проверяется каждое значение

```

```
if L.count(number) == N:
    print(f"Так как все цифры одинаковые, ответ равняется первому индексу: {L[0]}")
    break # Остановка цикла
elif L[id] < L[m]:
    if L[id] + 1 == L[m]: # Проверка условия, что L[id] < L[m] на 1.
        print(f"Сам список: {L}")
        print(f"Первый недопустимый элемент --> {id+1}")
        break
    elif L[m] >= len(L): # Если все числа не являются последовательностью
        print("Всё в порядке, все числа отличаются.")
        break
id += 1 # Прибавление к счётчику 1
m += 1
```

Текст программы №3.

```
import random
import math
```

""21 Вариант. Складов Владимир. Группа: ИС-23.""

""Постановка задачи.....

Дано множество А из N точек на плоскости и точка В (точки заданы своими координатами x, y). Найти точку из множества А, наиболее близкую к точке В. 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.""

```
def rasstoyanie(x1, x2, y1, y2): ###Функция которая принимает 4 параметра.
    R = math.sqrt(((x2 - x1)**2) + ((y2 - y1)**2)) ###Формула по которой находится расстояние
    list_2.append(R)
```

```
try:
    B = []
    c = int(input('Введите число точек в массиве A: ')) # Ввод кол-ва точек
    list_2 = []
    list_x = [random.randint(0, 100) for i in range(c)] # Добавление в список абсцисс и ординат точек
    list_y = [random.randint(0, 100) for i in range(c)]
```

```
""Ввод данных пользователем в переменную a,b в int формате""
a, b = int(input('Введите x координату для точки B: ')), int(input('Введите y координату для точки B: '))
B.append(a)
B.append(b)
except ValueError: ### Обработчик исключений, который выводит ошибку, в случае если введённое числа a,b не в int типе
```

```
print('Ошибка')
else: ### Условие иначе, где выполняется продолжение блока try
    print('Количество всех точек по x в массиве A:', list_x) ### Вывод списка list_x в котором
хранятся все x
    print('Количество всех точек по y в массиве A:', list_y) ### Вывод списка list_y в котором
хранятся все y
    print(f'Координаты точки B: x->{B[0]}, y->{B[1]}') ### Вывод координаты точки в списке B, x и y.

for i in range(len(list_x)): #Цикл в котором перебираются все значения в списке list_x
    расстояние(B[0], list_x[i], B[-1], list_y[i])
else:
    id = 0 # Счётчик
    while id <= c - 1: # Цикл в котором проверяются все значения в списке list_2 и выводится
ближайшая точка
        if math.sqrt(((list_x[id] - B[0])**2) + ((list_y[id] - B[-1])**2)) == min(list_2):
            print(f"Ближайшая точка к B имеет координаты-> x:{list_x[id]}, y:{list_y[id]}")
        id += 1
```

Протокол работы программы №1.

Ввод:

Введите размер списка N: 8

Введите число L, оно должно быть меньше N: 8

Введите число K, оно должно быть больше 1 и меньше L: 4

Вывод: Ошибка

Ввод:

Введите размер списка N: 5

Введите число L, оно должно быть меньше N: 4

Введите число K, оно должно быть больше 1 и меньше L: 3

Вывод:

Изначальный список: [93, 3, 51, 34, 10, 34, 1, 71]

Список после удаления индексов от k до L: [93, 3, 51,
34, 71]

Сумма списка: 252

Протокол работы программы №2.

Ввод:

Введите значение: asdasdasd

Вывод:

Ошибка

Ввод:

Введите значение: 8

Вывод:

Студент группы ИС-23 Складов В.Д

Сам список: [1, 2, 4, 5, 5, 6, 7, 7]

Первый недопустимый элемент --> 1

Протокол работы программы №3.

Ввод:

Введите размер списка: 6

Введите x координату для точки В: 4

Введите y координату для точки В: 3

Вывод:

Количество всех точек по x в массиве А: [55, 62, 53, 99, 68, 72]

Количество всех точек по y в массиве А: [70, 44, 30, 72, 43, 19]

Координаты точки В: x->4, y->3

Ближайшая точка к В имеет координаты-> x:53, y:30

Ввод:

Введите размер списка: s

Ввод:

Введите x координату для точки В: 6

Введите y координату для точки В: 5

Вывод:

Ошибка

Вывод: в процессе выполнения практического занятия выработал навыки

Студент группы ИС-23 Складов В.Д

составления программ циклической структуры(с использованием циклов и списков) в IDE PyCharm Community. Были использованы языковые конструкции while, if, else, try, except, list, for, def.

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

Готовые программные коды выложены на GitHub.