# Выведите все элементы, которые меньше 5

a = [1, 4, 2, 3, 5, 8, 13, 21, 34, 55, 89]

print([elem for elem in a if elem < 5])

#вернуть список, который состоит из элементов,

#общих для этих двух списков.

a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]

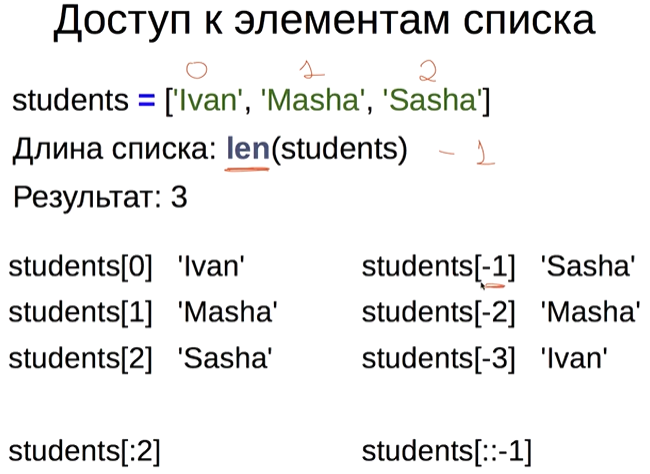
b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

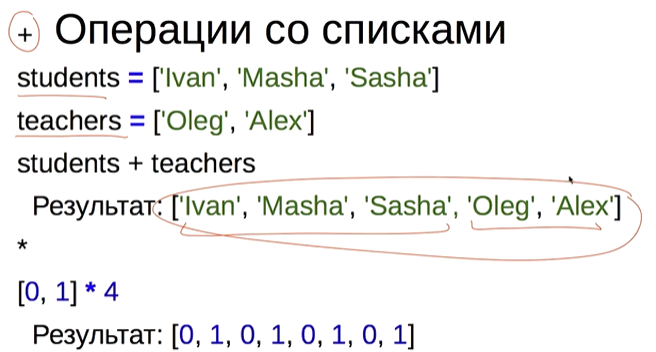
result = [elem for elem in a if elem in b]

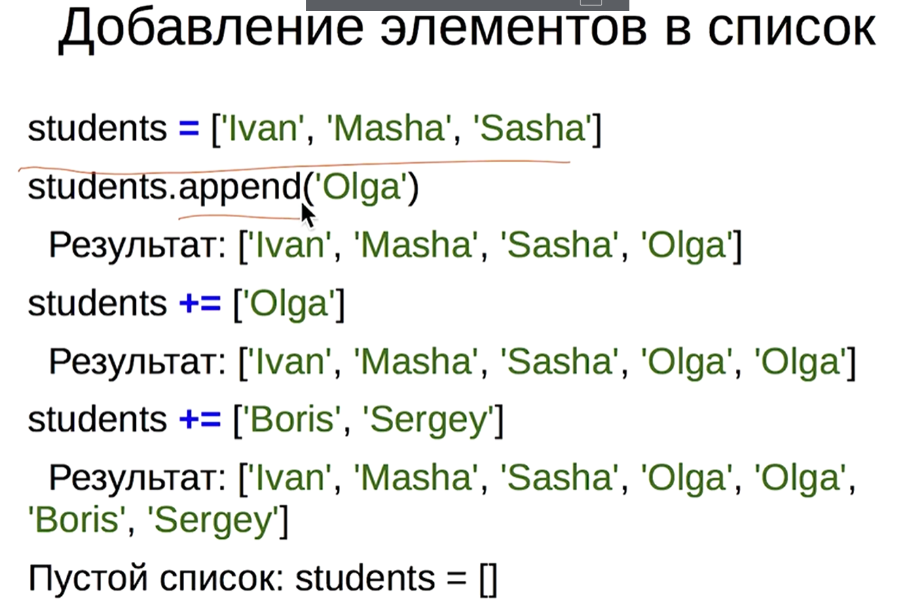
print(result)

**2.5 Списки**

|  |  |
| --- | --- |
| students = ['Ivan', 'Masha', 'Sasha']for student in students:print("Hello, " + student + "!") | Hello, Ivan!Hello, Masha!Hello, Sasha! |
| students = ['Ivan', 'Masha', 'Sasha']students[1] = 'Oleg'print(students) | ['Ivan', 'Oleg', 'Sasha'] |
| students = ['Ivan', 'Masha', 'Sasha']students.insert(1, 'Olga')print(students) | ['Ivan', 'Olga', 'Masha', 'Sasha'] |
| students = ['Ivan', 'Masha', 'Sasha']students += ['Olga']students += 'Olga'print(students) | ['Ivan', 'Masha', 'Sasha', 'Olga', 'O', 'l', 'g', 'a'] |
| students = ['Ivan', 'Masha', 'Sasha']students.remove('Sasha')print(students) | ['Ivan', 'Masha'] |
| students = ['Ivan', 'Masha', 'Sasha']students.remove('Sasha')print(students)del students[0]print(students) | ['Ivan', 'Masha']  ['Masha'] |
|  |  |







|  |  |
| --- | --- |
| students = ['Ivan', 'Masha', 'Sasha']if 'Ivan' in students:print('Ivan is here!')if 'Ann' not in students:print('Ann is out') | Ivan is here!  Ann is out |

Сортировка списка

|  |  |
| --- | --- |
| students = ['Ivan', 'Masha', 'Sasha']ordered\_students = sorted(students)print(students) | ['Ivan', 'Masha', 'Sasha'] |
| students = ['Ivan', 'Masha', 'Sasha']students.sort()print(students) | ['Ivan', 'Masha', 'Sasha'] |

Список в обратном порядке

|  |  |
| --- | --- |
| students = ['Ivan', 'Masha', 'Sasha']students.reverse()print(students) | ['Sasha', 'Masha', 'Ivan'] |
| students = ['Ivan', 'Masha', 'Sasha']reversed(students)print(students) | ['Ivan', 'Masha', 'Sasha'] |
| students = ['Ivan', 'Masha', 'Sasha']print(students[::-1]) | ['Sasha', 'Masha', 'Ivan'] |

Присвоение списков

|  |  |
| --- | --- |
| a = [1, 'A', 2]  b = a  a[0] = 42  print(a)  print(b)  b[2] = 30  print(a)  print(b) | [42, 'A', 2]  [42, 'A', 2]  [42, 'A', 30]  [42, 'A', 30]  > |
| a = [1, 2, 3]  b = a  print(b)  a[1] = 10  print(b)  b[0] = 20  print(a)  a = [5, 6]  print(b)  print(a) | [1, 2, 3]  [1, 10, 3]  [20, 10, 3]  [20, 10, 3]  [5, 6]  > Слушая только предыдущую лекцию, последний пример не решить. Можно было бы и объяснить, что если два объекта (переменные) ссылаются на одну ячейку и мы меняем элемент в ячейке, то ссылки сохраняются. А если мы одному из объектов присваиваем уже не элемент, список, то уже создается новая ячейка и связь от нее к объекту. Т.е. объекты уже ссылаются на разные ячейки. |

Генерация списков

|  |  |
| --- | --- |
| a = [0]\*5  print(a)  a = [0 for i in range(5)]  print(a)  a = [i \* i for i in range(5)]  print(a)  a = [int(i) for i in input().split()]  print(a) | [0, 0, 0, 0, 0]  [0, 0, 0, 0, 0]  [0, 1, 4, 9, 16] |

Напишите программу, на вход которой подается одна строка с целыми числами. Программа должна вывести сумму этих чисел.

Используйте метод split строки. ﻿﻿

**Sample Input:**

4 -1 9 3

**Sample Output:**

15

|  |  |
| --- | --- |
| print (sum([int(i) for i in input().split()])) | 4 -1 9 3  15  > |

Верное решение #28891552

Python 3

s = 0

for i in input().split():

s += int(i)

print(s)

Верное решение #27996477

Python 3

print(sum(map(int, input().split())))

Верное решение #117777349

Python 3

print(eval(input().replace(' ','+')))

Напишите программу, на вход которой подаётся список чисел одной строкой. Программа должна для каждого элемента этого списка вывести сумму двух его соседей. Для элементов списка, являющихся крайними, одним из соседей считается элемент, находящий на противоположном конце этого списка. Например, если на вход подаётся список "1 3 5 6 10", то на выход ожидается список "13 6 9 15 7" (без кавычек).

Если на вход пришло только одно число, надо вывести его же.

Вывод должен содержать одну строку с числами нового списка, разделёнными пробелом.

**Sample Input 1:**

1 3 5 6 10

**Sample Output 1:**

13 6 9 15 7

**Sample Input 2:**

10

**Sample Output 2:**

10

|  |  |
| --- | --- |
| a = [int(i) for i in input().split()]  if len(a) == 1:  print (a[0])  else:  for i in range(len(a)):  if i == 0:  print ((a[1] + a[-1]), end=' ')  elif i == len(a) - 1:  print ((a[i - 1] + a[0]))  else:  print ((a[i - 1] + a[i + 1]), end=' ') | **Sample Input 1:**  1 3 5 6 10  **Sample Output 1:**  13 6 9 15 7  **Sample Input 2:**  10  **Sample Output 2:**  10 |

Верное решение #29660537

Python 3

# put your python code here

numbers = [int(i) for i in input().split()]

if len(numbers) == 1:

print(numbers[0])

else:

for i in range(len(numbers)):

print(numbers[i - 1] + numbers[(i + 1) % len(numbers)], end=" ")

Верное решение #445306655

Python 3

arr = list(map(int, input().split()))

print(\*arr if len(arr) == 1 else [arr[i - 1] + arr[(i + 1) % len(arr)] for i in range(len(arr))])

Верное решение #22659477

Python 3

# put your python code here

a=[int(i) for i in input().split()]

if len(a)>1:

for i in range(len(a)):

print(a[i-1]+a[i+1-len(a)])

else:

print(a[0])

Верное решение #36524992

Python 3

# put your python code here

a = [int(i) for i in input().split()]

b = len(a)

[print(a[i-1]\*(1-(b==1))+a[i-(b-1)], end = ' ') for i in range(0, b)]

Напишите программу, которая принимает на вход список чисел в одной строке и выводит на экран в одну строку значения, которые встречаются в нём более одного раза.

Для решения задачи может пригодиться метод sort списка.

Выводимые числа не должны повторяться, порядок их вывода может быть произвольным.

|  |  |
| --- | --- |
| **a = sorted([int(i) for i in input().split()])**  **for i in range(len(a) - 1):**  **if i != 0:**  **if a[i] == a[i-1]:**  **continue**  **if a[i] == a[i+1]:**  **print (a[i], end=' ')** | **Sample Input 1:**  4 8 0 3 4 2 0 3  **Sample Output 1:**  0 3 4  **Sample Input 2:**  10  **Sample Output 2:**  **Sample Input 3:**  1 1 2 2 3 3  **Sample Output 3:**  1 2 3  **Sample Input 4:**  1 1 1 1 1 2 2 2  **Sample Output 4:**  1 2 |

Верное решение #28911627

Python 3

a, c = [str(i) for i in input().split()], []

for i in a:

if i not in c and a.count(i) > 1:

c.append(i)

print(i, end=' ')

Верное решение #21321232

Python 3

ls = [int(i) for i in input().split()]

for i in set(ls):

if ls.count(i) > 1:

print(i, end=' ')

Верное решение #30576590

Python 3

str = [int(i) for i in input().split()]

ans = []

[ans.append(x) for x in str if x not in ans and str.count(x) > 1]

print(\*ans)

Верное решение #25753831

Python 3

s = input().split()

print (\*(i for i in set(s) if s.count(i) > 1))

Верное решение #40160010

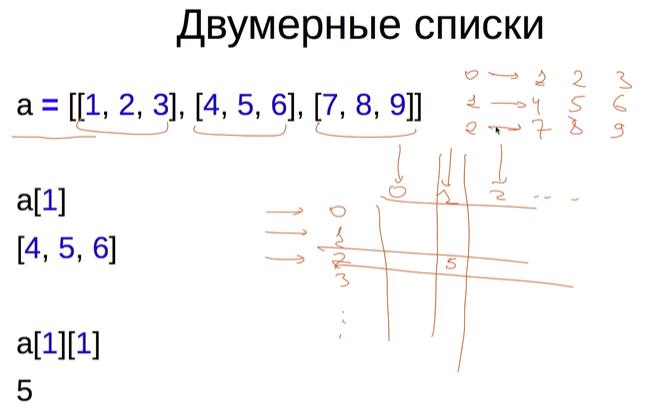
Python 3

a=input().split()

[a.remove(i) for i in set(a)]

print(\*set(a))

**Двумерные списки**

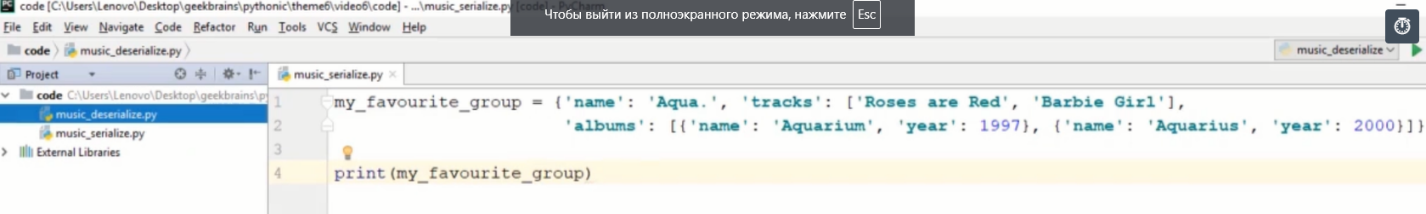
****

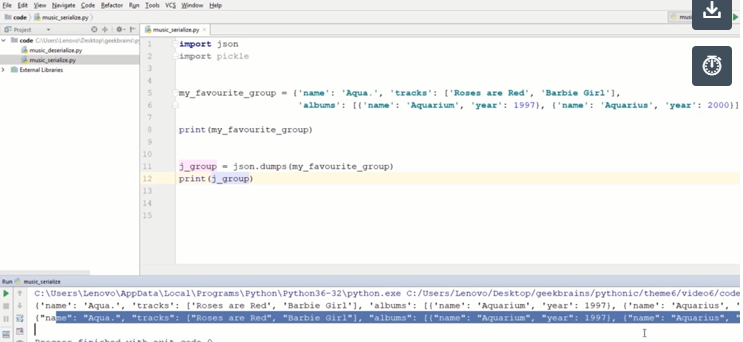
**Практическое задание**

**1: Создать модуль music\_serialize.py. В этом модуле определить словарь для вашей любимой музыкальной группы, например:**

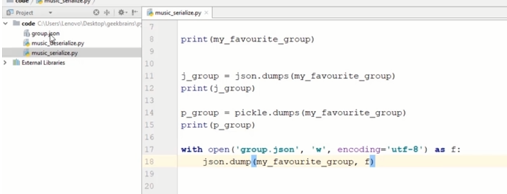
my\_favourite\_group = {  
‘name’: ‘Г.М.О.’,  
‘tracks’: [‘Последний месяц осени’, ‘Шапито’],  
‘Albums’: [{‘name’: ‘Делать панк-рок’,‘year’: 2016},  
{‘name’: ‘Шапито’,‘year’: 2014}]}

**С помощью модулей json и pickle сериализовать данный словарь в json и в байты, вывести результаты в терминал. Записать результаты в файлы group.json, group.pickle соответственно. В файле group.json указать кодировку utf-8.**

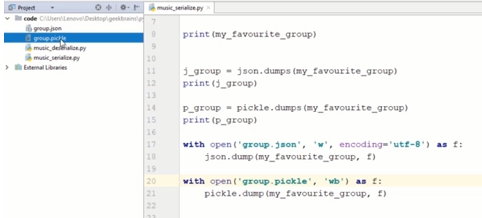


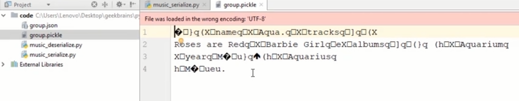












import pickle  
import json

my\_favourite\_group = {  
'name': 'Г.М.О.',  
'tracks': ['Последний месяц осени', 'Шапито'],  
'Albums': [{'name': 'Делать панк-рок', 'year': 2016},  
{'name': 'Шапито', 'year': 2014}]  
}

my\_favourite\_group\_pickle = pickle.dumps(my\_favourite\_group)  
my\_favourite\_group\_json = json.dumps(my\_favourite\_group)

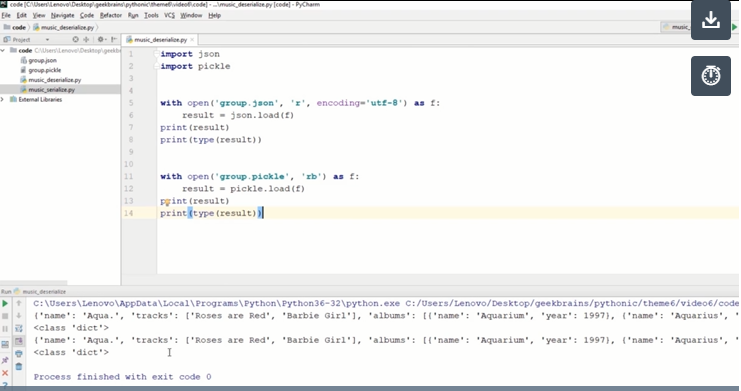
print(my\_favourite\_group\_pickle)  
print(my\_favourite\_group\_json)

with open('group.pickle', 'wb') as f:  
pickle.dump(my\_favourite\_group\_pickle, f)

with open('group.json', 'w', encoding='utf-8') as f:  
json.dump(my\_favourite\_group\_json, f, ensure\_ascii=False)

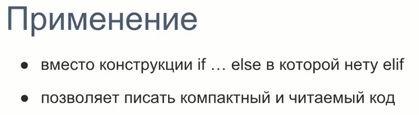
print("Записано")

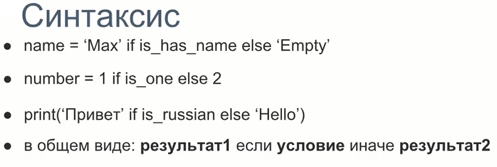
**2: Создать модуль music\_deserialize.py. В этом модуле открыть файлы group.json и group.pickle, прочитать из них информацию. И получить объект: словарь из предыдущего задания.**

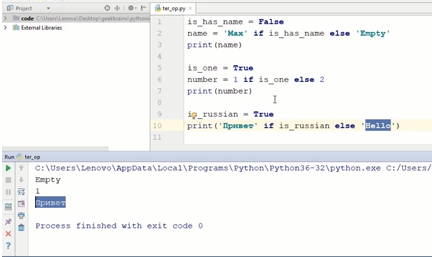


**урок 14 - Полезные инструменты. Обработка исключений**

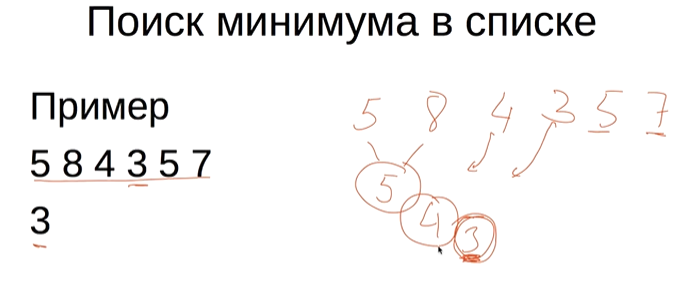












|  |  |
| --- | --- |
| s = [5, 8, 4, 3, 5, 7, 2]  min = s[0]  for i in range(s[-2]):  if min > s[i]:  min = s[i]  i+=1  print(min) | 2  > |
| Поиск минимума в списке[¶](file:///C:\Users\baitokayeva.e\Downloads\week_2.html#Поиск-минимума-в-списке) a = [int(i) **for** i **in** input().split()]  m = a[0]  **for** x **in** a:  **if** m > x:  m = x  **print**(m) | 4 7 3 8  3 |
| Поле для игры Сапер n, m, k = (int(i) for i in input().split()) # чтение размеров поля и числа мин  a = [[0 for j in range(m)] for i in range(n)] # заполнение поля нулями  for i in range(k):  row, col = (int(i) - 1 for i in input().split())  a[row][col] = -1 # расставляем мины  for i in range(n):    # перебираем строки  for j in range(m):   # перебираем столбцы  if a[i][j] == 0:   # ячейка без мины  for di in range(-1, 2):   # перебираем соседние строки (просто цифры -1 0 1)  for dj in range(-1, 2):   # перебираем соседние столбцы (просто цифры -1 0 1)  ai = i + di    # координата по строке  aj = j + dj    # координата по столбцу  if 0 <= ai < n and 0 <= aj < m and a[ai][aj] == -1:   # проверка вхождения в диапазон и мины по соседству  a[i][j] += 1  # вывод результа  та# дальше просто заменяем -1 на "\*" и 0 на "."  for i in range(n):  for j in range(m):  if a[i][j] == -1:  print('\*', end='')  elif a[i][j] == 0:  print('.', end='')  else:  print(a[i][j], end='')  print() | вход  5 4 4  1 1  2 2  3 2  4 4  вывод  \*21.  3\*2.  2\*31  112\*  ..11 |
| Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и **сразу после этого**выводит сумму квадратов всех считанных чисел.  Гарантируется, что в какой-то момент сумма введённых чисел окажется равной 0, **после этого считывание продолжать не нужно**.  В примере мы считываем числа 1, -3, 5, -6, -10, 13; в этот момент замечаем, что сумма этих чисел равна нулю и выводим сумму их квадратов, не обращая внимания на то, что остались ещё не прочитанные значения. | |
| # put your python code here  b = []  while True:  b += [int(input())]  if sum(b)==0:  break  print(sum(i\*i for i in b)) | **Sample Input:**  1  -3  5  -6  -10  13  4  -8  **Sample Output:**  340 |

<https://stepic.org/media/attachments/course67/week_2.html>  ﻿

Верное решение #20999160

Python 3

s=[int(input())]

while sum(s)!=0: s.append(int(input()))

print(sum([i\*\*2 for i in s]))

Верное решение #45327481

Python 3

s = [0, 0, 1]

while s[2]:

i = int(input())

s = [s[0] + i, s[1] + i \*\* 2, s[0] + i]

print(s[1])

Верное решение #16161654

Python 3

# put your python code here

arr = [int(input())]

while sum(arr) != 0:

arr.append(int(input()))

print(sum([i\*\*2 for i in arr]))

Верное решение #226116637

Python 3

a = int(input())

c = a \*\* 2

while a != 0:

b = int(input())

a += b

c += b \*\* 2

print(c)

|  |  |
| --- | --- |
| Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 ... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число n — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.  Например, если n = 7, то программа должна вывести 1 2 2 3 3 3 4. | |
| list, n = [], int(input())  for i in range(n):  count = 0  if n ==1:  print(n)  break  while count < i+1:  list.append(i+1)  count += 1  if len(list) == n:  print(\*list)  break | **Sample Input:**  7  **Sample Output:**  1 2 2 3 3 3 4 |

Верное решение #17909480

Python 3

n = int(input())

a = []

i = 0

while len(a) < n:

a += [i] \* i

i += 1

print(\*a[:n])

Загуглил последовательность, у нее оказалась готовая форма для любого элемента и соответственно печатаем его:

a(n)=floor(1/2+sqrt(2n))*a*(*n*)=*floor*(1/2+*sqrt*(2*n*))

Верное решение #46870284

Python 3

import math

x = int(input())

print(\*[int( 1/2 + math.sqrt(2 \* n) ) for n in range(1, x + 1)])

Верное решение #37665928

Python 3

a = int(input())

c = 0

for i in range(a+1):

for j in range(i):

c += 1

if c<a+1:

print(i, end=' ')

Верное решение #93679329

Python 3

n=int(input())

print(\*[i for i in range(1,n+1) for j in range(i)][:n])

|  |  |
| --- | --- |
| Напишите программу, которая считывает список чисел lst*lst* из первой строки и число x*x* из второй строки, которая выводит все позиции, на которых встречается число x*x* в переданном списке lst*lst*.  Позиции нумеруются с нуля, если число x*x* не встречается в списке, вывести строку "Отсутствует" (без кавычек, с большой буквы).  Позиции должны быть выведены в одну строку, по возрастанию абсолютного значения. | # put your python code here  lst, x, log = [int(i) for i in input().split()], int(input()), False  for i in range(len(lst)):  if lst[i] == x:  print(i, end=' ')  log = True  if log == False:  print('Отсутствует')  **Sample Input 1:**  5 8 2 7 8 8 2 4  8  **Sample Output 1:**  1 4 5 |
|  |  |

Верное решение #96582777

Python 3

# put your python code here

l, n = [int(i) for i in input().split()], int(input())

print(\*[x for x in range(len(l)) if l[x]==n] if n in l else ["Отсутствует"])

Верное решение #20851913

Python 3

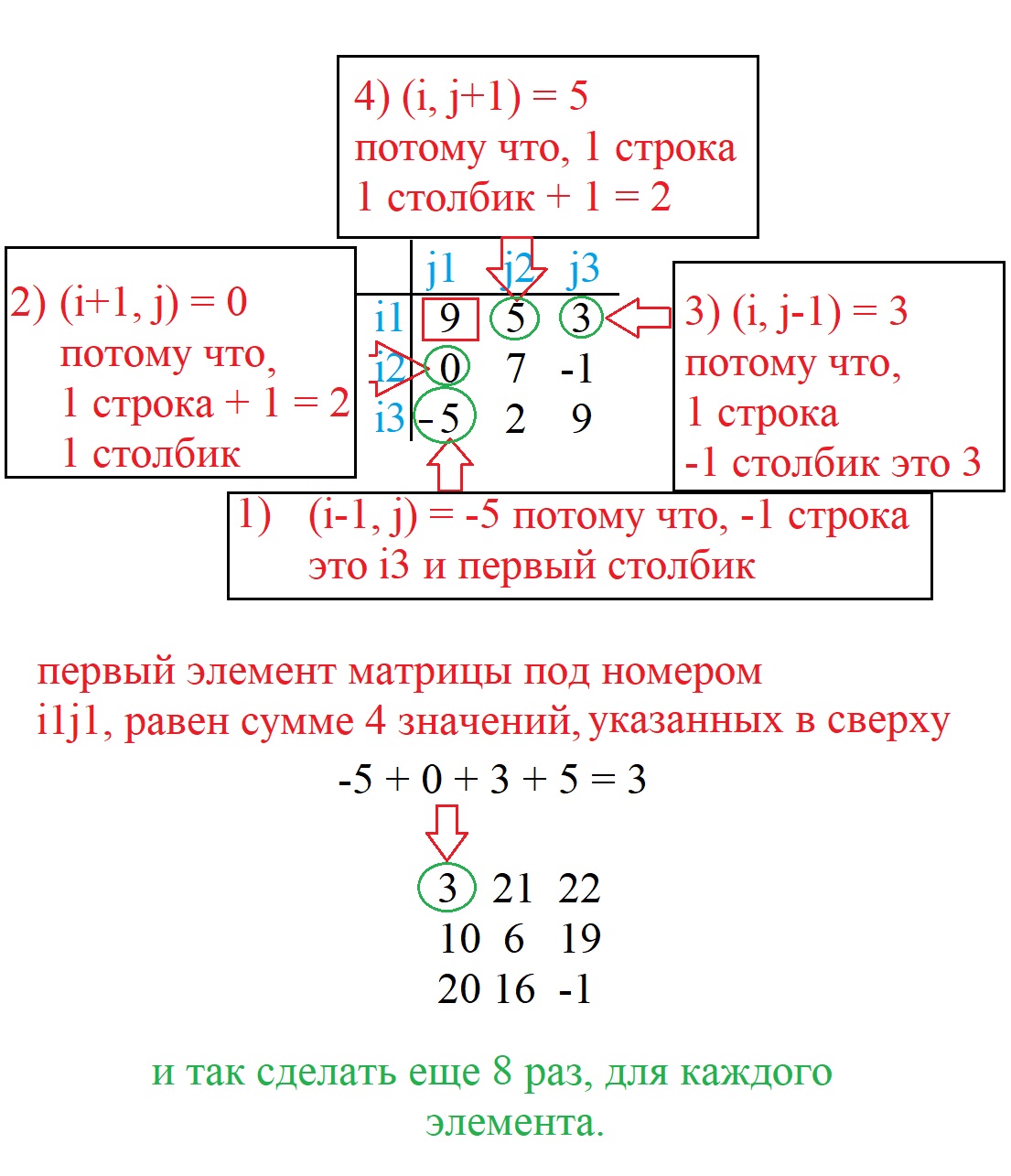
l = [int(i) for i in input().split()]

x = int(input())

if not x in l: print('Отсутствует')

for i in range(len(l)):

if l[i]==x: print(i, end = ' ')



|  |  |
| --- | --- |
| Напишите программу, на вход которой подаётся прямоугольная матрица в виде последовательности строк. После последней строки матрицы идёт строка, содержащая только строку "end" (без кавычек, см. Sample Input).  Программа должна вывести матрицу того же размера, у которой каждый элемент в позиции i, j равен сумме элементов первой матрицы на позициях (i-1, j), (i+1, j), (i, j-1), (i, j+1). У крайних символов соседний элемент находится с противоположной стороны матрицы.  В случае одной строки/столбца элемент сам себе является соседом по соответствующему направлению. | m = []  while True:  n = str(input()) # ввод строк  if n == 'end':  break  m.append([int(s) for s in n.split()])  li, lj = len(m), len(m[0])  new = [[sum([m[i-1][j], m[(i+1)%li][j], m[i][j-1], m[i][(j+1)%lj]]) for j in range(lj)] for i in range(li)]  for i in range (li):  for j in range (lj):  print(new[i][j], end =' ')  print()  **Sample Input 1:**  9 5 3  0 7 -1  -5 2 9  end  **Sample Output 1:**  3 21 22  10 6 19  20 16 -1 |

a =[list(map(int, b.split())) for b in iter(input, 'end')]

for row in [[a[i-1][j]+a[(i+1) % len(a)][j]+a[i][j-1]+a[i][(j+1) % len(a[i])] for j in range(len(a[i]))] for i in range(len(a))]:

print(\*row)

Верное решение #99988911

Python 3

a = []

b = input()

while b != "end":

a.append(b.split())

b = input()

for i in range(len(a)):

for j in range(len(a[i])):

print(int(a[i-1][j]) + int(a[i-len(a)+1][j]) + int(a[i][j-1]) + int(a[i][j-len(a[i])+1]), end =" ")

print()

Верное решение #112764503

Python 3

a = []

b = []

n = input()

while n != 'end':

a.append([int(i) for i in n.split()])

b.append([int(i) for i in n.split()])

n = input()

for i in range(len(a)):

for j in range(len(a[i])):

b[i][j] = a[i-1][j] + a[i+1-len(a)][j] + a[i][j-1] + a[i][j+1-len(a[i])]

print(\*b[i])

|  |  |
| --- | --- |
| Выведите таблицу размером n \times n*n*×*n*, заполненную числами от 11 до n^2*n*2 по спирали, выходящей из левого верхнего угла и закрученной по часовой стрелке, как показано в примере (здесь n=5*n*=5):  **Sample Input:**  5  **Sample Output:**  1 2 3 4 5  16 17 18 19 6  15 24 25 20 7  14 23 22 21 8  13 12 11 10 9 | n = int(input())  a = [[0] \* n for i in range(n)] #создаем матрицу согласно заданного размера  count = 0 # количество заполненных ячеек  for i in range(n):#заполнение 1 строки  count += 1  a[0][i] = count  j = 0 # указываем последнюю заполненную ячейку  i = n-1  n -= 1 # устанавливаем размер 1 блока 1 витка  while len(a)\*\*2 != count: #условие выхода из цикла  for k in range(n): #движение вниз  j += 1  count += 1  a[j][i] = count # заполнение матрицы  for k in range(n): #движение влево  i -= 1  count += 1  a[j][i] = count  for k in range(n-1): #движение вверх  j -= 1  count += 1  a[j][i] = count  for k in range(n-1): #движение вправо  i += 1  count += 1  a[j][i] = count  n -= 2 # обеспечиваем переход на внутренний виток  for i in range(len(a)): #вывод полученной матрицы  for j in range(len(a[i])):  print(a[i][j], end=' ')  print() |

Верное решение #26082360

Python 3

n=int(input())

t=[[0]\*n for i in range (n)]

i,j=0,0

for k in range(1, n\*n+1):

t[i][j]=k

if k==n\*n: break

if i<=j+1 and i+j<n-1: j+=1

elif i<j and i+j>=n-1: i+=1

elif i>=j and i+j>n-1: j-=1

elif i>j+1 and i+j<=n-1: i-=1

for i in range(n):

print(\*t[i])

Верное решение #37215733

Python 3

n = int(input())

m = [[0] \* n for i in range(n)]

i, j, di, dj = 0, 0, 0, 1

for k in range(n \* n):

m[i][j] = k + 1

if (not -1 < i + di < n) or (not -1 < j + dj < n) or m[i + di][j + dj] != 0:

di, dj = dj, -di

i, j = i + di, j + dj

[print(\*i) for i in m]

Верное решение #30603107

Python 3

n = int(input())

m = [[0 for i in range(n)] for j in range(n)]

for k in range(n, 0, -1):

for j in range(n-k, k):

m[n-k][j] = m[n-k][j-1] + 1

for i in range(n-k, k-1):

m[i+1][k-1] = m[i][k-1] + 1

for j in range(k-2, n-k-1, -1):

m[k-1][j] = m[k-1][j+1] + 1

for i in range(k-2, n-k, -1):

m[i][n-k] = m[i+1][n-k] + 1

for i in m:

print(\*i)

Верное решение #40577480

Python 3

n = int(input())

num = [[0] \* n for y in range(n)]

y, x, dy, dx = 0, 0, 0, 1

for k in range(n\*n):

num[y][x] = k+1

if (y+dy not in range(-1,n)or #не столбец

x+dx not in range(-1,n)or #не строка

num[y + dy][x + dx] > 0): #следующая занята

dx, dy = -dy, dx #то - переприсваивание

y, x = y + dy, x + dx

[print(\*i) for i in num]

3.1 Функции

Здравствуйте!

Третий и последний модуль нашего курса посвящена таким важным понятиям в программировании, как функции, модули и файловый ввод/вывод.

Также мы рассмотрим тип данных "словарь" и некоторые частные примеры наиболее популярных и полезных модулей Python.

Как всегда, мы выложили весь код, рассматриваемый на этой неделе:  
<https://stepic.org/media/attachments/course67/week_3.html> ﻿

[Текстовый конспект лекций недели](https://wiki.stepik.org/index.php/Category:%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D0%B8%D0%B8._%D0%A1%D0%BB%D0%BE%D0%B2%D0%B0%D1%80%D0%B8._%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80._%D0%A4%D0%B0%D0%B9%D0%BB%D1%8B._%D0%9C%D0%BE%D0%B4%D1%83%D0%BB%D0%B8._(M-149))

def min2(a, b)

if a<=b:

return a

else

return b

вызов функции

m= min2(42, 30) найти минимум из числе 42, 30

m = min2(min2(42, 30), 25) найти минимум из числе 42, 30, 25

функция должны быть объявлена ранее первого вызова

пример

|  |  |
| --- | --- |
| def f(n):  return n \* 10 + 5 | f(f(f(10)))  10555 |

### Функции[¶](file:///C:\Users\baitokayeva.e\Downloads\week_3.html#Функции)

In [1]:

**def** min2(a, b):

**if** a <= b:

**return** a

**else**:

**return** b

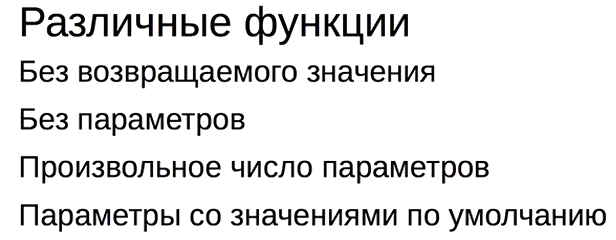
In [2]:

m = min2(42, 30)

**print**(min2(min2(42, 30), 25))

25

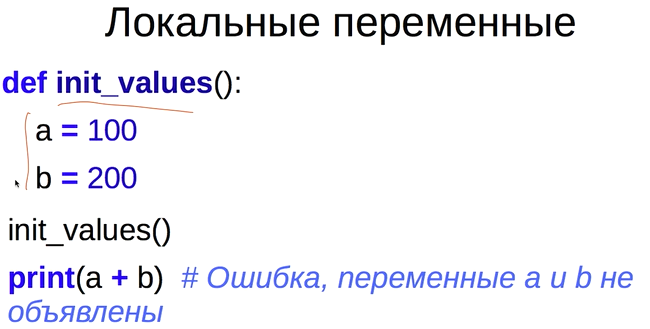
In [3]:



|  |  |
| --- | --- |
| **def** min(\*a):  m = a[0]  **for** x **in** a:  **if** m > x:  m = x  **return** m  **print**(min(5))  **print**(min(5, 3))  **print**(min(5, 3, 6, 10))  **print**(min([5, 3, 6, 10])) | 5  3  3  [5, 3, 6, 10] |

**Значения параметров по умолчанию**

|  |  |
| --- | --- |
| **def** my\_range(start, stop, step=1):  res = []  **if** step > 0:  x = start  **while** x < stop:  res += [x]  x += step  **elif** step < 0:  x = start  **while** x > stop:  res += [x]  x += step  **return** res  **print**(my\_range(2, 5))  **print**(my\_range(2, 15, 3))  **print**(my\_range(15, 2, -3)) | [2, 3, 4]  [2, 5, 8, 11, 14]  [15, 12, 9, 6, 3] |

****

**def** init\_values():

a = 100

b = 200

init\_values()

**print**(a + b) *# Ошибка, переменные a и b не объявлены*

NameError Traceback (most recent call last)

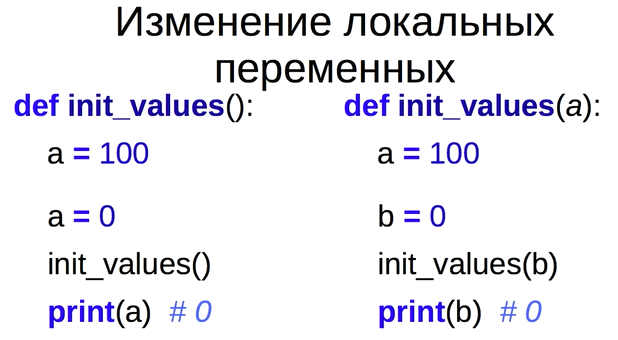
<ipython-input-10-b3d11a658b04> in <module>()

4

5 init\_values()

----> 6 print(a + b) # Ошибка, переменные a и b не объявлены

NameError: name 'a' is not defined

****

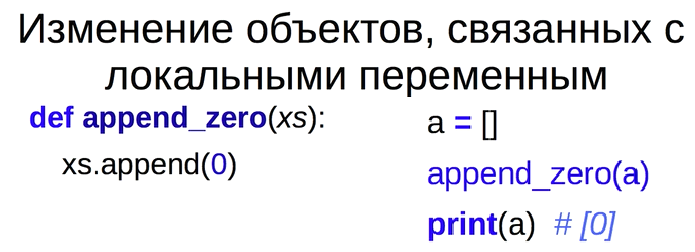
**def** init\_values():

a = 100

a = 0

init\_values()

**print**(a) *# 0*

****

**def** print\_value():

**print**(a)

a = 5

print\_value() *# 5*

**def** print\_value():

**print**(a)

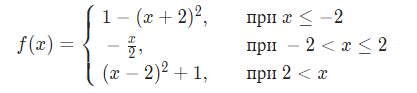
a = 10

**print**(a)

a = 5

print\_value() *# UnboundLocalError: local variable 'a' referenced before assignment*

Напишите функцию f(x), которая возвращает значение следующей функции, определённой на всей числовой прямой:



Требуется реализовать только функцию, решение не должно осуществлять операций ввода-вывода.

|  |  |
| --- | --- |
| def f(x):  if x <= -2:  x = 1 - (x + 2)\*\*2  elif -2 < x <= 2:  x = - x / 2  else:  x = (x - 2)\*\*2 + 1  return(x) | **Sample Input 1:**  4.5  **Sample Output 1:**  7.25 |

Верное решение #91334436

Python 3

def f(x):

return {

x <= -2: 1 - (x + 2) \*\* 2,

-2 < x <= 2: -x / 2,

2 < x: (x - 2) \*\* 2 + 1

}[True]

Верное решение #104906481

Python 3

def f(x):

return float(x<-2 and 1-(x+2)\*\*2 or 2<x and (x-2)\*\*2+1 or -2<=x<=2 and -x/2)

Верное решение #446322960

Python 3

def f(x): return (lambda x :1 - (x + 2) \*\* 2 if x <= -2 else -x / 2 if x < -2 or x <= 2 else (x - 2) \*\* 2 + 1)(x)

Напишите функцию modify\_list(l), которая принимает на вход список целых чисел, удаляет из него все нечётные значения, а чётные **нацело** делит на два. Функция не должна ничего возвращать, требуется только изменение переданного списка, например:

lst = [1, 2, 3, 4, 5, 6]

print(modify\_list(lst)) # None

print(lst)   # [1, 2, 3]

modify\_list(lst) print(lst) # [1]

lst = [10, 5, 8, 3]

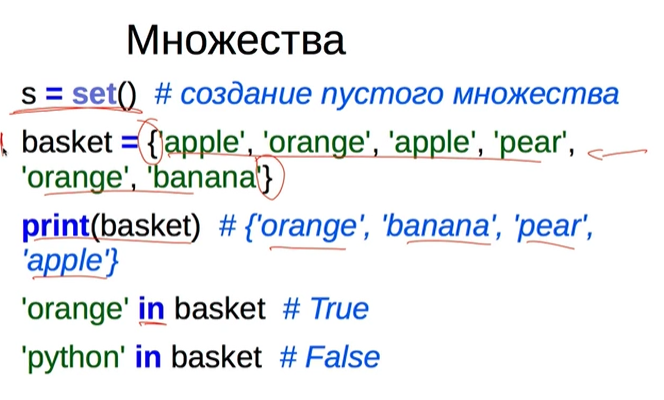
modify\_list(lst)

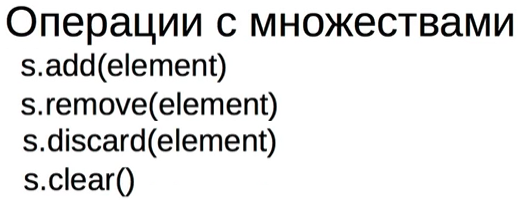
print(lst) # [5, 4]

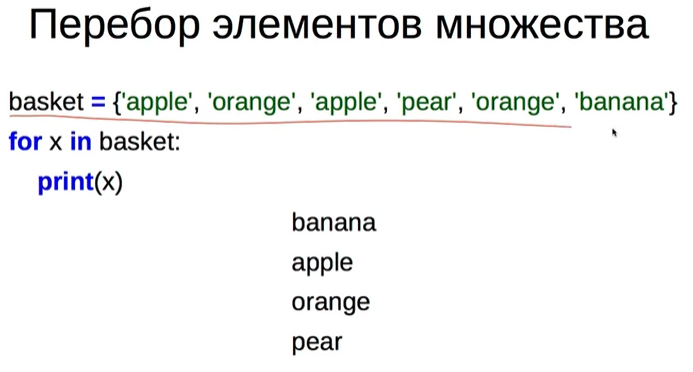
Функция не должна осуществлять ввод/вывод информации.

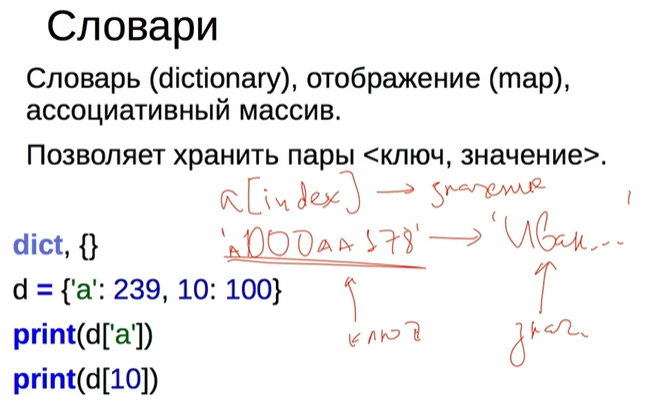
|  |  |
| --- | --- |
| **def modify\_list(l):**  **i = len(l) - 1**  **while i != -1:**  **if l[i] % 2:**  **del l[i]**  **else:**  **l[i] = l[i] // 2**  **i -= 1**  **return** | Верное решение #22076678  Python 3  def modify\_list(l):  l[:] = [i//2 for i in l if not i % 2]  Верное решение #30590214  Python 3  def modify\_list(l):  l[:] = [i//2 for i in l if i % 2 == 0]  Верное решение #28435016  Python 3  def modify\_list(l):  for x in l[:]:  if x % 2 == 0:  l.append(x//2)  l.remove(x) |

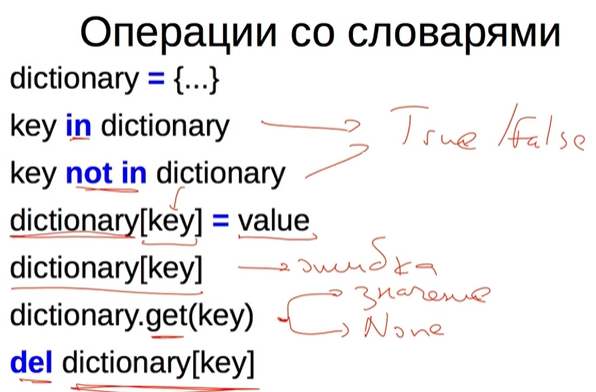
3.2 Словари

****

****

****

****

****

****

Напишите функцию update\_dictionary(d, key, value), которая принимает на вход словарь d*d* и два числа: key*key* и value*value*.

Если ключ key*key* есть в словаре d*d*, то добавьте значение value*value* в список, который хранится по этому ключу.  
Если ключа key*key* нет в словаре, то нужно добавить значение в список по ключу 2 \* key2∗*key*. Если и ключа 2 \* key2∗*key* нет, то нужно добавить ключ 2 \* key2∗*key* в словарь и сопоставить ему список из переданного элемента [value][*value*].

Требуется реализовать только эту функцию, кода вне её не должно быть.  
Функция не должна вызывать внутри себя функции input и print.

Пример работы функции:

|  |  |
| --- | --- |
| Верное решение #26282700  Python 3  def update\_dictionary(d, key, value):  if key in d:  d[key] += [value]  elif 2 \* key in d:  d[2 \* key] += [value]  else:  d[2 \* key] = [value]  d = {}  print(update\_dictionary(d, 1, -1))  print(d)  update\_dictionary(d, 2, -2) print(d)  update\_dictionary(d, 1, -3) print(d) | # None  # {2: [-1]}  # {2: [-1, -2]}  # {2: [-1, -2, -3]} |

Верное решение #99630942

Python 3

def update\_dictionary(d, key, value):

if key not in d:

key = 2 \* key

if key not in d:

d[key] = list()

d[key].append(value)

Верное решение #17081298

Python 3

def update\_dictionary(d, k, v):

if k in d:

d.setdefault(k, []).append(v)

else:

d.setdefault(k\*2, []).append(v)

Верное решение #37372034

Python 3

def update\_dictionary(d, key, value):

key += key \* (key not in d)

d[key] = d.get(key, []) + [value]

Верное решение #25921434

Python 3

def update\_dictionary(d, key, value):

if key in d.keys():

d[key].append(value)

elif 2\*key in d.keys():

d[2\*key].append(value)

else:

d[2\*key]=[value]

Верное решение #36089425

Python 3

# не добавляйте кода вне функции

def update\_dictionary(d, key, value):

key <<= key not in d

d.setdefault(key,[]).append(value)

Верное решение #45032624

Python 3

def update\_dictionary(d, key, value): d.setdefault(key+key\*(key not in d),[]).append(value)

Верное решение #86175742

Python 3

# не добавляйте кода вне функции

def update\_dictionary(d, key, value):

if key in d:

d[key] += [value]

else:

if key\*2 in d:

d[key\*2] += [value]

else:

d[key\*2] = [value]

Когда Антон прочитал «Войну и мир», ему стало интересно, сколько слов и в каком количестве используется в этой книге.

Помогите Антону написать упрощённую версию такой программы, которая сможет подсчитать слова, разделённые пробелом и вывести получившуюся статистику.

Программа должна считывать одну строку со стандартного ввода и выводить для каждого уникального слова в этой строке число его повторений (**без учёта регистра**) в формате "слово количество" (см. пример вывода).  
Порядок вывода слов может быть произвольным, каждое уникальное слово﻿ должно выводиться только один раз.

|  |  |
| --- | --- |
| from collections import Counter  text = Counter(input().lower().split())  for key, value in text.items():  print(key, value, end='\n') | **Sample Input 1:**  a aa abC aa ac abc bcd a  **Sample Output 1:**  ac 1  a 2  abc 2  bcd 1  aa 2  **Sample Input 2:**  a A a  **Sample Output 2:**  a 3 |

Верное решение #18976059

Python 3

# put your python code here

s = input().lower().split()

for i in set(s):

print(i, s.count(i))

Верное решение #136031793

Python 3

# put your python code here

text = input().lower().split()

d = { i : text.count(i) for i in text }

for key, value in d.items():

print(key, value, end='\n')

Напишите программу, которая считывает строку с числом n*n*, которое задаёт количество чисел, которые нужно считать. Далее считывает n*n* строк с числами x\_i*xi*​, по одному числу в каждой строке. Итого будет n+1*n*+1 строк.

При считывании числа x\_i*xi*​ программа должна на отдельной строке вывести значение f(x\_i)*f*(*xi*​). Функция f(x) уже реализована и доступна для вызова.

Функция вычисляется достаточно долго и зависит только от переданного аргумента x*x*. Для того, чтобы уложиться в ограничение по времени, нужно избежать повторного вычисления значений.

|  |  |
| --- | --- |
| **d = {}**  **k = []**  **n = int(input())**  **for i in range(n):**  **x = int(input())**  **k.append(x)**  **for j in range(0,len(k)):**  **key = k[j]**  **if key in d:**  **print(d[key])**  **elif key not in d:**  **p = k[j]**  **d[key] = f(p)**  **print(d.get(key))** | **Sample Input:**  5  5  12  9  20  12  **Sample Output:**  11  41  47  61  41 |

Верное решение #21028314

Python 3

# Считайте, что функция f(x) уже определена выше. Определять её отдельно не требуется.

a=[int(input()) for i in range(int(input()))]

b={x:f(x) for x in set(a)}

for i in a:

print(b[i])

Верное решение #26303970

Python 3

d = {}

for \_ in range(int(input())):

x = int(input())

if x not in d:

d[x] = f(x)

print(d[x])

Верное решение #29123682

Python 3

# Считайте, что функция f(x) уже определена выше. Определять её отдельно не требуется.

n=int(input())

d={}

for i in range(n):

x=int(input())

if x not in d:

d[x]=f(x)

print(d[x])

Верное решение #32406138

Python 3

x = [int(input()) for i in range(int(input()))]

b = {x : f(x) for x in set(x)}

print (\*[b[i] for i in x], sep='\n')