

Практическая работа № 5

«Списки»

Студента группы ИСиП-31

Калмыкова Александра

Преподаватель

Кондурар Марина Викторовна

Тольятти-2021

Оглавление

[Задание №1: 3](#_Toc70027614)

[Задание №2: 4](#_Toc70027615)

[Задание №3: 5](#_Toc70027616)

[Задание №4: 6](#_Toc70027617)

[Задание №5: 7](#_Toc70027618)

[Задание №6: 8](#_Toc70027619)

[Задание №7: 9](#_Toc70027620)

[Задание №8: 10](#_Toc70027621)

[Задание №9: 11](#_Toc70027622)

[Задание №10: 12](#_Toc70027623)

[Задание №11: 13](#_Toc70027624)

[Задание №13: 14](#_Toc70027625)

[Задание №14: 15](#_Toc70027626)

[Задание №15: 16](#_Toc70027627)

[Задание №16: 17](#_Toc70027628)

[Задание №17: 19](#_Toc70027629)

# Задание №1:

Выведите все элементы списка с четными индексами.

**Входные данные:**

mass[] - string

**Выходные данные:**

mass[i] - string

**Код:**

mass = input("Введите информацию: ").split()

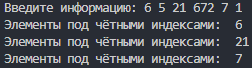
for i in range(0, len(mass), 2):

    print("Элементы под чётными индексами: ", mass[i])

**Контрольный пример:**

mass[6, 5, 21, 672, 7, 1]

**Результат выполнения программы:**



# Задание №2:

Выведите все четные элементы списка. При этом используйте цикл for, перебирающий элементы списка, а не их индексы!

**Входные данные:**

mass[] - string

**Выходные данные:**

i - string

**Код:**

mass=input("Введите информацию:")

massEdit=[int(mass) for mass in mass.split()]

for i in massEdit:

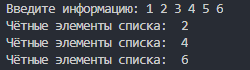
    if int(i)%2 == 0:

       print("Чётные элементы списка: ",i)

**Контрольный пример:**

mass[1,2,3,4,5,6]

**Результат выполнения программы:**



# Задание №3:

Дан список чисел. Выведите все элементы списка, которые больше предыдущего элемента.

**Входные данные:**

mass[] - string

**Выходные данные:**

mass[massEdit] - string

**Код:**

mass = [int(massEdit) for massEdit in input("Введите информацию: ").split()]

for massEdit in range(1, len(mass)):

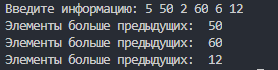
    if mass[massEdit] > mass[massEdit - 1]:

        print("Элементы больше предыдущих: ",mass[massEdit])

**Контрольный пример:**

mass[5,50,2,60,6,12]

**Результат выполнения программы:**



# Задание №4:

Дан список чисел. Если в нем есть два соседних элемента одного знака, выведите эти числа. Если соседних элементов одного знака нет — не выводите ничего. Если таких пар соседей несколько — выведите первую пару.

**Входные данные:**

mass[] – string

**Выходные данные:**

mass[massEdit] - string

**Код:**

mass = [int(massEdit) for massEdit in input("Введите информацию: ").split()]

for massEdit in range(1, len(mass)):

    if mass[massEdit - 1] \* mass[massEdit] > 0:

        print("Итог: ",mass[massEdit - 1], mass[massEdit])

        break

**Контрольный пример:**

mass[50,-2,17,-7,6,-7,2,2]

**Результат выполнения программы:**



# Задание №5:

Дан список чисел. Определите, сколько в этом списке элементов, которые больше двух своих соседей, и выведите количество таких элементов. Крайние элементы списка никогда не учитываются, поскольку у них недостаточно соседей.

**Входные данные:**

mass[] – string

**Математические операции:**

counter = 0

**Выходные данные:**

counter - int

**Код:**

mass = [int(massEdit) for massEdit in input("Введите информацию: ").split()]

countElem = 0

for massEdit in range(1, len(mass) - 1):

    if mass[massEdit - 1] < mass[massEdit] > mass[massEdit + 1]:

        countElem += 1

print("Элементов больше соседей: ",countElem)

**Контрольный пример:**

mass[2,5,7,3]

**Результат выполнения программы:**



# Задание №6:

Дан список чисел. Выведите значение наибольшего элемента в списке, а затем индекс этого элемента в списке. Если наибольших элементов несколько, выведите индекс первого из них.

**Входные данные:**

mass[] – string

**Математические операции:**

indexMax = 0

**Выходные данные:**

mass[indexMax], indexMax - string

**Код:**

indexMax = 0

mass = [int(i) for i in input("Введите информацию: ").split()]

for i in range(1, len(mass)):

    if mass[i] > mass[indexMax]:

        indexMax = i

print("Наибольшее значение: ",mass[indexMax], "Его индекс: " ,indexMax)

**Контрольный пример:**

mass[25,50,70,1,2,7,100]

**Результат выполнения программы:**



# Задание №7:

Петя перешёл в другую школу. На уроке физкультуры ему понадобилось определить своё место в строю. Помогите ему это сделать. Программа получает на вход невозрастающую последовательность натуральных чисел, означающих рост каждого человека в строю. После этого вводится число X – рост Пети. Все числа во входных данных натуральные и не превышают 200.

Выведите номер, под которым Петя должен встать в строй. Если в строю есть люди с одинаковым ростом, таким же, как у Пети, то он должен встать после них.

**Входные данные:**

height - string, X - string

**Математические операции:**

position = 0

**Выходные данные:**

position - int

**Код:**

height = [int(i) for i in input("Введите рост ученика: ").split()]

X = int(input("Введите рост Пети: "))

position = 0

while position < len(height) and height[position] >= X:

    position += 1

print("Позиция Пети в шеренге", position + 1)

**Контрольный пример:**

height[175 172 162 176]

X = 175

**Результат выполнения программы:**



# Задание №8:

Дан список, упорядоченный по не убыванию элементов в нем. Определите, сколько в нем различных элементов.

**Входные данные:**

mass[] - string

**Математические операции:**

countElements = 1

**Выходные данные:**

countElements - int

**Код:**

mass = [int(i) for i in input("Введите информацию:").split()]

countElements = 1

for i in range(0, len(mass) - 1):

    if mass[i] != mass[i + 1]:

        countElements += 1

print("Различных элементов в массиве:", countElements)

**Контрольный пример:**

mass[1 1 2 2 3 3 4 4 5 5 6 7 8 9 10]

**Результат выполнения программы:**



# Задание №9:

Переставьте соседние элементы списка (A[0] c A[1], A[2] c A[3] и т. д.). Если элементов нечетное число, то последний элемент остается на своем месте.

**Входные данные:**

mass[] - string

**Выходные данные:**

mass[i] - string

**Код:**

mass = [int(i) for i in input("Введите информацию: ").split()]

for i in range(1, len(mass), 2):

    mass[i - 1], mass[i] = mass[i], mass[i - 1]

print("Итого: ",' '.join([str(i) for i in mass]))

**Контрольный пример:**

mass[1,2,3,4,5,6,7]

**Результат выполнения программы:**



# Задание №10:

В списке все элементы различны. Поменяйте местами минимальный и максимальный элемент этого списка.

**Входные данные:**

mass[s] - string

**Выходные данные:**

mass[i] - string

**Код:**

mass = [int(s) for s in input("Введите информацию: ").split()]

indexMin = 0

indexMax = 0

for i in range(1, len(mass)):

    if mass[i] > mass[indexMax]:

        indexMax = i

    if mass[i] < mass[indexMin]:

        indexMin = i

mass[indexMin], mass[indexMax] = mass[indexMax], mass[indexMin]

print(' '.join([str(i) for i in mass]))

**Контрольный пример:**

mass[5 6 70 10 80 10 50]

**Результат выполнения программы:**



# Задание №11:

Дан список чисел. Посчитайте, сколько в нем пар элементов, равных друг другу. Считается, что любые два элемента, равные друг другу образуют одну пару, которую необходимо посчитать.

**Входные данные:**

mass[s] – string

**Математические операции:**

counter - int

**Выходные данные:**

counter - int

**Код:**

mass = [int(s) for s in input("Введите информацию: ").split()]

counter = 0

for i in range(len(mass)):

    for j in range(i + 1, len(mass)):

        if mass[i] == mass[j]:

            counter += 1

print("Кол-во совпадений: ",counter)

**Контрольный пример:**

mass[5 2 6 6 1 7 12 27 71 71]

**Результат выполнения программы:**



# Задание №13:

Дан список из чисел и индекс элемента в списке k. Удалите из списка элемент с индексом k, сдвинув влево все элементы, стоящие правее элемента с индексом k. Программа получает на вход список, затем число k. Программа сдвигает все элементы, а после этого удаляет последний элемент списка при помощи метода pop() без параметров. Программа должна осуществлять сдвиг непосредственно в списке, а не делать это при выводе элементов. Также нельзя использовать дополнительный список. Также не следует использовать метод pop() с параметром.

**Входные данные:**

mass[] - string

**Выходные данные:**

mass[] - string

**Код:**

mass = [int(s) for s in input("Введите информацию: ").split()]

k = int(input("Введите индекс, который нужно удалить: "))

for i in range(k + 1, len(mass)):

    mass[i - 1] = mass[i]

mass.pop()

print(' '.join([str(i) for i in mass]))

**Контрольный пример:**

mass[1 2 3 4 5 6 7 8 9 10]

**Результат выполнения программы:**



# Задание №14:

Дан список целых чисел, число k и значение c. Необходимо вставить в список на позицию с индексом k элемент, равный c, сдвинув все элементы, имевшие индекс не менее k, вправо. Поскольку при этом количество элементов в списке увеличивается, после считывания списка в его конец нужно будет добавить новый элемент, используя метод append. Вставку необходимо осуществлять уже в считанном списке, не делая этого при выводе и не создавая дополнительного списка.

**Входные данные:**

mass[s], k,C[s] - string

**Выходные данные:**

mass[] - string

**Код:**

mass = [int(s) for s in input("Введите информацию: ").split()]

*#Индекс и число*

k, C = [int(s) for s in input("Введите индекс и число: ").split()]

mass.append(0)

for i in range(len(mass) - 1, k, -1):

    mass[i] = mass[i - 1]

mass[k] = C

print(' '.join([str(i) for i in mass]))

**Контрольный пример:**

mass[1 2 3 4 5 6 7]

**Результат выполнения программы:**



# Задание №15:

Дан список. Выведите те его элементы, которые встречаются в списке только один раз. Элементы нужно выводить в том порядке, в котором они встречаются в списке.

**Входные данные:**

mass[] - string

**Выходные данные:**

mass[i] - string

**Код:**

mass = [int(s) for s in input("Введите информацию: ").split()]

for i in range(len(mass)):

    for j in range(len(mass)):

        if i != j and mass[i] == mass[j]:

            break

    else:

        print(mass[i], end=' ')

**Контрольный пример:**

mass[1 2 1 3 4 4 5 6 7 6]

**Результат выполнения программы:**



# Задание №16:

Известно, что на доске 8×8 можно расставить 8 ферзей так, чтобы они не били друг друга. Вам дана расстановка 8 ферзей на доске, определите, есть ли среди них пара бьющих друг друга.

Программа получает на вход восемь пар чисел, каждое число от 1 до 8 — координаты 8 ферзей. Если ферзи не бьют друг друга, выведите слово NO, иначе выведите YES.

**Входные данные:**

mass[], k[] ,C[] – string

new\_x[], new\_y[] - string

**Математические операции:**

n = 8

**Код:**

n = 8

x = []

y = []

for i in range(n):

    new\_x, new\_y = [int(s) for s in input("Введите пару:").split()]

    x.append(new\_x)

    y.append(new\_y)

correct = True

for i in range(n):

    for j in range(i + 1, n):

        if x[i] == x[j] or y[i] == y[j] or abs(x[i] - x[j]) == abs(y[i] - y[j]):

            correct = False

if correct:

    print('NO')

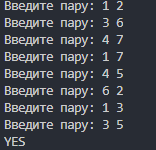
else:

    print('YES')

**Контрольный пример:**

new\_x, new\_y = 1 2, 3 6, 4 7, 1 7, 4 5, 6 2, 1 3, 3 5,

**Результат выполнения программы:**



# Задание №17:

N кеглей выставили в один ряд, занумеровав их слева направо числами от 1 до N. Затем по этому ряду бросили K шаров, при этом i-й шар сбил все кегли с номерами от l i до ri включительно. Определите, какие кегли остались стоять на месте. Программа получает на вход количество кеглей N и количество бросков K. Далее идет K пар чисел l i , ri , при этом 1≤ l i≤ ri≤ N. Программа должна вывести последовательность из N символов, где j-й символ есть “I”, если j-я кегля осталась стоять, или “.”, если j-я кегля была сбита.

**Входные данные:**

n[], k[]- string

left[], right[]- string

**Выходные данные:**

bahn[] - string

**Код:**

n, k = [int(s) for s in input().split()]

bahn = ['I'] \* n

for i in range(k):

    left, right = [int(s) for s in input().split()]

    for j in range(left - 1, right):

        bahn[j] = '.'

print(''.join(bahn))

**Контрольный пример:**

n, k = 8 3

left, right = 2 1; 1 2; 1 1

**Результат выполнения программы:**

