**#task1**------------------------------------------------------------------

"По данному целому числу N распечатайте все квадраты натуральных чисел, не превосходящие N, в порядке возрастания."

#defaults

defaults\_number = 1 #int

#input

print ('Task 1: Список квадратов натуральных чисел, не превосходящие N.')

number = int(input('Ведите натуральное число N: ')) #int; натуральное число

#main

if number < 0:

print('Введено не допустимое значение')

else:

print('По данному целому числу', number, 'все квадраты натуральных чисел равны:')

while (defaults\_number \*\* 2) <= number:

#output

print (defaults\_number \*\* 2, end=' ') #int

defaults\_number +=1

print()

print()

#----------------------------------------------------------------------

**#task2**------------------------------------------------------------------

"Дано целое число, не меньшее 2. Выведите его наименьший натуральный делитель, отличный от 1."

print('Task 2: Вывод наименьшего отличного от 1 делителя числа n,')

#defaults

defaults\_number = 2 #int

#input

number = int(input('Введите число n больше 2')) #int; целое число больше 2

#main

if number < 2:

print('Введено недопустимое значение')

else:

while number % defaults\_number != 0:

defaults\_number += 1

#output

print('Минимальный делитель числа',number,'равен',defaults\_number) #int;

print()

#------------------------------------------------------------------------

**#task3**-------------------------------------------------------------------

"По данному натуральному числу N найдите наибольшую целую степень двойки, не превосходящую N. Выведите показатель степени и саму степень."

#defaults

two\_in\_power = 2 #int

power = 1 #int

#input

print('Task 3: Вычесление наибольшей целой степени двойки, не превосходящей N')

number = int(input('Введите натуральное число N больше 1: ')) #int; целое число больше 1

#main

if number < 1:

print('Введено недопустимое значение!')

else:

while two\_in\_power <= number:

two\_in\_power \*= 2

power += 1

#output

print('Наибольшая степень двойки, не привосходящая', number, ', равна', power - 1,', а сама сама степень - ', two\_in\_power // 2)

print()

#------------------------------------------------------------------------------------

**#task4**--------------------------------------------------------------------

"В первый день спортсмен пробежал x километров, а затем он каждый день увеличивал пробег на 10% от предыдущего значения. По данному числу y определите номер дня, на который пробег спортсмена составит не менее y километров."

#defaults

day = 1 #int

#input

print ('Task 4: Утренняя пробежка')

first\_day\_running = float(input('Сколько километров спортсмен пробежал в первый день?')) #действительное число в километрах больше нуля; формат: 0.0.

last\_running = float(input('Сколько километров спортсмен должен пробежать в последний день?')) #действительное число в километрах больше нуля; формат: 0.0.

#main

if first\_day\_running < 0 or last\_running < 0:

print('Введено недопустимое значение!')

else:

while first\_day\_running < last\_running :

first\_day\_running \*= 1.1

day += 1

#output

print('Если спортсмен каждый день будет увеличивать пробег на 10%, то на', day,'день он пробежит не менее', last\_running) #int

print()

#---------------------------------------------------------------------------

**#task5----------------------------------------------------------------------**

"Вклад в банке составляет x рублей. Ежегодно он увеличивается на p процентов, после чего дробная часть копеек отбрасывается. Определите, через сколько лет вклад составит не менее y рублей."

"Выражение «дробная часть копеек отбрасывается» означает, что если у вас оказалось 123.4567 рублей, т. е. 123 рубля и 45.67 копеек, то после округления у вас получится 123 рубля и 45 копеек, т.е. 123.45 рублей."

"Программа получает на вход три натуральных числа: x, p, y и должна вывести одно целое число."

#defaults

year = 0 #int

#input

print('Task 5: Вычесление, через сколько лет вклад в банк составит не менее заданого количества рублей.')

first\_sum = int(input('Введите сумму вклада в рублях: ')) #int в рублях

percent = int(input('Введите процентную ставку: ')) #int в процентах

last\_sum = int(input('Введите накопившуюся сумму в рублях: ')) #int в рублях

#main

if first\_sum < 0 or percent < 0 or last\_sum < 0:

print('Введено не допустимое значение!')

else:

while first\_sum < last\_sum:

first\_sum += first\_sum \* percent/100

first\_sum = round (first\_sum, 2)

year += 1

#output

print ('Через' ,year , ' года вклад в банк составит не менее', last\_sum) #int

print()

#--------------------------------------------------------------------------

**#task6**------------------------------------------------------------

"Программа получает на вход последовательность целых неотрицательных чисел, каждое число записано в отдельной строке. Последовательность завершается числом 0, при считывании которого программа должна закончить свою работу и вывести количество членов последовательности (не считая завершающего числа 0). Числа, следующие за числом 0, считывать не нужно"

#defaults

linght = 0 #int

#

print('Tast 6: Вычисление количества членов последовательности')

#main

while float (input('Введите неотрицательное число:')) != 0: #int; число больше нуля c фиксированой точкой; формат: 0.0.

linght += 1

else:

#output

print('Количество членов последовательности:' ,linght) #int

print()

#-------------------------------------------------------------------

**#task7**-----------------------------------------------------------

"Определите сумму всех элементов последовательности, завершающейся числом 0."

#defaults

sum = 0 #int

#input

print('Task 7: Вычисление суммы всех элементов последовательности')

number = float (input('Введите неотрицательное число:')) #float; формат: 0.0.

#main

while number != 0:

if number < 0:

print ('Введено отрицательное число!')

break

else:

sum += number

number = int(input('Введите неотрицательное число:'))

else:

#output

print('Сумма всех элементов последовательности равна =' , sum) #int

print()

#-----------------------------------------------------------------

**#task8**----------------------------------------------------------------

"Определите среднее значение всех элементов последовательности, завершающейся числом 0."

#defaults

sum = 0 #int

amount\_of\_numbers = 0 #int

#input

print('Task 8: Вычисление среднего значения последовательности')

number = int(input('Введите неотрицательное число:')) #int; целое число больше нуля

#main

while number != 0:

if number < 0:

print('Введено отрицательное число!')

break

else:

sum += number

amount\_of\_numbers += 1

number = int(input('Введите неотрицательное число:'))

else:

#output

print('Среднее значение всех елементов последовательности равно', sum/amount\_of\_numbers)

print()

#------------------------------------------------------

**#test9**------------------------------------------------------

"Последовательность состоит из натуральных чисел и завершается числом 0. Определите значение наибольшего элемента последовательности."

#defaults

max\_number = 0 #int

#input

print('Test 9: Определение значения наибольшого елемента последовательности')

number = int(input('Введите натуральное число: ')) #int

max\_number = number

#main

while number != 0:

if number < 0:

print ('Введено не натуральное число!')

break

else:

if number > max\_number:

max\_number = number

number = int(input('Введите натуральное число: '))

else:

#output

print('Наибольший элемент последовательности равен', max\_number)

print()

#--------------------------------------------------------------------

**#test10**----------------------------------------------------------------

"Последовательность состоит из натуральных чисел и завершается числом 0. Определите индекс наибольшего элемента последовательности. Если наибольших элементов несколько, выведите индекс первого из них. Нумерация элементов начинается с нуля."

#defaults

max = 0 #int

index\_of\_max = -1 #int;

index\_of\_numbers = 0 #int

#input

print('Task 10: Определение индекса максимума последовательности')

number = int(input('Введите натуральное число: ')) #int

#main

while number != 0:

if number <= 0:

print ('Введено не натуральное число!')

break

else:

if number > max\_number:

max\_number = number

index\_of\_max = index\_of\_numbers

index\_of\_numbers += 1

number = int(input('Введите натуральное число: '))

else:

#output

print('Индекс наибольшего элемента последовательности равен', index\_of\_max)

print()

#----------------------------------------------------------------------------

**#test11**------------------------------------------------------------------

"Определите количество четных элементов в последовательности, завершающейся числом 0."

#defoults

num\_even = 0 #int

#input

print('Task 11: Определение количества четных элементов последовательности')

number = int(input('Введите число: ')) # int; элементы последовательности

#main

while number != 0:

if number % 2 == 0:

num\_even += 1

number = int(input('Введите число: '))

#output

print('В последовательности ',num\_even, 'четных элемента') #int

print()

#--------------------------------------------------------------------------

**#test12**--------------------------------------------------------------------

"Последовательность состоит из натуральных чисел и завершается числом 0. Определите, сколько элементов этой последовательности больше предыдущего элемента."

#defoults

num = 0 #int

last\_number = 0 #int

#input

print('Task 12: Вычисление количества элементов, которые больше предыдущего')

number = int(input('Введите натуральное число: ')) #int; элементы последовательности

#main

while number != 0:

if number < 0:

print ('Введено не натуральное число!')

break

else:

if number > last\_number:

num +=1

last\_number = number

number = int(input('Введите натуральное число: '))

else:

print('В последовательности', num, 'элемента, которые небольше предыдущего') #int

print()

#----------------------------------------------------------------------------

**#task13**----------------------------------------------------------------------

"Последовательность состоит из различных натуральных чисел и завершается числом 0. Определите значение второго по величине элемента в этой последовательности. Гарантируется, что в последовательности есть хотя бы два элемента."

#input

print('Task 13: Определение значения второго по величине элемента в введенной последовательности')

first\_max = int (input('Введите натуральное число:')) #int

second\_max = int (input('Введите натуральное число:')) #int

number = int (input('Введите натуральное число:'))

#main

if first\_max <= 0 or second\_max <= 0:

print ('Введено не натуральное число!')

else:

if first\_max < second\_max:

first\_max, second\_max = second\_max, first\_max

while number != 0:

if number < 0:

print ('Введено не натуральное число!')

break

else:

if number > first\_max:

second\_max = first\_max

first\_max = number

elif number > second\_max:

second\_max = number

number = int (input('Введите натуральное число:'))

else:

print('Второй максимум последовательности равен', second\_max) #int

print()

#---------------------------------------------------------------------------

**#task14**--------------------------------------------------------------------

"Последовательность состоит из натуральных чисел и завершается числом 0. Определите, сколько элементов этой последовательности равны ее наибольшему элементу."

#defoults

maximal = 0 #int

num\_maximal = -1 #int

#input

print('Task 14: Определение количества элементов, равных максимуму')

number = int(input('Введите натуральное число')) #int

#input

while number != 0:

if number < 0:

print ('Введено не натуральное число!')

break

else:

if number > maximal:

maximal, num\_maximal = number, 1

elif number == maximal:

num\_maximal += 1

number = int(input('Введите натуральное число'))

else:

print(num\_maximal, 'элементов равно максимуму') #int

print()

#-------------------------------------------------------------------------

#**task15**-----------------------------------------------------------------------

"Последовательность Фибоначчи определяется так: φ0 = 0, φ1 = 1, φn = φn−1 + φn−2.По данному числу n определите n-е число Фибоначчи φn."

#defaults

first\_number = 0 #int

second\_number = 1 #int

sum\_numbers = 0 #int

counter = 1 #int

#input

print('Task 14: Числа Фибоначчи')

position = int(input('Введите позицию числа Фибоначчи'))

#main

if position < 0:

print('Введен не допустимый символ!')

else:

while counter < position:

counter += 1

sum\_numbers = first\_number + second\_number

first\_number, second\_number = second\_number, sum\_numbers

#result

print('На позиции',position,'стоит число Фибоначчи', sum\_numbers) #sum\_numbers is result = int

#------------------------------------------------------------------------------

#**task16-**-------------------------------------------------------------------

"Дано натуральное число A. Определите, каким по счету числом Фибоначчи оно является, то есть выведите такое число n, что φn = A. Если А не является числом Фибоначчи, выведите число -1."

#defaults

first\_number = 1 #int

second\_number = 1 #int

sum\_numbers = 0 #int

position = -1 #int

#input

number = int(input('Введите число Фибоначчи: ')) #int; целое число с фиксированной точкой

#main

if first\_number == number:

position = 1

elif second\_number == number:

position = 2

elif number > 0:

position = 2

while second\_number < number:

sum\_numbers = first\_number + second\_number

first\_number, second\_number = second\_number, sum\_numbers

position += 1

else:

if second\_number > number:

position = -1

#result

print('Позиция числа ',number,'является ',position) #int

#------------------------------------------------------------------------------------------------

#task17---------------------------------------------------------------

"Дана последовательность натуральных чисел, завершающаяся числом 0. Определите, какое наибольшее число подряд идущих элементов этой последовательности равны друг другу."

#defaults

prev\_number = -1 #int

num\_repetition = 0 #int

max\_num\_repetition = 0 #int

#input

print('Task 17: Вычисление наибольшего числа равных подряд идущих элементов введенной последовательности ')

number = int(input('Введите натуральное число: ')) #int

#main

while number != 0:

if number < 0:

print('Введено недопустимое значение!')

break

else:

if prev\_number == number:

num\_repetition += 1

else:

prev\_number = number

max\_num\_repetition = max(max\_num\_repetition, num\_repetition)

num\_repetition = 1

number = int(input('Введите натуральное число: '))

else:

#output

max\_num\_repetition = max(max\_num\_repetition, num\_repetition)

print('В последовательности', max\_num\_repetition, 'элементf, которые равны друг другу и идут по очереди') #int

print()

#--------------------------------------------------------------------------

#task18---------------------------------------------------------------

"Определите стандартное отклонение для данной последовательности натуральных чисел, завершающейся числом 0."

#defaults

from math import sqrt

partial\_sum = 0 #int

partial\_sum\_squares = 0 #int

n = 0 #int

#input

print('Task 18: Определение стандартного отклонения')

number = int(input("Введите натуральное число:")) #int

#math

while number != 0:

if number < 0:

print('Введено недопущеное число!')

break

else:

n += 1

partial\_sum += number

partial\_sum\_squares += number \*\* 2

number = int(input("Введите натуральное число:"))

else:

#output

res = (sqrt((partial\_sum\_squares - partial\_sum \*\* 2 / n) / (n - 1))) #float

print('Стандартное отклонение равно', res)

print()

#------------------------------------------------------------------------