**Циклы и условия**

**Домашняя работа**

Эта домашняя работа должна быть решена **БЕЗ** использования готовых функций (если именно они "решают" задачу) и сторонних библиотек. Используйте циклы и условия.

**1**

Напишите программу, которая будет запрашивать на вход числа (через запятую на одной строке) и выводить наибольшее значение из списка.  
  
*Ограничение: нельзя пользоваться готовой функцией для нахождения максимума (например, max()), готовыми функциями и методами сортировки (например, sort(),sorted()), функцией set().*  
  
*Формат ввода:*  
1, 2, 3, 6, 7

*Формат вывода:*  
7

In [2]:

*###*

maximum = 0

a = list(map(int, input().split(','))

**for** i **in** a:

**if** i > maximum:

maximum = i

**else**:

**continue**

print (maximum)

*###*

0

**2**

Напишите программу, которая будет запрашивать на вход числа (каждое с новой строки) до тех пор, пока не будет введен ноль (0). На выход должно выводиться второе по величине значение.  
  
*Ограничение: нельзя пользоваться готовой функцией для нахождения максимума (например, max()), готовыми функциями и методами сортировки (например, sort(),sorted()).*  
  
*Формат ввода:*  
4  
6  
7  
7  
0

*Формат вывода:*  
6

In [0]:

*###*

a\_1 = 0

b\_2 = 0

x = int(input())

**while** x != 0:

**if** x > a\_1:

a\_1 = x

b\_2 = a\_1

**elif** a\_1 > x > b\_2:

b\_2 = x

**else**:

**continue**

print (b\_2)

*###*

**3**

Напишите программу, которая принимает на вход год, а на выход выдает количество дней в этом году.  
  
*Формат ввода:*  
2020

*Формат вывода:*  
366

In [0]:

*###*

a = int (input ())

**if** (a % 4 == 0) **is** **True**:

print (366)

**else**:

print (365)

*###*

**4**

Напишите программу, которая на вход получает координаты двух клеток шахматной доски и выводит соощение о том, являются ли эти клетки одного цвета.  
  
*Формат ввода:*  
c2  
e4

*Формат вывода:*  
Клетки одинакового цвета

In [0]:

*###*

white = 0

black = 0

enter1 = input("Введите координаты первой клетки ")

k = list(enter1)

enter2= input("Введите координаты второй клетки ")

l = list(enter2)

k[1] = int(k[1])

l[1] = int(l[1])

**if** (k[0] == ('a' **or** 'c' **or** 'e' **or** 'g')) **is** **True**:

**if** (k[1] % 2 == 0) **is** **True**:

k = 'white'

**else**:

k = 'black'

**else**:

**if** (k[1] % 2 == 0) **is** **True**:

k = 'black'

**else**:

k ='white'

**if** (l[0] ==('a' **or** 'c' **or** 'e' **or** 'g')) **is** **True**:

**if** (l[1] % 2 == 0) **is** **True**:

l = 'white'

**else**:

l = 'black'

**else**:

**if** (l[1] % 2 == 0) **is** **True**:

l = 'black'

**else**:

l = 'white'

**if** (k == 'white' **and** l == 'white') **is** **True**:

print ("Клетки одинаковые")

**elif**(k == 'black' **and** l == 'black') **is** **True**:

print ("Клетки одинаковые")

**else**:

print ("Клетки разные")

*###*

**5**

Напишите программу, которая на вход получает число, а на выходе сообщает, простое это число или составное.  
  
*Формат ввода:*  
169

*Формат вывода:*  
Составное число

In [0]:

*###*

a = int(input ('Введите натуральное число, кроме 1: '))

**if** (a == 2) **is** **True**:

print ('Простое число')

**else**:

my\_list = list(range(1, a))

n = list()

**for** i **in** my\_list:

**if** ((a % i) != 0)**is** **True**:

n.append(1)

s = len(n)

**if** (s >= 1):

print ("Простое число")

**else**:

print ('Составное число')

*###*

**6**

Напишите программу, которая на вход получает целое число больше 2 и выводит по нему его наименьший натуральный делитель, отличный от 1.  
  
*Формат ввода:*  
169

*Формат вывода:*  
13

In [0]:

*###*

num = int (input ('Введите натуральное число, больше 2: '))

my\_list = list(range(3, (num + 1)))

**for** i **in** range(3, (num + 1)):

**if** ((num % i) != 0) **is** **True**:

i +=1

**continue**

**else**:

print(i)

**break**

*###*

**7**

Напишите программу, которая поможет составить план тренировок для подготовки к марафону. Она получает на вход число километров на планируемом марафоне, сколько пользователь планирует пробежать в первый день тренировок и на сколько процентов планирует увеличивать каждый день это расстояние. На выходе программа должна выдавать, сколько дней пользователю потребуется для того, чтобы подготовиться пробежать целевое количество километров.  
  
*Ограничение: нельзя пользоваться функцией ceil() из модуля math и ее аналогами.*  
  
*Формат ввода:*  
12  
3  
10

*Формат вывода:*  
16

In [0]:

*###*

a = int (input ('Введите, сколько километров планируется пробежать на марафоне '))

b = int (input ("Введите, сколько километров планируется пробежать в первый день тренировок "))

c = int (input ("Введите, на сколько процентов планируется увеличивать ежедневные тренировки "))

f = float (c / 100)

d = 1

**while** b < a:

b = b + (b \* f)

d += 1

**else**:

print (d)

*###*

**8**

Напишите программу, которая на вход получает число n и считает по нему сумму 1²+2²+3²+...+n².  
  
*Ограничение: нельзя пользоваться функцией sum() и ее аналогами.*

*Формат ввода:*  
3

*Формат вывода:*  
14

In [0]:

*###*

n = int(input("Введите любое целое число больше единицы "))

n1 = n + 1

b = 0

**for** i **in** range((n1)):

a = i \*\* 2

b += a

print(b)

*###*

**9**

Напишите программу, которая на вход получает число n и считает по нему сумму сумму 1! + 2! + 3! + ... + n!  
  
*Ограничение: нельзя пользоваться готовой функцией factorial() из модуля math, функцией sum() и их аналогами.*

*Формат ввода:*  
3

*Формат вывода:*  
9

In [0]:

*###*

n = int(input("Введите любое целое число больше единицы "))

b = 1

c = 0

**for** i **in** range(1, n+1):

c = c+ (b \*i)

b =i

print(c)

*###*

**10**

Напишите программу, которая получает на вход последовательность чисел (каждое число с новой строки до того момента, пока пользователь не введет 0) и считает количество четных элементов в последовательности.  
  
*Формат ввода:*  
5  
4  
2  
7  
8  
9  
0

*Формат вывода:*  
3

In [0]:

*###*

c = 0

**while** **True**:

z = int(input())

**if** z == 0:

**break**

**elif** **not** z % 2:

c += 1

print(c)

*###*

**11**

Напишите программу, которая формирует список игроков женской команды по мини-футболу. Программа должна записывать возраст и пол претендента. Возраст должен запрашиваться после пола и только в том случае, если пол претендента женский. Если пол претендента мужской, программа должна сообщать о том, что он не подходит. Возраст претенденток должен быть от 18 до 35 лет. Если кандидат удовлетворяет требованиям, должно появляться соответствующее сообщение. Всего в команде могут быть только шесть человек. Когда необходимое число набирается, запись закрывается и выводится сообщение "Запись в команду закрыта".

In [0]:

*###*

pretendents = input ("Имя претендента ")

sex = input ('')

**while** pretendents <= 6:

**if** sex != female:

print ("Это женская команда по мини-футболу, мы не можем принять вас в команду. Приходите на наши соревнования")

**elif** age >= 18 **and** <= 35:

print ('Поздравляем, вы приняты в команду!')

**else**:

print('Жаль, но в команде принимают участие женщины от 18 до 35 лет')

**else**:

print ("Сожалеем, запись закрыта")

**def** pretendent (sex, age)

sex = female

age >= 18 **and** <= 35

**return** sex, age

*###*

**12**

Напишите программу, которая на вход получает максимальную ширину ромба и рисует его. Гарантируется, что входное число всегда нечетное.

*Формат ввода:*  
5

*Формат вывода:*

\*

\*\*\*

\*\*\*\*\*

\*\*\*

\*

In [0]:

*###*

a = int (input('Введите максимальную ширину ромба '))

z = int((a - 1) / 2)

k = 1

s = 1

m = 1

p = a - 2

**while** s <= ((a + 1 ) / 2):

print ((z \* ' '),('\*' \* k))

z = z - 1

k += 2

s += 1

**while** s <= a:

print ((m \* ' '), (('\*') \* p))

m = m + 1

p = p - 2

s += 1

*###*

**13**

Напишите программу, которая запрашивает у пользователя сторону квадрата и символ, а затем рисует этот символ по диагоналям квадрата. Гарантируется, что входное число всегда нечетное.

*Формат ввода:*  
5  
#

*Формат вывода:*

# #

# #

#

# #

# #

In [0]:

*###*

a = int (input ('Введите длину стороны квадрата '))

b = input ('Введите знак ')

s = 1

c = a - 3

z = 0

**while** s < ((a + 1) / 2):

print ((z \* ' '), b, (c \* ' '), b)

z += 1

c -= 2

s += 1

**else**:

print ((z \* ' '), b,)

z -= 1

c = 0

s += 1

**while** s <= a:

print ((z \* ' '), b, (c \* ' '), b)

c += 2

s += 1

z -= 1

*###*

**14**

В корзине лежат шары. Если разложить их в кучи по 2, останется один. Если разложить в кучи по 3, останется один. Если разложить в кучи по 4, останется один. Если разложить в кучи по 5, останется один. Если разложить в кучи по 6, останется один. Если разложить в кучи по 7, не будет остатка. Нужно найти минимальное количество шаров, удовлетворяющее условию.

In [0]:

*###*

**for** i **in** range (7,1000000000,1):

**while** ((i % 2 == 1) **and** (i % 3 == 1) **and** (i % 4 == 1) **and** (i % 5 == 1) **and** (i % 6 == 1)

**and** (i % 7 == 0)):

print ('Минимальное число шаров в корзине ',i)

**break**

*###*

**15**

Рабочие клеили обои на стены. Первую стену поклеили за M минут, а каждую следующую на 5 минут больше, чем предыдущую. Напишите программу, которая запрашивает сколько стен было в квартире под поклейку, а также время работы с первой стеной в минутах. Программа должна выводить, сколько **часов** рабочие потратили на поклейку обоев во всей квартире. Час отсчитывается с первой минуты. Ответом должно быть целое число.  
  
*Ограничение: нельзя пользоваться готовыми функциями bool(), floor() и их аналогами.*  
  
*Формат ввода:*  
6  
10

*Формат вывода:*  
3

In [0]:

*###*

a = int (input ("Введите количество стен "))

b = int (input ("Введите продолжительность работы с первой стеной,мин "))

z = 0

c = 0

**while** z < a **and** a > 1:

c +=b

b +=5

z += 1

**else**:

**if** b % 60 != 0:

c = b //60

**else**:

c = b / 60

print(c)

*###*

**16**

Напишите программу, которая убирает из списка повторяющиеся элементы. Программа должна запрашивать на вход слова, каждое с новой строки, пока пользователь не введет пустую строку. Затем должна выводить список без повторяющихся элементов.  
  
*Ограничение: нельзя пользоваться готовой функцией set() и ее аналогами.*

*Формат ввода:*  
Алиса  
Алиса  
Алиса  
Петя

*Формат вывода:*  
[Алиса, Петя]

In [0]:

*###*

a = input ()

my\_set = []

**while** a != ' ':

**if** a **in** my\_set:

a = input ()

**else**:

my\_set.append(a)

a = input ()

**else**:

print (my\_set)

*###Напишите программу, которая выводит число пар одинаковых элементов в списке. При этом если два элемента образовали пару, они не могут вступить в пару с другими элементами. Программа должна запрашивать на вход слова, каждое с новой строки, пока пользователь не введет пустую строку.*

**17**

Напишите программу, которая выводит число пар одинаковых элементов в списке. При этом если два элемента образовали пару, они не могут вступить в пару с другими элементами. Программа должна запрашивать на вход слова, каждое с новой строки, пока пользователь не введет пустую строку.  
  
*Формат ввода:*  
Алиса  
Алиса  
Алиса

*Формат вывода:*  
1

In [0]:

*###*

d[]= {}

**while** **break**:

**if** i **in** d **is** 0:

i=1

**else**:

i+=1

**for** i **in** d( ):

i=1+i//2

print (i)

*###*

**18**

Будем считать, что кубик может иметь неограниченное количество граней (натуральное число). Напишите программу, которая запрашивает, сколько граней имеется у двух разных кубиков. Затем выводит все возможные комбинации результатов бросков двух таких кубиков.  
  
*Формат ввода:*  
3  
4

*Формат вывода:*  
1 1  
1 2  
1 3  
1 4  
2 1  
2 2  
2 3  
2 4  
3 1  
3 2  
3 3  
3 4

In [0]:

*###*

a = input ()

b = input ()

**for** i **in** range (1, a+1):

**for** q **in** range (1, b+1)

print (i,q)

*###*