

1. Напишите программу, которая выводит на экран текст «Здравствуй, мир!» (без кавычек).

```
print ("Здравствуй, мир!")
```

2. В популярном сериале «Остаться в живых» использовалась последовательность чисел 4 8 15 16 23 42, которая принесла героям удачу и помогла сорвать джекпот в лотерее. Напишите программу, которая выводит данную последовательность чисел с одним пробелом между ними.

```
print ("4", "8", "15", "16", "23", "42")
```

3. Измените предыдущую программу так, чтобы каждое число последовательности 4 8 15 16 23 42 печаталось на отдельной строке.

```
print ("4")
print ("8")
print ("15")
print ("16")
print ("23")
print ("42")
```

4. Напишите программу, которая выводит указанный треугольник, состоящий из звездочек (\*).

```
print (*)
print (**)
print (***)
print (****)
print (*****)
print (******)
print (*****)
print (*****)
```

5. На вход программе подается строка текста – имя человека. Напишите программу, которая выводит на экран приветствие в виде слова «Привет» (без кавычек), после которого должна стоять запятая и пробел, а затем введенное имя.

```
name=input()
print("Привет,", name)
```

6. На вход программе подается строка текста – название футбольной команды. Напишите программу, которая повторяет ее на экране со словами « - чемпион!» (без кавычек).

```
team=input()
print(team, "- чемпион!")
```

7. Напишите программу, которая считывает три строки по очереди, а затем выводит их в той же последовательности, каждую на отдельной строчке.

```
a=input()
print(a)
b=input()
print(b)
c=input()
print(c)
```

8. Напишите программу, которая считывает три строки по очереди, а затем выводит их в обратной последовательности, каждую на отдельной строчке.

```
a=input()
b=input()
c=input()
print(c)
print(b)
print(a)
```

9. Напишите программу, которая выводит на экран текст «I\*\*\*like\*\*\*Python» (без кавычек).

```
print("I", "like", "Python", sep="***")
```

10. Напишите программу, которая считывает строку-разделитель и три строки, а затем выводит указанные строки через разделитель.

```
a=input()
b=input()
c=input()
d=input()
print(b, c, d, sep=a)
```

11. Напишите программу, которая приветствует пользователя, выводя слово «Привет» (без кавычек), после которого должна стоять запятая и пробел, а затем введенное имя и восклицательный знак.

```
a=input()
print("Привет,", a, end="!")
```

12. Напишите программу вывода на экран трех последовательно идущих чисел, каждое на отдельной строке. Первое число вводит пользователь, остальные числа вычисляются в программе.

```
num1 = int(input())
num2 = num1 + 1
num3 = num1 + 2
print(num1, num2, num3, sep = '\n')
```

13. Напишите программу, которая считывает три целых числа и выводит на экран их сумму. Каждое число записано в отдельной строке.

```
num1 = int(input())
num2 = int(input())
num3 = int(input())
print(num1 + num2 + num3)
```

14. Напишите программу, вычисляющую объём куба и площадь его полной поверхности, по введённому значению длины ребра.

```
a = int(input())
print('Объем =', a * a * a)
print('Площадь полной поверхности =', 6 * (a * a))
```

15. Напишите программу вычисления значения функции  $f(a, b) = 3(a + b)^3 + 275b^2 - 127a - 41$  по введенным целым значениям  $a$  и  $b$ .

```
a = int(input())
b = int(input())
print(3 * ((a + b) * (a + b) * (a + b)) + 275 * (b * b) - 127 * a - 41)
```

16. Напишите программу, которая считывает целое число, после чего на экран выводится следующее и предыдущее целое число с пояснительным текстом.

```
a = int(input())
b = int(input())
print(3 * ((a + b) * (a + b) * (a + b)) + 275 * (b * b) - 127 * a - 41)
```

17. Напишите программу, которая считывает целое число, после чего на экран выводится следующее и предыдущее целое число с пояснительным текстом.

```
num1 = int(input())
print('Следующее за числом', num1, 'число:', num1 + 1)
print('Для числа', num1, 'предыдущее число:', num1 - 1)
```

18. Напишите программу, которая считает стоимость трех компьютеров, состоящих из монитора, системного блока, клавиатуры и мыши.

```
scr = int(input())
sys = int(input())
keyb = int(input())
mou = int(input())
print(scr + sys + keyb + mou) * 3
```

19. Напишите программу, в которой вычисляется сумма, разность и произведение двух целых чисел, введенных с клавиатуры.

```
a = int(input())
b = int(input())
print(a, '+', b, '=', a + b)
print(a, '-', b, '=', a - b)
print(a, '*', b, '=', a * b)
```

20. Арифметической прогрессией называется последовательность чисел  $a_1, a_2, \dots, a_n$ , каждое из которых, начиная с  $a_2$ , получается из предыдущего прибавлением к нему одного и того же постоянного числа  $d$  (разность прогрессии). Программа должна вывести  $n$ -ый член арифметической прогрессии.

```
a1 = int(input())
d = int(input())
n = int(input())
print(a1 + d * (n - 1))
```

21. Напишите программу, которая считывает целое положительное число  $x$  и выводит на экран последовательность чисел  $x, 2x, 3x, 4x$  и  $5x$ , разделённых тремя черточками.

```
x = int(input())
print(x, 2 * x, 3 * x, 4 * x, 5 * x, sep = '---')
```

22. Геометрической прогрессией называется последовательность чисел  $b_1, b_2, \dots, b_n$ , каждое из которых, начиная с  $b_2$ , получается из предыдущего умножением на одно и то же постоянное число  $q$  (знаменатель прогрессии). На вход программе подаётся три целых числа:  $b_1, q$  и  $n$ , каждое на отдельной строке. Программа должна вывести  $n$ -ый член геометрической прогрессии.

```
b1 = int(input())
q = int(input())
n = int(input())
print(b1 * q ** (n - 1))
```

23. Напишите программу, которая находит полное число метров по заданному числу сантиметров. На вход программе подаётся натуральное число – количество сантиметров. Программа должна вывести одно число – полное число метров.

```
x = int(input())
print(x // 100)
```

24.  $n$  школьников делят  $k$  мандаринов поровну, неделящийся остаток остается в корзине. Сколько целых мандаринов достанется каждому школьнику? Сколько целых мандаринов останется в корзине? На вход программе подаётся два целых числа: количество школьников и количество мандаринов, каждое на отдельной строке. Программа должна вывести два числа: количество мандаринов, которое достанется каждому школьнику, и количество мандаринов, которое останется в корзине, каждое на отдельной строке.

```
n = int(input())
k = int(input())
print(k // n)
print(k % n)
```

25. Безумный титан Танос собрал все 6 камней бесконечности и намеревается уничтожить половину населения Вселенной по щелчку пальцев. При этом если население Вселенной является нечетным числом, то титан проявит милосердие и округлит количество выживших в большую сторону. Помогите Мстителям подсчитать количество выживших. На вход дается число целое  $n$  – население Вселенной. Программа должна вывести одно число – количество выживших.

```
n = int(input())
print((n + 1) // 2)
```

26. В купейном вагоне имеется 99 купе с четырьмя местами для пассажиров в каждом. Напишите программу, которая определяет номер купе, в котором находится место с заданным номером (нумерация мест сквозная, начинается с 1). На вход программе подаётся целое число – место с заданным номером в вагоне. Программа должна вывести одно число – номер купе, в котором находится указанное место.

```
n = int(input())
print((n + 3) // 4)
```

27. Напишите программу для пересчёта величины временного интервала, заданного в минутах, в величину, выраженную в часах и минутах. На вход программе подаётся целое число – количество минут. Программа должна вывести текст в соответствии с условием задачи.

```
t = int(input())
print(t, 'мин - это', t // 60, 'час', t % 60, 'минут.')
```

28. Напишите программу, в которой рассчитывается сумма и произведение цифр положительного трёхзначного числа. На вход программе подаётся положительное трёхзначное число. Программа должна вывести два числа с поясняющим текстом: сумма цифр и произведение цифр.

```
x = int(input())
digit3 = x % 10
digit2 = (x % 100) // 10
digit1 = x // 100
print('Сумма цифр =', digit1 + digit2 + digit3)
print('Произведение цифр =', digit1 * digit2 * digit3)
```

29. Дано трехзначное число  $abc$ , в котором все цифры различны. Напишите программу, которая выводит шесть чисел, образованных при перестановке цифр заданного числа. На вход программе подаётся положительное трёхзначное целое число, все цифры которого различны. Программа должна вывести шесть чисел, образованных при перестановке цифр заданного числа в следующем порядке:  $abc, acb, bac, bca, cab, cba$ .

```
x = int(input())
c = x % 10
b = (x % 100) // 10
a = x // 100
print(a * 100 + b * 10 + c)
print(a * 100 + c * 10 + b)
print(b * 100 + a * 10 + c)
print(b * 100 + c * 10 + a)
print(c * 100 + a * 10 + b)
print(c * 100 + b * 10 + a)
```

30. Напишите программу для нахождения цифр четырёхзначного числа. На вход программе подаётся положительное четырёхзначное целое число. Программа должна вывести текст в соответствии с условием задачи.

```
x = int(input())
a4 = x % 10
a3 = (x % 100) // 10
a2 = (x % 1000) // 100
a1 = x // 1000
print('Цифра в позиции тысяч равна', a1)
print('Цифра в позиции сотен равна', a2)
print('Цифра в позиции десятков равна', a3)
print('Цифра в позиции единиц равна', a4)
```

31. Напишите программу, которая выводит прямоугольник, по периметру состоящий из звездочек (\*). Высота и ширина прямоугольника равны 44 и 1717 звёздочкам соответственно.

```
print('*****')
print('*          *')
print('*          *')
print('*****')
```

32. Напишите программу, которая считывает два целых числа  $a$  и  $b$  и выводит на экран квадрат суммы  $(a+b)^2$  и сумму квадратов  $a^2+b^2$  этих чисел.

```
a = int(input())
b = int(input())
print('Квадрат суммы', a, 'и', b, 'равен', (a + b) ** 2)
print('Сумма квадратов', a, 'и', b, 'равна', a ** 2 + b ** 2)
```

33. Как известно, целые числа в языке Python не имеют ограничений, которые встречаются в других языках программирования. Напишите программу, которая считывает четыре целых положительных числа  $a$ ,  $b$ ,  $c$  и  $d$  и выводит на экран значение выражения  $a^b+c^d$ .

```
a = int(input())
b = int(input())
c = int(input())
d = int(input())
print(a ** b + c ** d)
```

34. Напишите программу, которая считывает целое положительное число  $n$ ,  $n \in [1;9]$  и выводит значение числа  $n+nn+nnn$ .

```
n = int(input())
print(n + (n * 10 + n) + (n * 10 + n * 100 + n))
```

35. При регистрации на сайтах требуется вводить пароль дважды. Это сделано для безопасности, поскольку такой подход уменьшает возможность неверного ввода пароля. Напишите программу, которая сравнивает пароль и его подтверждение. Если они совпадают, то программа выводит: «Пароль принят», иначе: «Пароль не принят». На вход программе подаются две строки. Программа должна вывести одну строку в соответствии с условием задачи.

```
pass1, pass2 = input(), input()
if pass1 == pass2:
    print('Пароль принят')
else:
    print('Пароль не принят')
```

36. Напишите программу, которая определяет, является ли число четным или нечетным. На вход программе подаётся одно целое число. Программа должна вывести «Четное», если число четное, и «Нечетное» — если число нечетное.

```
num = int(input())
if num % 2 == 0:
    print('Четное')
else:
    print('Нечетное')
```

37. Напишите программу, которая проверяет, что для заданного четырехзначного числа выполняется следующее соотношение: сумма первой и последней цифр равна разности второй и третьей цифр. На вход программе подаётся одно целое положительное четырёхзначное число. Программа должна вывести «ДА», если соотношение выполняется, и «НЕТ» — если не выполняется.

```
num = int(input())
num4 = num % 10
num3 = (num % 100) // 10
num2 = (num % 1000) // 100
num1 = num // 1000
if (num1 + num4) == (num2 - num3):
    print('ДА')
else:
    print('НЕТ')
```

38. Напишите программу, которая определяет, разрешен ли пользователю доступ к интернет-ресурсу или нет. На вход программе подаётся целое число — возраст пользователя. Программа должна вывести текст «Доступ разрешен» если возраст не менее 18, и «Доступ запрещен» в противном случае.

```
age = int(input())
if age >= 18:
    print('Доступ разрешен')
else:
    print('Доступ запрещен')
```

39. Напишите программу, которая определяет, являются ли три заданных числа (в указанном порядке) последовательными членами арифметической прогрессии. На вход программе подаются три числа, каждое на отдельной строке. Программа должна вывести «YES» или «NO» (без кавычек) в соответствии с условием задачи.

```
a, b, c = int(input()), int(input()), int(input())
if (b - a) == (c - b):
    print('YES')
else:
    print('NO')
```

40. Напишите программу, которая определяет наименьшее из двух чисел. На вход программе подаётся два различных целых числа. Программа должна вывести наименьшее из двух чисел.

```
a, b = int(input()), int(input())
if a - b < 0:
    print(a)
else:
    print(b)
```

41. Напишите программу, которая определяет наименьшее из четырёх чисел. На вход программе подаётся четыре целых числа. Программа должна вывести наименьшее из четырёх чисел.

```
a, b, c, d = int(input()), int(input()), int(input()), int(input())
if a < b < c < d:
    print(a)
else:
    if b < c < d:
        print(b)
    else:
        if c < d:
            print(c)
        else:
            print(d)
```

42. Напишите программу, которая по введённому возрасту пользователя сообщает, к какой возрастной группе он относится:

- до 13 включительно – детство;
- от 14 до 24 – молодость;
- от 25 до 59 – зрелость;
- от 60 – старость.

На вход программе подаётся одно целое число – возраст пользователя. Программа должна вывести название возрастной группы.

```
age = int(input())
if age <= 13:
    print('детство')
else:
    if age <= 24:
        print('молодость')
```

```
else:
    if age <=59:
        print('зрелость')
    else:
        print('старость')
```

**43. Напишите программу, которая считывает три числа и подсчитывает сумму только положительных чисел. На вход программе подаются три целых числа. Программа должна вывести одно число – сумму положительных чисел.**

```
a, b, c = int(input()), int(input()), int(input())
count1 = 0
count2 = 0
count3 = 0
if a > 0:
    count1 = a
else:
    count1 = 0
if b > 0:
    count2 = b
else:
    count2 = 0
if c > 0:
    count3 = c
else:
    count3 = 0
print(count1 + count2 + count3)
```

**44. Напишите программу, которая принимает целое число  $x$  и определяет, принадлежит ли данное число указанному промежутку. На вход программе подаётся целое число  $x$ . Программа должна вывести текст в соответствии с условием задачи. Если точка выколота, то граница не включается, если точка закрашенная, то граница включается.**

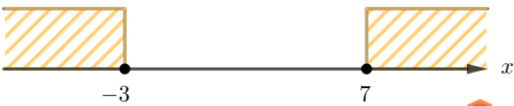


$x$



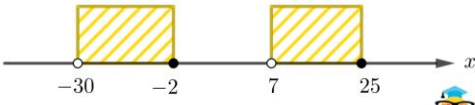
```
x = int(input())
if -1 < x < 17:
    print('Принадлежит')
else:
    print('Не принадлежит')
```

**45. Напишите программу, которая принимает целое число  $x$  и определяет, принадлежит ли данное число указанным промежуткам. На вход программе подаётся целое число  $x$ . Программа должна вывести текст в соответствии с условием задачи.**



```
x = int(input())
if x <= -3 or x >= 7:
    print('Принадлежит')
else:
    print('Не принадлежит')
```

**46. Напишите программу, которая принимает целое число  $x$  и определяет, принадлежит ли данное число указанным промежуткам. На вход программе подаётся целое число  $x$ . Программа должна вывести текст в соответствии с условием задачи.**



```
x = int(input())
if -30 < x <= -2 or 7 < x <= 25:
    print('Принадлежит')
else:
    print('Не принадлежит')
```

**47. Назовем число красивым, если оно является четырехзначным и делится нацело на 77 или на 1717. Напишите программу, определяющую, является ли введённое число красивым. Программа должна вывести «YES», если число является красивым, или «NO» в противном случае. На вход программе подаётся натуральное число.**

```
x = int(input())
if (1000 <= x <= 9999) and (x % 7 == 0 or x % 17 == 0):
    print('YES')
else:
    print('NO')
```

**48. Напишите программу, которая принимает три положительных числа и определяет, существует ли невырожденный треугольник с такими сторонами. На вход программе подаётся три положительных целых числа.**

```
a, b, c = int(input()), int(input()), int(input())
if (a + b > c) and (a + c > b) and (b + c > a):
    print('YES')
else:
    print('NO')
```

**49. Напишите программу, которая определяет, является ли год с данным номером високосным. Если год является високосным, то выведите «YES», иначе выведите «NO». Год является високосным, если его номер кратен 4, но не кратен 100, или если он кратен 400. На вход программе подаётся натуральное число.**

```
x = int(input())
if x % 4 == 0 and not (x % 100 == 0) or x % 400 == 0:
    print('YES')
else:
    print('NO')
```

**50. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли ладья попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом ладьи можно попасть во вторую, или «NO» в противном случае. На вход программе подаётся четыре числа от 1 до 8.**

```
a, b, c, d = int(input()), int(input()), int(input()), int(input())
if (a != c and b == d) or (a == c and b != d):
    print('YES')
else:
    print('NO')
```

**51. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли король попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом короля можно попасть во вторую, или «NO» в противном случае. На вход программе подаётся четыре числа от 1 до 8.**

```
a, b, c, d = int(input()), int(input()), int(input()), int(input())
if (c <= a + 1 and c >= a - 1) and (d <= b + 1 and d >= b - 1):
    print('YES')
else:
    print('NO')
```

52. Зум бросил вызов Флэшу и предложил ему честный поединок в виде гонки вокруг магнетара. В случае проигрыша эта нейтронная звезда зарядится и уничтожит мир, поэтому Флэш решил не рисковать без причины, и узнать у своего друга Циско Рамона есть ли смысл принимать вызов. Циско получил данные, что скорость Зума равна  $n$ , а скорость Флэша равна  $k$ . Напишите программу, которая должна вывести ответ Циско на вопрос Флэша.

На вход программе подаётся два целых числа  $n$  и  $k$ , скорость Зума и Флэша. Если Зум быстрее Флэша нужно вывести «NO», если Флэш быстрее Зума нужно вывести «YES», если их скорости равны нужно вывести "Don't know".

```
n, k = int(input()), int(input())
if n > k:
    print('NO')
else:
    if n < k:
        print('YES')
    else:
        if n == k:
            print("Don't know")
```

53. Напишите программу, которая принимает три положительных числа и определяет вид треугольника, длины сторон которого равны введенным числам. На вход программе подаются три числа – длины сторон существующего треугольника. Программа должна вывести на экран текст – вид треугольника («Равносторонний», «Равнобедренный» или «Разносторонний»).

```
a, b, c = int(input()), int(input()), int(input())
if a == b == c:
    print('Равносторонний')
elif a != b != c and a != c:
    print('Разносторонний')
else:
    print('Равнобедренный')
```

54. Даны три различных целых числа. Напишите программу, которая находит среднее по величине число. На вход программе подаётся три различных целых числа, каждое на отдельной строке. Программа должна вывести среднее число. Средним называется число, которое будет вторым, если три числа отсортировать в порядке возрастания.

```
a = int(input())
b = int(input())
c = int(input())
if b < a < c or c < a < b:
    print(a)
elif a < b < c or c < b < a:
    print(b)
else:
    print(c)
```

55. Дан порядковый номер месяца (1,2,..., 12). Напишите программу, которая выводит на экран количество дней в этом месяце. Принять, что год является невисокосным. На вход программе подаётся одно целое число – порядковый номер месяца. Программа должна вывести количество дней в этом месяце.

```
month = int(input())
if month == 1 or month == 3 or month == 5 or month == 7 or month == 8 or month == 10 or month == 12:
    print('31')
elif month == 2:
    print('28')
else:
    print('30')
```

56. Известен вес боксера-любителя (целое число). Известно, что вес таков, что боксер может быть отнесён к одной из трех весовых категорий:

1. Легкий вес – до 60 кг;
2. Первый полусредний вес – до 64 кг;
3. Полусредний вес – до 69 кг.

Напишите программу, определяющую, в какой категории будет выступать данный боксер. На вход программе подаётся одно целое число. Программа должна вывести текст – название весовой категории.

```
weight = int(input())
if weight < 60:
    print('Легкий вес')
elif 60 <= weight < 64:
    print('Первый полусредний вес')
else:
    print('Полусредний вес')
```

57. Напишите программу, которая считывает с клавиатуры два целых числа и строку. Если эта строка является обозначением одной из четырёх математических операций (+, -, \*, /), то выведите результат применения этой операции к введённым ранее числам, в противном случае выведите «Неверная операция». Если пользователь захочет поделить на ноль, выведите текст «На ноль делить нельзя!».

На вход программе подаются два целых числа, каждое на отдельной строке, и строка. Программа должна вывести результат применения операции к введенным числам или соответствующий текст, если операция неверная либо если происходит деление на ноль.

```
a = int(input())
b = int(input())
c = input()
if c == '+':
    print(a + b)
elif c == '-':
    print(a - b)
elif c == '*':
    print(a * b)
elif c == '/' and b != 0:
    print(a / b)
elif c == '/' and b == 0:
    print('На ноль делить нельзя!')
else:
    print('Неверная операция')
```

58. Красный, синий и желтый называются основными цветами, потому что их нельзя получить путем смешения других цветов. При смешивании двух основных цветов получается вторичный цвет:

- если смешать красный и синий, то получится фиолетовый;
- если смешать красный и желтый, то получится оранжевый;
- если смешать синий и желтый, то получится зеленый.

Напишите программу, которая считывает названия двух основных цветов для смешивания. Если пользователь вводит что-нибудь помимо названий «красный», «синий» или «желтый», то программа должна вывести сообщение об ошибке. В противном случае программа должна вывести название вторичного цвета, который получится в результате.

На вход программе подаются две строки, каждая на отдельной строке. Программа должна вывести полученный цвет смешения либо сообщение «ошибка цвета», если введён был не цвет. Если смешать красный и красный, то получится красный и т.д.

```
color1, color2 = input(), input()
if (color1 != 'красный' and color1 != 'синий' and color1 != 'желтый') or (color2 != 'красный' and color2 != 'синий' and color2 != 'желтый'):
    print('ошибка цвета')
elif color1 == color2:
    print(color1)
elif color1 == 'красный' and color2 == 'синий':
    print('фиолетовый')
elif color1 == 'красный' and color2 == 'желтый':
    print('оранжевый')
elif color1 == 'синий' and color2 == 'желтый':
    print('зеленый')
elif color2 == 'красный' and color1 == 'синий':
    print('фиолетовый')
elif color2 == 'красный' and color1 == 'желтый':
    print('оранжевый')
```

```
elif color2 == 'синий' and color1 == 'желтый':
    print('зеленый')
```

59. На колесе рулетки карманы пронумерованы от 0 до 36. Ниже приведены цвета карманов:

- карман 0 зеленый;
- для карманов с 1 по 10 карманы с нечетным номером имеют красный цвет, карманы с четным номером – черный;
- для карманов с 11 по 18 карманы с нечетным номером имеют черный цвет, карманы с четным номером – красный;
- для карманов с 19 по 28 карманы с нечетным номером имеют красный цвет, карманы с четным номером – черный;
- для карманов с 29 по 36 карманы с нечетным номером имеют черный цвет, карманы с четным номером – красный.

Напишите программу, которая считывает номер кармана и показывает, является ли этот карман зеленым, красным или черным. Программа должна вывести сообщение об ошибке, если пользователь вводит число, которое лежит вне диапазона от 0 до 36.

На вход программе подаётся одно целое число. Программа должна вывести цвет кармана либо сообщение «ошибка ввода», если введённое число лежит вне диапазона от 0 до 36.

```
x = int(input())
if x < 0 or x > 36:
    print('ошибка ввода')
elif x == 0:
    print('зеленый')
elif 0 < x <= 10 and x % 2 == 0:
    print('черный')
elif 0 < x <= 10 and x % 2 == 1:
    print('красный')
elif 11 <= x <= 18 and x % 2 == 0:
    print('красный')
elif 11 <= x <= 18 and x % 2 == 1:
    print('черный')
elif 19 <= x <= 28 and x % 2 == 0:
    print('черный')
elif 19 <= x <= 28 and x % 2 == 1:
    print('красный')
elif 29 <= x <= 36 and x % 2 == 0:
    print('красный')
elif 29 <= x <= 36 and x % 2 == 1:
    print('черный')
```

60. На числовой прямой даны два отрезка:  $[a_1; b_1]$  и  $[a_2; b_2]$ . Напишите программу, которая находит их пересечение.

Пересечением двух отрезков может быть:

- отрезок;
- точка;
- пустое множество.

На вход программе подаются 4 целых числа  $a_1, b_1, a_2, b_2$ , каждое на отдельной строке. Гарантируется, что  $a_1 < b_1$  и  $a_2 < b_2$ . Программа должна вывести на экран границы отрезка, являющегося пересечением, либо общую точку, либо текст «пустое множество».

```
a1 = int(input())
b1 = int(input())
a2 = int(input())
b2 = int(input())
if a2 > b1:
    print('пустое множество')
elif a1 > b2:
    print('пустое множество')
elif a2 == b1:
    print(a2)
elif a1 == b2:
    print(a1)
elif a1 <= a2:
    if b1 <= b2:
        print(a2, b1)
    else:
        if a1 < a2:
            if b1 > b2:
                print(a2, b2)
elif a2 <= a1:
    if b2 <= b1:
        print(a1, b2)
    else:
        if a2 < a1:
            if b2 > b1:
                print(a1, b1)
```

61. Напишите программу, которая определяет, оканчивается ли год с данным номером на два нуля. Если год оканчивается, то выведите «YES», иначе выведите «NO».

На вход программе подаётся натуральное число. Программа должна вывести текст в соответствии с условием задачи

```
year = int(input())
year4 = year % 10
year3 = (year % 100) // 10
if year4 == year3 == 0:
    print("YES")
else:
    print("NO")
```

62. Заданы две клетки шахматной доски. Напишите программу, которая определяет имеют ли указанные клетки один цвет или нет. Если они покрашены в один цвет, то выведите слово «YES», а если в разные цвета - то «NO».

На вход программе подаётся четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести текст в соответствии с условием задачи.

```
a1 = int(input())
b1 = int(input())
a2 = int(input())
b2 = int(input())
if (a1 + a2 + b1 + b2) % 2 == 0:
    print('YES')
else:
    print('NO')
```

63. Футбольная команда набирает девочек от 10 до 15 лет включительно. Напишите программу, которая запрашивает возраст и пол претендента, используя обозначение пола буквы m (от male – мужчина) и f (от female – женщина) и определяет подходит ли претендент для вступления в команду или нет. Если претендент подходит, то выведите «YES», иначе выведите «NO».

На вход программе подаётся натуральное число – возраст претендента и буква обозначающая пол m (мужчина) или f (женщина). Программа должна вывести текст в соответствии с условием задачи.

```
age = int(input())
male = input()
if male == 'f' and 10 <= age <= 15:
    print("YES")
else:
    print("NO")
```

64. Напишите программу, которая считывает целое число и выводит соответствующую ему римскую цифру. Если число находится вне диапазона 1-10, то программа должна вывести текст «ошибка».

```
x = int(input())
if x == 1:
    print('I')
elif x == 2:
    print('II')
```

```
elif x == 3:
    print('III')
elif x == 4:
    print('IV')
elif x == 5:
    print('V')
elif x == 6:
    print('VI')
elif x == 7:
    print('VII')
elif x == 8:
    print('VIII')
elif x == 9:
    print('IX')
elif x == 10:
    print('X')
else:
    print('ошибка')
```

65. Напишите программу, которая принимает на вход число и в зависимости от условий выводит текст «YES», либо «NO».

- Условия:
- если число нечётное, то вывести «YES»;
  - если число чётное в диапазоне от 2 до 5 (включительно), то вывести «NO»;
  - если число чётное в диапазоне от 6 до 20 (включительно), то вывести «YES»;
  - если число чётное и больше 20, то вывести «NO».

```
x = int(input())
if x % 2 != 0:
    print('YES')
elif x % 2 == 0 and 2 <= x <= 5:
    print('NO')
elif x % 2 == 0 and 6 <= x <= 20:
    print('YES')
elif x % 2 == 0 and x > 20:
    print('NO')
```

66. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли слон попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом слона можно попасть во вторую или «NO» в противном случае.

На вход программе подаётся четыре числа от 1 до 8. Программа должна вывести текст в соответствии с условием задачи. Шахматный слон ходит по диагоналям.

```
a, b, c, d = int(input()), int(input()), int(input()), int(input())
if (a - c)**2 == (b - d)**2:
    print('YES')
else:
    print('NO')
```

67. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли конь попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом коня можно попасть во вторую или «NO» в противном случае.

На вход программе подаётся четыре числа от 1 до 8. Программа должна вывести текст в соответствии с условием задачи. Шахматный конь ходит буквой «Г».

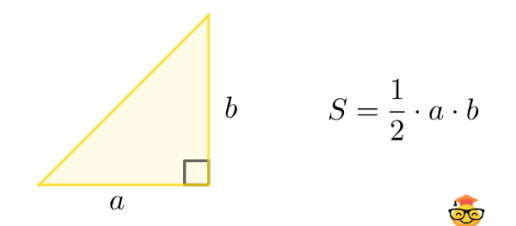
```
a, b, c, d = int(input()), int(input()), int(input()), int(input())
if (a + b + c + d) % 2 == 1 and a != c and b != d:
    print('YES')
else:
    print('NO')
```

68. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли ферзь попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом ферзя можно попасть во вторую или «NO» в противном случае.

На вход программе подаётся четыре числа от 1 до 8. Программа должна вывести текст в соответствии с условием задачи. Шахматный ферзь ходит по диагонали, горизонтали или вертикали.

```
a, b, c, d = int(input()), int(input()), int(input()), int(input())
if (a != c and b == d) or (a == c and b != d) or (a - c)**2 == (b - d)**2:
    print('YES')
else:
    print('NO')
```

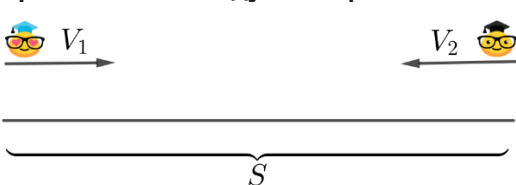
69. Напишите программу, которая считывает длины двух катетов в прямоугольном треугольнике и выводит его площадь.



На вход программе подаётся два числа с плавающей точкой – длины катетов, каждое на отдельной строке. Программа должна вывести одно число – площадь треугольника.

```
a = float(input())
b = float(input())
print(0.5 * a * b)
```

70. Две старушки идут навстречу друг другу с постоянными скоростями  $V_1$  и  $V_2$  км/ч. Определите, через какое время старушки встретятся, если расстояние между ними равно  $S$  км.



На вход программе подаются три числа с плавающей точкой  $S, V_1, V_2$ , каждое на отдельной строке. Программа должна вывести одно число в соответствии с условием задачи.

```
s = float(input())
v1 = float(input())
v2 = float(input())
print(s / (v1 + v2))
```

71. Напишите программу, которая считывает с клавиатуры одно число и выводит обратное ему. Если при этом введённое с клавиатуры число – ноль, то вывести «Обратного числа не существует» (без кавычек).

$$\left(\text{👓}\right)^{-1} = \frac{1}{\text{👓}}$$

На вход программе подается одно действительное число. Программа должна вывести действительное число обратное данному, либо текст в соответствии с условием задачи.

```
x = float(input())
if x == 0:
    print('Обратного числа не существует')
else:
    print(1 / x)
```



72. У известного американского писателя Рэя Бредбери есть роман «451 градус по Фаренгейту». Напишите программу, которая определяет, какой температуре по шкале Цельсия соответствует указанное значение по шкале Фаренгейта.  
На вход программе подаётся вещественное число градусов по шкале Фаренгейта. Программа должна вывести число градусов по шкале Цельсия.

```
f = float(input())
print(5 / 9 * (f - 32))
```

73. На вход программе подается число  $n$  – количество собачьих лет. Напишите программу, которая вычисляет возраст собаки в человеческих годах.  
На вход программе подаётся натуральное число – количество собачьих лет. Программа должна вывести возраст собаки в человеческих годах. В течение первых двух лет собачий год равен 10.5 человеческим годам. После этого каждый год собаки равен 4 человеческим годам.

```
age = float(input())
if 1 <= age <= 2:
    print(age * 10.5)
else:
    print(21 + (age - 2) * 4)
```

74. Дано положительное действительное число. Выведите его первую цифру после десятичной точки.  
На вход программе подается положительное действительное число. Программа должна вывести цифру в соответствии с условием задачи.

```
x = float(input())
y = int(x * 10) % 10
print(y)
```

75. Дано положительное действительное число. Выведите его дробную часть.  
На вход программе подается положительное действительное число. Программа должна вывести дробную часть числа в соответствии с условием задачи.

```
x = float(input())
y = x - int(x)
print(y)
```

76. Напишите программу, которая находит наименьшее и наибольшее из пяти чисел.  
На вход программе подается пять целых чисел, каждое на отдельной строке. Программа должна вывести наименьшее и наибольшее число с поясняющей надписью.

```
a = int(input())
b = int(input())
c = int(input())
d = int(input())
e = int(input())
print('Наименьшее число =', min(a, b, c, d, e))
print('Наибольшее число =', max(a, b, c, d, e))
```

77. Напишите программу, которая упорядочивает три числа от большего к меньшему.  
На вход программе подается три целых числа, каждое на отдельной строке. Программа должна вывести три числа, каждое на отдельной строке, упорядоченных от большего к меньшему.

```
a, b, c = int(input()), int(input()), int(input())
x = max(a, b, c)
y = min(a, b, c)
z = a + b + c - x - y
print(x)
print(z)
print(y)
```

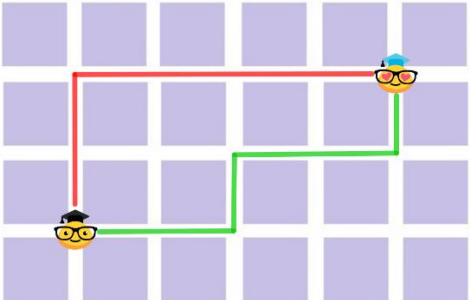
78. Назовем число интересным, если в нем разность максимальной и минимальной цифры равняется средней по величине цифре. Напишите программу, которая определяет интересное число или нет. Если число интересное, следует вывести – «Число интересное» иначе «Число неинтересное».  
На вход программе подается целое трехзначное число. Программа должна вывести текст в соответствии с условием задачи.

```
x = int(input())
x3 = x % 10
x2 = x % 100 // 10
x1 = x // 100
if max(x1, x2, x3) - min(x1, x2, x3) == x1 + x2 + x3 - max(x1, x2, x3) - min(x1, x2, x3):
    print('Число интересное')
else:
    print('Число неинтересное')
```

79. Даны пять чисел  $a_1, a_2, a_3, a_4, a_5$ . Напишите программу, которая вычисляет сумму их модулей  $|a_1| + |a_2| + |a_3| + |a_4| + |a_5|$ .  
На вход программе подается пять действительных чисел  $a_1, a_2, a_3, a_4, a_5$ , каждое на отдельной строке. Программа должна вывести одно число – сумму модулей введенных чисел.

```
a1 = float(input())
a2 = float(input())
a3 = float(input())
a4 = float(input())
a5 = float(input())
b1 = abs(a1)
b2 = abs(a2)
b3 = abs(a3)
b4 = abs(a4)
b5 = abs(a5)
print(b1 + b2 + b3 + b4 + b5)
```

80. Прогуливаясь по Манхэттену, вы не можете попасть из точки А в точку Б по кратчайшему пути. Если только вы не умеете проходить сквозь стены, вам обязательно придется идти вдоль его параллельно перпендикулярных улиц. На плоскости манхэттенское расстояние между двумя точками  $(p_1; p_2)$  и  $(q_1; q_2)$  определяется так  $|p_1 - q_1| + |p_2 - q_2|$ .  
Напишите программу определяющую манхэттенское расстояние между двумя точками, координаты которых заданы.



На вход программе подается четыре целых числа, каждое на отдельной строке –  $p_1, p_2, q_1, q_2$ . Программа должна вывести одно число – манхэттенское расстояние.

```
p1 = int(input())
p2 = int(input())
q1 = int(input())
q2 = int(input())
print(abs(p1 - q1) + abs(p2 - q2))
```

81. Напишите программу, которая выводит текст:

```
"Python is a great language!", said Fred. "I don't ever remember having this much fun before."
```

Используйте конкатенацию строк.

```
print("Python is a great language!", said Fred. "I don' + "t" + ' ever remember having this much fun before.")
```



82. Напишите программу, которая считывает с клавиатуры две строки – имя и фамилию пользователя и выводит фразу:

«Hello [введенное имя] [введенная фамилия]! You just delved into Python».

На вход программе подаётся две строки (имя и фамилия), каждая на отдельной строке. Программа должна вывести текст в соответствии с условием задачи.

```
firstname = input()
lastname = input()
print('Hello', firstname, lastname + '! You just delved into Python')
```
83. Напишите программу, которая считывает с клавиатуры название футбольной команды и выводит фразу:

«Футбольная команда [введённая строка] имеет длину [длина введённой строки] символов».

На вход программе подаётся строка – название футбольной команды. Программа должна вывести текст в соответствии с условием задачи.

```
team = input()
print('Футбольная команда', team, 'имеет длину', len(team), 'символов')
```
84. Даны названия трех городов. Напишите программу, которая определяет самое короткое и самое длинное название города.

На вход программе подаётся названия трех городов, каждое на отдельной строке. Программа должна вывести самое короткое и длинное название города, каждое на отдельной строке. Гарантируется, что длины названий всех трех городов различны.

```
city1 = input()
city2 = input()
city3 = input()
if len(city1) < len(city2) < len(city3):
    print(city1, city3, sep='\n')
elif len(city1) < len(city3) < len(city2):
    print(city1, city2, sep='\n')
elif len(city2) < len(city1) < len(city3):
    print(city2, city3, sep='\n')
elif len(city2) < len(city3) < len(city1):
    print(city2, city1, sep='\n')
elif len(city3) < len(city1) < len(city2):
    print(city3, city2, sep='\n')
elif len(city3) < len(city1) < len(city2):
    print(city3, city2, sep='\n')
```
85. Вводятся 3 строки в случайном порядке. Напишите программу, которая выясняет можно ли из длин этих строк построить возрастающую арифметическую прогрессию.

На вход программе подаются три строки, каждая на отдельной строке. Программа должна вывести строку «YES», если из длин введенных слов можно построить арифметическую прогрессию, «NO» в ином случае.

```
a = len(input())
b = len(input())
c = len(input())
x = max(a, b, c)
y = min(a, b, c)
z = a + b + c - x - y
if z == (x + y) / 2:
    print('YES')
else:
    print('NO')
```
86. Напишите программу, которая считывает одну строку, после чего выводит «YES», если в введенной строке есть подстрока «синий» и «NO» в противном случае.

На вход программе подается одна строка. Программа должна вывести текст в соответствии с условием задачи.

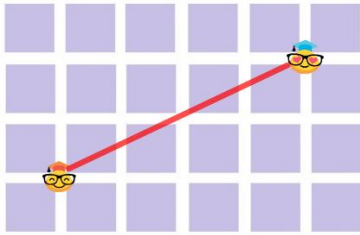
```
s = input()
if 'синий' in s:
    print('YES')
else:
    print('NO')
```
87. Напишите программу, которая считывает одну строку, после чего выводит «YES», если в введённой строке есть подстрока «суббота» или «воскресенье», и «NO» в противном случае.

На вход программе подается одна строка. Программа должна вывести текст в соответствии с условием задачи.

```
s = input()
if 'суббота' in s or 'воскресенье' in s:
    print('YES')
else:
    print('NO')
```
88. Будем считать email адрес корректным, если в нем есть символ собачки (@) и точки. Напишите программу проверяющую корректность email адреса.

На вход программе подаётся одна строка – email адрес. Программа должна вывести строку «YES», если email адрес является корректным и «NO» в ином случае.

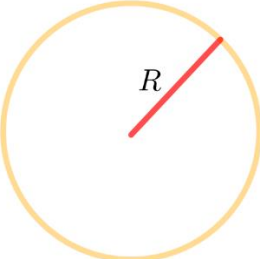
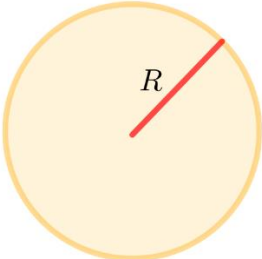
```
s = input()
if '@' in s and '.' in s:
    print('YES')
else:
    print('NO')
```
89. На плоскости евклидово расстояние между двумя точками (x1;y1) и (x2;y2) определяется так  $p=\sqrt{(x1-x2)^2+(y1-y2)^2}$ .




Напишите программу определяющую евклидово расстояние между двумя точками, координаты которых заданы. На вход программе подается четыре вещественных числа, каждое на отдельной строке – x1,y1,x2 ,y2. Программа должна вывести одно число – евклидово расстояние.

```
from math import *
x1 = float(input())
y1 = float(input())
x2 = float(input())
y2 = float(input())
p = sqrt((x1 - x2) ** 2 + (y1 - y2) ** 2)
print(p)
```
90. Напишите программу определяющую площадь круга и длину окружности по заданному радиусу R.

На вход программе подается одно вещественное число R. Программа должна вывести два числа – площадь круга и длину окружности радиуса R.



$$S = \pi R^2$$
$$C = 2\pi R$$


```
from math import *
r = float(input())
print(pi * pow(r, 2))
print(2 * pi * r)
```

91. В математике выделяют следующие средние значения:

- среднее арифметическое чисел  $a$  и  $b$ ;
- среднее геометрическое чисел  $a$  и  $b$ ;
- среднее гармоническое чисел  $a$  и  $b$ ;
- среднее квадратичное чисел  $a$  и  $b$ .

На вход программе подается два вещественных числа  $a$  и  $b$ , каждое на отдельной строке. Программа должна вывести 4 числа – среднее арифметическое, геометрическое, гармоническое и квадратичное.

```
from math import *
a = float(input())
b = float(input())
print((a + b) / 2)
print(sqrt(a * b))
print(2 * a * b / (a + b))
print(sqrt((a ** 2 + b ** 2) / 2))
```

92. Напишите программу, вычисляющую значение тригонометрического выражения  $\sin x + \cos x + \tan 2x$  по заданному числу градусов  $x$ .

На вход программе подается одно вещественное число  $x$  измеряемое в градусах. Программа должна вывести одно число – значение тригонометрического выражения. Тригонометрические функции принимают аргумент в радианах. Модуль `math` содержит встроенную функцию `radians()`, которая переводит угол из градусов в угол в радианах.

```
from math import *
x = float(input())
print(sin(radians(x)) + cos(radians(x)) + tan(radians(x)) ** 2)
```

93. Напишите программу, вычисляющую значение  $\lceil x \rceil + \lfloor x \rfloor$  по заданному вещественному числу  $x$ .

На вход программе подается одно вещественное число  $x$ . Программа должна вывести одно число – значение указанного выражения.

```
from math import *
x = float(input())
print(ceil(x) + floor(x))
```

94. Даны три вещественных числа  $a$ ,  $b$ ,  $c$ . Напишите программу, которая находит вещественные корни квадратного уравнения  $ax^2 + bx + c = 0$ .

На вход программе подается три вещественных числа, каждое на отдельной строке. Программа должна вывести вещественные корни уравнения если они существуют или текст «Нет корней» в противном случае. Если уравнение имеет два корня, то следует вывести их в порядке возрастания.

```
from math import *
a = float(input())
b = float(input())
c = float(input())
d = b ** 2 - 4 * a * c
if d == 0:
    print(-b / (2 * a))
elif d > 0:
    x1 = (-b - sqrt(d)) / (2 * a)
    x2 = (-b + sqrt(d)) / (2 * a)
    print(min(x1, x2))
    print(max(x1, x2))
elif d < 0:
    print('Нет корней')
```

95. Правильный **многоугольник** — выпуклый многоугольник, у которого равны все стороны и все углы между смежными сторонами. Даны два числа: натуральное число  $n$  и вещественное число  $a$ . Напишите программу, которая находит площадь указанного правильного многоугольника.

На вход программе подается два числа  $n$  и  $a$ , каждое на отдельной строке. Программа должна вывести вещественное число – площадь многоугольника

```
from math import *
n = float(input())
a = float(input())
s = n * (a ** 2) / (4 * tan(pi / n))
print(s)
```

96. Напишите программу, которая выводит слова «Python is awesome!» (без кавычек) 10 раз.

Программа должна вывести 10 раз текст «Python is awesome!», каждый на отдельной строке.

```
for i in range(10):
    print('Python is awesome!')
```

97. Дано предложение и количество раз которое его надо повторить. Напишите программу, которая повторяет данное предложение нужное количество раз.

В первой строке записано текстовое предложение, во второй — количество повторений. Программа должна вывести указанное текстовое предложение нужное количество раз. Каждое повторение должно начинаться с новой строки.

```
a = input()
b = int(input())
for i in range(b):
    print(a)
```

98. Напишите программу, которая использует ровно три цикла `for` для печати следующей последовательности символов.

```
for i in range(6):
    print('AAA')
for i in range(5):
    print('BBBB')
print('E')
for i in range(9):
    print('TTTTT')
print('G')
```

99. На вход программе подается натуральное число  $n$ . Напишите программу, которая печатает звездный прямоугольник размерами  $n \times 19$ .

На вход программе подаётся натуральное число  $n \in [1; 20]$  — высота звездного прямоугольника. Программа должна вывести звездный прямоугольник размерами  $n \times 19$ .

```
n = int(input())
for i in range(n):
    print('*****')
```

100. Напишите программу, которая считывает одну строку текста и выводит 10 строк, пронумерованных от 0 до 9, каждая с указанной строкой текста.

На вход программе подается одна строка текста. Программа должна вывести десять строк в соответствии с условием задачи.

```
x = input()
for i in range(10):
    print(i, x)
```

101.

На вход программе подается натуральное число  $n$ . Напишите программу, которая для каждого из чисел от 00 до  $n$  (включительно) выводит фразу: «Квадрат числа [число] равен [число]» (без кавычек).  
На вход программе подается натуральное число  $n$ . Программа должна вывести текст в соответствии с условием задачи.

```
x = int(input())
for i in range(x + 1):
    print('Квадрат числа', i, 'равен', i ** 2)
```
102.

На вход программе подается натуральное число  $n(n \geq 2)$  – катет прямоугольного равнобедренного треугольника.  
Напишите программу, которая выводит звездный треугольник в соответствии с примером. На вход программе подается одно натуральное число  $n(n \geq 2)$ .

```
n = int(input())
for i in range(n):
    print('*' * (n - i))
```
103.

На вход программе подается три натуральных числа  $p, n$ :

  - $m$ :  $m$ : стартовое количество организмов;
  - $p$ :  $p$ : среднесуточное увеличение в %;
  - $n$ :  $n$ : количество дней для размножения.

Напишите программу, которая предсказывает размер популяции организмов. Программа должна выводить размер популяции в каждый день, начиная с 11 и заканчивая  $n$ -м днем. На вход программе подается три натуральных числа.

```
m = int(input())
p = int(input())
n = int(input())
for i in range(n):
    print(i + 1, m)
    m = m + m * p / 100
```
104.

Даны два целых числа  $m$  и  $n (m \leq n)$ . Напишите программу, которая выводит все числа от  $m$  до  $n$  включительно.

```
m = int(input())
n = int(input())
for i in range(m, n + 1):
    print(i)
```
105.

Даны два целых числа  $m$  и  $n$ . Напишите программу, которая выводит все числа от  $m$  до  $n$  включительно в порядке возрастания, если  $m < n$ , или в порядке убывания в противном случае.

```
m = int(input())
n = int(input())
if m > n:
    for i in range(m, n - 1, -1):
        print(i)
elif m < n:
    for i in range(m, n + 1):
        print(i)
elif m == n:
    print(m)
```
106.

Даны два целых числа  $m$  и  $n (m > n)$ . Напишите программу, которая выводит все нечетные числа от  $m$  до  $n$  включительно в порядке убывания.

```
m = int(input())
n = int(input())
for i in range(m, n - 1, -1):
    if i % 2 != 0:
        print(i)
```
107.

Даны два натуральных числа  $m$  и  $n (m \leq n)$ . Напишите программу, которая выводит все числа от  $m$  до  $n$  включительно удовлетворяющие хотя бы одному из условий:

  - число кратно 17;
  - число оканчивается на 9;
  - число кратно 3 и 5 одновременно.

На вход программе подаются два натуральных числа  $m$  и  $n (m \leq n)$ , каждое на отдельной строке. Программа должна вывести числа в соответствии с условием задачи. Если чисел удовлетворяющих условию нет, выводить ничего не надо.

```
m = int(input())
n = int(input())
for i in range(m, n + 1):
    if (i % 17 == 0) or (i % 10 == 9) or (i % 3 == 0 and i % 5 == 0):
        print(i)
```
108.

Дано натуральное число  $n$ . Напишите программу, которая выводит таблицу умножения на  $n$ .  
На вход программе подается натуральное число. Программа должна вывести таблицу умножения на введенное число.

```
n = int(input())
for i in range(10):
    print(n, 'x', i + 1, '=', n * (i + 1))
```
109.

На вход программе подаются два целых числа  $a$  и  $b (a \leq b)$ . Напишите программу, которая подсчитывает количество чисел в диапазоне от  $a$  до  $b$  включительно, куб которых оканчивается на 44 или 99.

```
a = int(input())
b = int(input())
counter = 0
for i in range(a, b + 1):
    if i ** 3 % 10 == 4 or i ** 3 % 10 == 9:
        counter += 1
print(counter)
```
110.

На вход программе подается натуральное число  $n$ , а затем  $n$  целых чисел, каждое на отдельной строке. Напишите программу, которая подсчитывает сумму введенных чисел.  
На вход программе подаются натуральное число  $n$ , а затем  $n$  целых чисел, каждое на отдельной строке. Программа должна вывести сумму данных чисел.

```
total = 0
n = int(input())
for i in range(n):
    num = int(input())
    total = total + num
print(total)
```
111.

На вход программе подается натуральное число  $n$ . Напишите программу, которая вычисляет значение выражения  $(1 + 1/2 + 1/1 + \dots + 1/n) - \ln(n)$ .  
Для вычисления натурального логарифма воспользуйтесь функцией  $\log(n)$ , которая находится в модуле `math`.

```
from math import *
total = 0
n = int(input())
for i in range(1, n + 1):
    total = total + 1 / i
print(total - log(n))
```
112.

На вход программе подается натуральное число  $n$ . Напишите программу, которая подсчитывает сумму тех чисел от 1 до  $n$  (включительно) квадрат которых оканчивается на 2 5 или 8.  
На вход программе подается натуральное число  $n$ . Программа должна вывести единственное число в соответствии с условием задачи. Если таких чисел нет в указанном диапазоне, то следует вывести 0.

```
total = 0
n = int(input())
for i in range(1, n + 1):
```

```
if i ** 2 % 10 == 2 or i ** 2 % 10 == 5 or i ** 2 % 10 == 8:
    total = total + i
print(total)
```

- 113. На вход программе подается натуральное число  $n$ . Напишите программу, которая вычисляет  $n!$ . На вход программе подается натуральное число  $n$ , ( $n \leq 12$ ). Программа должна вывести единственное число в соответствии с условием задачи. Факториалом натурального числа  $n$ , называется произведение всех натуральных чисел от 1 до  $n$ , то есть  $n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$**

```
total = 1
n = int(input())
for i in range(n):
    total = total * (i + 1)
print(total)
```

- 114. Напишите программу, которая считывает 10 чисел и выводит произведение отличных от нуля чисел. На вход программе подаются 10 целых чисел, каждое на отдельной строке. Программа должна вывести произведение отличных от нуля чисел. Гарантируется, что хотя бы одно из 10 чисел является ненулевым.**

```
total = 1
for i in range(1, 11):
    num = int(input())
    if num != 0:
        total = total * num
print(total)
```

- 115. На вход программе подается натуральное число  $n$ . Напишите программу, которая вычисляет сумму всех его делителей.**

```
total = 0
n = int(input())
for i in range(1, n + 1):
    if n % i == 0:
        total += i
print(total)
```

- 116. На вход программе подается натуральное число  $n$ . Напишите программу вычисления знакопеременной суммы  $1 - 2 + 3 - 4 + 5 - 6 + \dots + (-1)^{n+1}n$ .**

```
total = 0
n = int(input())
for i in range(1, n + 1):
    total = total + (-1) ** (i + 1) * i
print(total)
```

- 117. На вход программе подается натуральное число  $n$ , а затем  $n$  различных натуральных чисел, каждое на отдельной строке. Напишите программу, которая выводит наибольшее и второе наибольшее число последовательности. На вход программе подаются натуральное число  $n \geq 2$ , а затем  $n$  различных натуральных чисел, каждое на отдельной строке. Программа должна вывести два наибольших числа, каждое на отдельной строке.**

```
largest1 = 0
largest2 = 0
n = int(input())
for i in range(1, n + 1):
    num = int(input())
    if num > largest1:
        largest2 = largest1
        largest1 = num
    elif num > largest2:
        largest2 = num
print(largest1)
print(largest2)
```

- 118. Напишите программу, которая считывает последовательность из 10 целых чисел и определяет является ли каждое из них четным или нет. На вход программе подаются 10 целых чисел, каждое на отдельной строке. Программа должна вывести строку «YES», если все числа четные и «NO» в ином случае.**

```
counter = 0
for i in range(1, 11):
    num = int(input())
    if num % 2 == 0:
        counter = counter + 1
if counter == 10:
    print('YES')
else:
    print('NO')
```

- 119. Напишите программу, которая считывает натуральное число  $n$  и выводит первые  $n$  чисел последовательности Фибоначчи. На вход программе подается одно число  $n$  ( $n \leq 100$ ) – количество членов последовательности. Программа должна вывести члены последовательности Фибоначчи, отделенные символом пробела. Последовательность Фибоначчи – это последовательность натуральных чисел, где каждое последующее число является суммой двух предыдущих.**

```
f1 = 0
f2 = 1
n = int(input())
for i in range(1, n + 1):
    f1, f2 = f2, f1 + f2
    print(f1, end=' ')
```

- 120. На вход программе подается последовательность слов, каждое слово на отдельной строке. Концом последовательности является слово «КОНЕЦ» (без кавычек). Напишите программу, которая выводит члены данной последовательности. На вход программе подается последовательность слов, каждое слово на отдельной строке. Программа должна вывести члены данной последовательности.**

```
word = input()
while word != 'КОНЕЦ':
    print(word)
    word = input()
```

- 121. На вход программе подается последовательность слов, каждое слово на отдельной строке. Концом последовательности является слово «КОНЕЦ» или «конец» (большими или маленькими буквами, без кавычек). Напишите программу, которая выводит члены данной последовательности. На вход программе подается последовательность слов, каждое слово на отдельной строке. Программа должна вывести члены данной последовательности.**

```
word = input()
while word != 'КОНЕЦ' and word != 'конец':
    print(word)
    word = input()
```

- 122. На вход программе подается последовательность слов, каждое слово на отдельной строке. Концом последовательности является одно из трех слов: «стоп», «хватит», «достаточно» (маленькими буквами, без кавычек). Напишите программу, которая выводит общее количество членов данной последовательности. На вход программе подается последовательность слов, каждое слово на отдельной строке. Программа должна вывести общее количество членов данной последовательности.**

```
x = input()
total = 0
while x != 'стоп' and x != 'хватит' and x != 'достаточно':
    total += 1
    x = input()
print(total)
```

**123. На вход программе подается последовательность целых чисел делящихся на 77, каждое число на отдельной строке. Концом последовательности является любое число не делящееся на 77. Напишите программу, которая выводит члены данной последовательности.**  
**На вход программе подается последовательность чисел, каждое число на отдельной строке. Программа должна вывести члены данной последовательности.**

```
x = int(input())
while x % 77 == 0:
    print(x)
    x = int(input())
```

**124. На вход программе подается последовательность целых чисел, каждое число на отдельной строке. Концом последовательности является любое отрицательное число. Напишите программу, которая выводит сумму всех членов данной последовательности.**  
**На вход программе подается последовательность чисел, каждое число на отдельной строке. Программа должна вывести сумму членов данной последовательности.**

```
x = int(input())
total = 0
while x >= 0:
    total += x
    x = int(input())
print(total)
```

**125. На вход программе подается последовательность целых чисел от 11 до 55, характеризующее оценку ученика, каждое число на отдельной строке. Концом последовательности является любое отрицательное число, либо число большее 55. Напишите программу, которая выводит количество пятерок.**  
**На вход программе подается последовательность чисел, каждое число на отдельной строке. Программа должна вывести количество пятерок.**

```
x = int(input())
counter = 0
while 0 <= x <= 55:
    if x == 5:
        counter += 1
    x = int(input())
print(counter)
```

**126. Всем известно, что ведьмак способен одолеть любых чудовищ, однако его услуги обойдутся недешево, к тому же ведьмак не принимает купюры, он принимает только чеканные монеты. В мире ведьмака существуют монеты с номиналами 1,5,10,25.**  
**Напишите программу, которая определяет какое минимальное количество чеканных монет нужно заплатить ведьмаку.**  
**На вход программе подается одно натуральное число, цена за услугу ведьмака. Программа должна вывести минимально возможное количество чеканных монет для оплаты.**

```
x = int(input())
counter = 0
while x >= 25:
    counter += 1
    x = x - 25
while x >= 10:
    counter += 1
    x = x - 10
while x >= 5:
    counter += 1
    x = x - 5
while x >= 1:
    counter += 1
    x = x - 1
print(counter)
```

**127. Дано натуральное число. Напишите программу, которая выводит его цифры в столбик в обратном порядке.**  
**На вход программе подается одно натуральное число.**  
**Программа должна вывести цифры введенного числа в столбик в обратном порядке.**

```
n = int(input())
while n != 0:
    last_digit = n % 10
    n = n // 10
    print(last_digit)
```

**128. Дано натуральное число. Напишите программу, которая меняет порядок цифр числа на обратный.**  
**На вход программе подается одно натуральное число.**  
**Программа должна вывести число, записанное в обратном порядке.**

```
n = int(input())
while n != 0:
    last_digit = n % 10
    n = n // 10
    print(last_digit, end="")
```

**129. Дано натуральное число  $n$ , ( $n \geq 10$ ). Напишите программу, которая определяет его максимальную и минимальную цифры.**  
**На вход программе подается одно натуральное число.**  
**Программа должна вывести максимальную и минимальную цифры введенного числа (с поясняющей надписью).**

```
n = int(input())
biggest = 0
smallest = 9
while n != 0:
    last_digit = n % 10
    if last_digit > biggest:
        biggest = last_digit
    if last_digit < smallest:
        smallest = last_digit
    n = n // 10
print('Максимальная цифра равна', biggest)
print('Минимальная цифра равна', smallest)
```

**130. Дано натуральное число. Напишите программу, которая вычисляет:**

- сумму его цифр;
- количество цифр в нем;
- произведение его цифр;
- среднее арифметическое его цифр;
- его первую цифру;
- сумму его первой и последней цифры.

**На вход программе подается одно натуральное число.**  
**Программа должна вывести значения указанных величин в указанном порядке.**

```
n = int(input())
summa = 0
counter = 0
proizv = 1
sred = 0
first = n % 10
while n != 0:
    last_digit = n % 10
    summa += last_digit
    counter += 1
    proizv *= last_digit
    n = n // 10
sred = summa / counter
print(summa)
print(counter)
print(proizv)
```

```
print(sred)
print(last_digit)
print(first + last_digit)
```

**131. Дано натуральное число  $n(n > 9)$ . Напишите программу, которая определяет его вторую (с начала) цифру. На вход программе подается одно натуральное число, состоящее как минимум из двух цифр. Программа должна вывести его вторую (с начала) цифру.**

```
n = int(input())
while n > 9:
    last_digit = n % 10
    n = n // 10
print(last_digit)
```

**132. Дано натуральное число. Напишите программу, которая определяет, состоит ли указанное число из одинаковых цифр. На вход программе подается одно натуральное число. Программа должна вывести «YES» если число состоит из одинаковых цифр и «NO» в противном случае.**

```
n = int(input())
counter1 = 0
counter2 = 0
last = n % 10
while n != 0:
    last_digit = n % 10
    counter1 += 1
    if last == last_digit:
        counter2 += 1
    n = n // 10
if counter1 == counter2:
    print('YES')
else:
    print('NO')
```

**133. Дано натуральное число. Напишите программу, которая определяет, является ли последовательность его цифр при просмотре справа налево упорядоченной по неубыванию. На вход программе подается одно натуральное число. Программа должна вывести «YES» если последовательность его цифр при просмотре справа налево является упорядоченной по неубыванию и «NO» в противном случае.**

```
n = int(input())
counter1 = 0
counter2 = 0
last = n % 10
while n != 0:
    last_digit = n % 10
    counter1 += 1
    if last <= last_digit:
        last = last_digit
        counter2 += 1
    n = n // 10
if counter1 == counter2:
    print('YES')
else:
    print('NO')
```

**134. На вход программе подается число  $n > 1$ . Напишите программу, которая выводит его наименьший отличный от 11 делитель. На вход программе подается одно натуральное число  $n$ . Программа должна вывести наименьший делитель отличный от 11. Используйте оператор `break` при обнаружении делителя.**

```
n = int(input())
flag = False
for i in range(1, n+1):
    if n % i == 0 and i != 1:
        flag = True
        break
if flag == True:
    print(i)
```

**135. На вход программе подается натуральное число  $n$ . Напишите программу, которая выводит числа от 11 до  $n$  включительно за исключением:**

- чисел от 55 до 99 включительно;
- чисел от 1717 до 3737 включительно;
- чисел от 7878 до 8787 включительно.

**На вход программе подается одно натуральное число  $n$ . Программа должна вывести числа в соответствии с условием задачи, каждое на отдельной строке. Используйте оператор `continue`.**

```
n = int(input())
for i in range(1, n + 1):
    if 5 <= i <= 9 and 17 <= i <= 37 and 78 <= i <= 87:
        continue
    print(i)
```

**136. На обработку поступает последовательность из 10 целых чисел. Известно, что вводимые числа по абсолютной величине не превышают  $10^{**6}$ . Нужно написать программу, которая выводит на экран количество неотрицательных чисел последовательности и их произведение. Если неотрицательных чисел нет, требуется вывести на экран «NO». Программист торопился и написал программу неправильно. Найдите все ошибки в этой программе (их ровно 4). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Число  $x$  не превышает по абсолютной величине  $10^{**6}$ , если  $-10^{**6} \leq x \leq 10^{**6}$ . При необходимости вы можете добавить необходимые строки кода.**

```
count = 0
p = 1
for i in range(1, 10 + 1):
    x = int(input())
    if x >= 0:
        p = p * x
        count = count + 1
if count > 0:
    print(count)
    print(p)
else:
    print('NO')
```

**137. На обработку поступает последовательность из 10 целых чисел. Известно, что вводимые числа по абсолютной величине не превышают  $10^{**6}$ . Нужно написать программу, которая выводит на экран сумму всех отрицательных чисел последовательности и максимальное отрицательное число в последовательности. Если отрицательных чисел нет, требуется вывести на экран «NO». Программист торопился и написал программу неправильно. Найдите все ошибки в этой программе (их ровно 5). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Число  $x$  не превышает по абсолютной величине  $10^{**6}$ , если  $-10^{**6} \leq x \leq 10^{**6}$ . При необходимости вы можете добавить необходимые строки кода.**

```
mx = -10**6 - 1
s = 0
for i in range(1, 11):
    x = int(input())
    if x < 0:
        s = s + x
    if x > mx and x < 0:
        mx = x
if s < 0:
    print(s)
```



```
print(mx)
else:
    print('NO')
```

138. На обработку поступает последовательность из 7 целых чисел. Известно, что вводимые числа по абсолютной величине не превышают  $10^{**6}$ . Нужно написать программу, которая подсчитывает и выводит сумму всех чётных чисел последовательности или 00, если чётных чисел в последовательности нет. Программист торопился и написал программу неправильно.  
Найдите все ошибки в этой программе (их ровно 4). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.  
Число  $x$  не превышает по абсолютной величине  $10^{**6}$ , если  $-10^{**6} \leq x \leq 10^{**6}$ .  
При необходимости вы можете добавить необходимые строки кода.

```
s = 0
for i in range(1, 8):
    n = int(input())
    if n % 2 == 0:
        s = s + n
if s > 0:
    print(s)
else:
    print(0)
```

139. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран максимальную цифру числа, кратную 3. Если в числе нет цифр, кратных 3, требуется на экран вывести «NO». Программист торопился и написал программу неправильно.  
Найдите все ошибки в этой программе (их ровно 5). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.  
Число 0 делится на любое натуральное число.  
При необходимости вы можете добавить нужные строки кода.

```
n = int(input())
max_digit = -1
while n > 0:
    digit = n % 10
    if digit % 3 == 0:
        if digit > max_digit:
            max_digit = digit
    n = n // 10
if max_digit < 0:
    print('NO')
else:
    print(max_digit)
```

140. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран его первую (старшую) цифру. Программист торопился и написал программу неправильно.  
Найдите все ошибки в этой программе (их ровно 2). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

```
n = int(input())
while n != 0:
    digit = n % 10
    n //= 10
print(digit)
```

141. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран произведение цифр введенного числа. Программист торопился и написал программу неправильно.  
Найдите все ошибки в этой программе (их ровно 3). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.

```
n = int(input())
product = 1
while n != 0:
    digit = n % 10
    product = product * digit
    n //= 10
print(product)
```

142. Дано натуральное число  $n$ , ( $n \leq 9$ ). Напишите программу, которая печатает таблицу размером  $n \times 3$  состоящую из данного числа (числа отделены одним пробелом).  
На вход программе подается одно натуральное число.  
Программа должна вывести таблицу размером  $n \times 3$  состоящую из данного числа.  
В конце строки может быть пробел.

```
n = int(input())
for i in range(1, n + 1):
    for j in range(n, n + 3):
        print(n, end=' ')
    print()
```

143. Дано натуральное число  $n$ , ( $n \leq 9$ ). Напишите программу, которая печатает таблицу размером  $n \times 5$ , где в  $i$ -ой строке указано число  $i$  (числа отделены одним пробелом).  
На вход программе подается одно натуральное число.  
Программа должна вывести таблицу размером  $n \times 5$  в соответствии с условием.  
В конце строки может быть пробел.

```
n = int(input())
for i in range(1, n + 1):
    for j in range(n, n + 5):
        print(i, end=' ')
    print()
```

144. Дано натуральное число  $n$ , ( $n \leq 9$ ). Напишите программу, которая печатает таблицу сложения для всех чисел от 1 до  $n$  в соответствии с примером.  
На вход программе подается одно натуральное число.  
Программа должна вывести таблицу сложения для всех чисел от 1 до  $n$ .  
В конце строки может быть пробел.

```
n = int(input())
for i in range(1, n + 1):
    for j in range(1, 10):
        print(i, '+', j, '=', i + j, end='\n')
    print()
```

145. Дано нечетное натуральное число  $n$ . Напишите программу, которая печатает равнобедренный звездный треугольник с основанием, равным  $n$  в соответствии с примером:

```
*
**
***
****
***
**
*
```

На вход программе подается одно нечетное натуральное число.  
Программа должна вывести треугольник в соответствии с условием.

```
n = int(input())
for i in range(n//2 + 1):
    for j in range(i + 1):
        print('*', end='')
    print()
for i in range(n//2, 0, -1):
    for j in range(i):
        print('*', end='')
    print()
```



146. Дано натуральное число  $n$ . Напишите программу, которая печатает численный треугольник в соответствии с примером:

```
1
22
333
4444
55555
...
```

На вход программе подается одно натуральное число.  
Программа должна вывести треугольник в соответствии с условием.

```
n = int(input())
for i in range(1, n + 1):
    for j in range(i):
        print(i, end="")
    print()
```

147. Дано натуральное число  $n$ . Напишите программу, которая печатает численный треугольник с высотой равной  $n$ , в соответствии с примером:

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21
...
```

На вход программе подается одно натуральное число.  
Программа должна вывести треугольник в соответствии с условием.

```
n = int(input())
counter = 0
for i in range(1, n + 1):
    for j in range(i):
        counter += 1
        print(counter, end=' ')
    print()
```

148. Дано натуральное число  $n$ . Напишите программу, которая печатает численный треугольник с высотой равной  $n$ , в соответствии с примером:

```
1
121
12321
1234321
123454321
...
```

На вход программе подается одно натуральное число.  
Программа должна вывести треугольник в соответствии с условием.

```
n = int(input())
for i in range(1, n + 1):
    for j in range(i):
        print(j + 1, end="")
    for k in range(i - 1, 0, - 1):
        print(k, end="")
    print()
```

149. На вход программе подается два натуральных числа  $a$  и  $b$  ( $a < b$ ). Напишите программу, которая находит натуральное число из отрезка  $[a; b]$  с максимальной суммой делителей.

На вход программе подаются два числа, каждое на отдельной строке.  
Программа должна вывести два числа на одной строке, разделенных пробелом: число с максимальной суммой делителей и сумму его делителей.  
Если таких чисел несколько, то выведите наибольшее из них.

```
a = int(input())
b = int(input())
biggest = 0
total = 0
for i in range(a, b + 1):
    counter = 0
    for j in range(1, i + 1):
        if i % j == 0:
            counter += j
    if counter >= total:
        total = counter
        biggest = i
print(biggest, total)
```

150. На вход программе подается натуральное число  $n$ . Напишите программу, выводящую графическое изображение делимости чисел от 1 до  $n$  включительно. В каждой строке надо напечатать очередное число и столько символов «+», сколько делителей у этого числа.

На вход программе подается одно натуральное число.  
Программа должна вывести графическое изображение чисел от 1 до  $n$ , каждое на отдельной строке.

```
n = int(input())
total = 0
for i in range(1, n + 1):
    counter = 0
    for j in range(1, i + 1):
        if i % j == 0:
            counter += 1
            total = counter
    print(i, total * '+', sep="")
```

151. На вход программе подается натуральное число  $n$ . Напишите программу, которая находит цифровой корень данного числа. Цифровой корень числа  $n$  получается следующим образом: если сложить все цифры этого числа, затем все цифры найденной суммы и повторить этот процесс, то в результате будет получено однозначное число (цифра), которое и называется цифровым корнем данного числа.

На вход программе подается одно натуральное число.  
Программа должна вывести цифровой корень введенного числа.

```
n = int(input())
while n > 9:
    total = 0
    while n > 0:
        last = n % 10
        total += last
        n //= 10
    n = total
print(n)
```

152. Дано натуральное число  $n$ . Напишите программу, которая выводит значение суммы  $1!+2!+3!+...+n!$ .

На вход программе подается одно натуральное число.  
Программа должна вывести значение суммы  $1!+2!+3!+...+n!$ .

```
n = int(input())
counter = 1
total = 0
for i in range(1, n + 1):
    counter *= i
    total += counter
print(total)
```

**153. На вход программе подается два натуральных числа  $a$  и  $b$  ( $a < b$ ). Напишите программу, которая находит все простые числа от  $a$  до  $b$  включительно.**  
**На вход программе подаются два числа, каждое на отдельной строке.**  
**Программа должна вывести все простые числа от  $a$  до  $b$  включительно, каждое на отдельной строке.**  
**Число 11 простым не является.**

```
a = int(input())
b = int(input())
for i in range(a, b + 1):
    count = 0
    for j in range(1, i + 1):
        if i % j == 0:
            count += 1
    if count == 2:
        print(i)
```

**154. На обработку поступает натуральное число. Нужно написать программу, которая выводит на экран сумму чётных цифр этого числа или 0, если чётных цифр в записи нет. Программист торопился и написал программу неправильно.**  
**Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.**

```
n = int(input())
s = 0
while n > 0:
    if n % 2 == 0:
        s += n % 10
    n //= 10
print(s)
```

**155. На обработку поступает последовательность из 8 целых чисел. Известно, что вводимые числа по абсолютной величине не превышают  $10^{**12}$ . Нужно написать программу, которая выводит на экран количество делящихся нацело на 4 чисел в исходной последовательности и максимальное делящееся нацело на 4 число. Если делящихся нацело на 4 чисел нет, требуется на экран вывести «NO». Программист торопился и написал программу неправильно.**  
**Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.**

```
n = 8
count = 0
maximum = -10 ** 12
for i in range(1, n + 1):
    x = int(input())
    if x % 4 == 0:
        count += 1
        if x > maximum:
            maximum = x
if count > 0:
    print(count)
    print(maximum)
else:
    print('NO')
```

**156. На обработку поступает последовательность из 4 целых чисел. Известно, что вводимые числа по абсолютной величине не превышают  $10^{**8}$ . Нужно написать программу, которая выводит на экран количество нечётных чисел в исходной последовательности и максимальное нечётное число. Если нечётных чисел нет, требуется на экран вывести «NO». Программист торопился и написал программу неправильно.**  
**Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк.**

```
n = 4
count = 0
maximum = -10 ** 8
for i in range(1, n + 1):
    x = int(input())
    if x % 2 != 0:
        count += 1
        if x > maximum:
            maximum = x
if count > 0:
    print(count)
    print(maximum)
else:
    print('NO')
```

**157. На вход программе подается натуральное число  $n$ . Напишите программу, которая печатает звездную рамку размерами  $n \times 19$ .**  
**На вход программе подаётся натуральное число  $n \in [3; 19]$  — высота звездной рамки.**  
**Программа должна вывести звездную рамку размерами  $n \times 19$ .**

```
n = int(input())
print('*' * 19)
for i in range(n-2):
    print('*' + ' ' * 17 + '*')
print('*' * 19)
```

**158. Дано натуральное число  $n$ , ( $n > 99$ ). Напишите программу, которая определяет его третью (с начала) цифру.**  
**На вход программе подается одно натуральное число, состоящее как минимум из трех цифр.**  
**Программа должна вывести его третью (с начала) цифру.**

```
n = int(input())
digit = 0
while n > 100:
    last = n % 10
    digit = last
    n //= 10
print(digit)
```

**159. Дано натуральное число. Напишите программу, которая вычисляет:**

- количество цифр 3 в нем;
- сколько раз в нем встречается последняя цифра;
- количество четных цифр;
- сумму его цифр, больших пяти;
- произведение цифр, больших семи (если цифр больших семи нет, то вывести 1, если такая цифра одна, то вывести ее);
- сколько раз в нем встречается цифры 0 и 5 (всего суммарно).

**На вход программе подается одно натуральное число.**  
**Программа должна вывести значения указанных величин в указанном порядке.**

```
n = int(input())
posled = n % 10
sum_three = 0
sum_last = 0
even = 0
more_five = 0
more_seven = 1
zero_five = 0
while n > 0:
    last = n % 10
    if last == 3:
        sum_three += 1
    if posled == last:
        sum_last += 1
    if last % 2 == 0:
        even += 1
    if last > 5:
        more_five += last
    if last > 7:
```

```
        more_seven *= last
    if last == 0 or last == 5:
        zero_five += 1
    n //= 10
print(sum_three)
print(sum_last)
print(even)
print(more_five)
print(more_seven)
print(zero_five)
```

160. Дополните приведенный код, используя индексатор, так чтобы он вывел символ запятой.

```
s = "In 2010, someone paid 10k Bitcoin for two pizzas."
print(s[7])
```

161. Дополните приведенный код, используя индексатор, так чтобы он вывел символ `w`.

```
s = "In 2010, someone paid 10k Bitcoin for two pizzas."
print(s[-10])
```

162. На вход программе подается одна строка. Напишите программу, которая выводит элементы строки с индексами `0, 2, 4, ...` в столбик.

На вход программе подается одна строка.

Программа должна вывести элементы строки с индексами `0, 2, 4, ...`, каждое на отдельной строке.

```
s = input()
for i in range(0, len(s), 2):
    print(s[i])
```

163. На вход программе подается одна строка. Напишите программу, которая выводит в столбик элементы строки в обратном порядке.

На вход программе подается одна строка.

Программа должна вывести в столбик элементы строки в обратном порядке.

```
s = input()
for i in range(1, len(s) + 1):
    print(s[-i])
```

164. На вход программе подаются три строки: имя, фамилия и отчество. Напишите программу, которая выводит инициалы человека.

На вход программе подаются три строки, каждая на отдельной строке.

Программа должна вывести ФИО человека.

Гарантируется, что имя, фамилия и отчество начинаются с заглавной буквы.

```
s1, s2, s3 = input(), input(), input()
print(s2[0], s1[0], s3[0], sep="")
```

165. На вход программе подается одна строка состоящая из цифр. Напишите программу, которая считает сумму цифр данной строки.

На вход программе подается одна строка состоящая из цифр.

Программа должна вывести сумму цифр данной строки.

```
s = input()
total = 0
for i in s:
    total += int(i)
print(total)
```

166. На вход программе подается одна строка. Напишите программу, которая выводит сообщение «Цифра» (без кавычек), если строка содержит цифру. В противном случае вывести сообщение «Цифр нет» (без кавычек).

```
s = input()
flag = False
for i in range(len(s)):
    if s[i] in '0123456789':
        flag = True
if flag == True:
    print('Цифра')
else:
    print('Цифр нет')
```

167. На вход программе подается одна строка. Напишите программу, которая определяет сколько раз в строке встречаются символы `+` и `*`.

На вход программе подается одна строка.

Программа должна вывести сколько раз встречаются символы `+` и `*` в строке.

```
s = input()
count1 = 0
count2 = 0
for i in range(len(s)):
    if s[i] in '+':
        count1 += 1
    if s[i] in '*':
        count2 += 1
print('Символ + встречается', count1, 'раз')
print('Символ * встречается', count2, 'раз')
```

168. На вход программе подается одна строка. Напишите программу, которая определяет сколько в ней одинаковых соседних символов.

На вход программе подается одна строка.

Программа должна вывести количество одинаковых соседних символов.

```
s = input()
count = 0
for i in range(len(s)-1):
    if s[i] == s[i + 1]:
        count += 1
print(count)
```

169. На вход программе подается одна строка с буквами русского языка. Напишите программу, которая определяет количество гласных и согласных букв.

На вход программе подается одна строка.

Программа должна вывести количество гласных и согласных букв.

```
s = input()
vowel = 0
consonant = 0
for i in range(len(s)):
    if s[i] in 'ауоыиэяюёеАУОЫИЭЯЮЁЕ':
        vowel += 1
    if s[i] in 'бвгджзйклмнпрстфхцчшщБВГДЖЗЙКЛМНПРСТФХЦЧШЩ':
        consonant += 1
print('Количество гласных букв равно', vowel)
print('Количество согласных букв равно', consonant)
```

170. На вход программе подается натуральное число, записанное в десятичной системе счисления. Напишите программу, которая переводит данное число в двоичную систему счисления.

На вход программе подается одно натуральное число.

Программа должна вывести число записанное в двоичной системе счисления.

```
n = int(input())
text = ""
while n != 0:
    text = str(n % 2) + text
    n //= 2
print(text)
```

**171. Дополните приведенный код, используя срезы, так чтобы он вывел первые 12 символов строки `s`.**

```
s = "In 2010, someone paid 10k Bitcoin for two pizzas."
print(s[:12])
```

**172. Дополните приведенный код, используя срезы, так чтобы он вывел последние 9 символов строки `s`.**

```
s = "In 2010, someone paid 10k Bitcoin for two pizzas."
print(s[-9:])
```

**173. Дополните приведенный код, используя срезы, так чтобы он вывел каждый 7 символ строки `s` начиная от начала строки.**

```
s = "In 2010, someone paid 10k Bitcoin for two pizzas."
print(s[::7])
```

**174. Дополните приведенный код, используя срезы, так чтобы он вывел строку `s` в обратном порядке.**

```
s = "In 2010, someone paid 10k Bitcoin for two pizzas."
print(s[::-1])
```

**175. На вход программе подается одно слово, записанное в нижнем регистре. Напишите программу, которая определяет является ли оно палиндромом.  
На вход программе подается одно слово в нижнем регистре.  
Программа должна вывести «YES», если слово является палиндромом и «NO» в противном случае.**

```
s = input()
flag = False
for i in range(len(s)):
    if s[i] == s[-i-1]:
        flag = True
if flag == True:
    print('YES')
else:
    print('NO')
```

**176. На вход программе подается одна строка. Напишите программу, которая выводит:**

- 1. общее количество символов в строке;
- 2. исходную строку повторенную 3 раза;
- 3. первый символ строки;
- 4. первые три символа строки;
- 5. последние три символа строки;
- 6. строку в обратном порядке;
- 7. строку с удаленным первым и последним символом.

На вход программе подается одна строка, длина которой больше 3 символов.  
Программа должна вывести данные в соответствии с условием. Каждое значение выводится на отдельной строке.

```
s = input()
total = len(s)
three_times = s * 3
first_symbol = s[0]
first_three_sym = s[:3]
last_three_sym = s[-3:]
reverse = s[::-1]
del_first_last = s[1:-1]
print(total)
print(three_times)
print(first_symbol)
print(first_three_sym)
print(last_three_sym)
print(reverse)
print(del_first_last)
```

**177. На вход программе подается одна строка. Напишите программу, которая выводит:**

- 1. третий символ этой строки;
- 2. предпоследний символ этой строки;
- 3. первые пять символов этой строки;
- 4. всю строку, кроме последних двух символов;
- 5. все символы с четными индексами;
- 6. все символы с нечетными индексами;
- 7. все символы в обратном порядке;
- 8. все символы строки через один в обратном порядке, начиная с последнего.

На вход программе подается одна строка, длина которой больше 5 символов.  
Программа должна вывести данные в соответствии с условием. Каждое значение выводится на отдельной строке.

```
s = input()
third_sym = s[2]
pre_last = s[-2]
first_five = s[:5]
without_two_last = s[:-2]
all_even = s[::2]
all_not_even = s[1::2]
reverse = s[::-1]
all_sym = s[::2]
print(third_sym)
print(pre_last)
print(first_five)
print(without_two_last)
print(all_even)
print(all_not_even)
print(reverse)
print(all_sym)
```

**178. На вход программе подается строка текста. Напишите программу, которая разрежет ее на две равные части, переставит их местами и выведет на экран.  
На вход программе подается строка текста.  
Программа должна вывести текст в соответствии с условием задачи.  
Если длина строки нечетная, то длина первой части должна быть на один символ больше.**

```
s = input()
length = len(s)
s1 = ""
s2 = ""
flag = False
if length % 2 == 0:
    flag = True
    for i in range(length//2):
        s1 += s[i]
    for j in range(length//2, length):
        s2 += s[j]
elif length % 2 != 0:
    for k in range(length//2 + 1):
        s1 += s[k]
    for g in range(length//2 + 1, length):
        s2 += s[g]
if flag == True:
    print(s2 + s1)
else:
    print(s2 + s1)
```

**179. На вход программе подается строка состоящая из имени и фамилии человека, разделенных одним пробелом. Напишите программу, которая проверяет, что имя и фамилия начинаются с заглавной буквы.  
На вход программе подается строка.  
Программа должна вывести «YES» если имя и фамилия начинаются с заглавной буквы и «NO» в противном случае.  
Строка содержит только буквы.**

- ```
s = input()
if s == s.title():
    print('YES')
else:
    print('NO')
```
180. На вход программе подается строка. Напишите программу, которая меняет регистр символов, другими словами замените все строчные символы заглавными и наоборот.

```
s = input()
print(s.swapcase())
```
181. На вход программе подается строка текста. Напишите программу, которая определяет является ли оттенок текста хорошим или нет. Текст имеет хороший оттенок, если содержит подстроку «хорош» во всевозможных регистрах.  
На вход программе подается строка текста.  
Программа должна вывести «YES» если текст имеет хороший оттенок и «NO» в противном случае.  
Текст содержащий хорош, ХОРОШ, Хорош, хОРОШ и т.д. имеет хороший оттенок.

```
s = input()
s = s.lower()
if 'хорош' in s:
    print('YES')
else:
    print('NO')
```
182. На вход программе подается строка. Напишите программу, которая подсчитывает количество буквенных символов в нижнем регистре.  
На вход программе подается строка.  
Программа должна вывести количество буквенных символов в нижнем регистре.

```
s = input()
count = 0
for i in range(len(s)):
    if 'a' <= s[i] <= 'z':
        count += 1
print(count)
```
183. На вход программе подается строка текста, состоящая из слов, разделенных ровно одним пробелом. Напишите программу, которая подсчитывает количество слов в ней.  
На вход программе подается строка текста.  
Программа должна вывести количество слов.  
Строка текста не содержит пробелов в начале и конце.  
Используйте для решения задачи метод `count`.

```
s = input()
print(s.count(' ') + 1)
```
184. На вход программе подается строка генетического кода, состоящая из букв А (аденин), Г (гуанин), Ц (цитозин), Т (тимин). Напишите программу, которая подсчитывает сколько аденина, гуанина, цитозина и тимина входит в данную строку генетического кода.  
На вход программе подается строка генетического кода, состоящая из символов А, Г, Ц, Т, а, г, ц, т.  
Программа должна вывести сколько гуанина, тимина, цитозина, аденина входит в данную строку генетического кода.  
Строка не содержит символов, кроме как А, Г, Ц, Т, а, г, ц, т.

```
s = input()
s = s.lower()
print('Аденин:', s.count('a'))
print('Гуанин:', s.count('r'))
print('Цитозин:', s.count('ц'))
print('Тимин:', s.count('т'))
```
185. Джим Хоппер с помощью радиоприемника пытается получить сообщение Оди. На приемник ему поступает *n* различных последовательностей кода Морзе. Декодировав их, он получает последовательности из цифр и строчного латинского алфавита, при этом во всех сообщениях Оди содержится число 11, причем минимум 3 раза. Помогите определить Джиму количество сообщений от Оди. В первой строке подаётся число *n* – количество сообщений, в последующих *n* строках вводятся строки, содержащие латинские строчные буквы и цифры.  
Программа должна вывести количество строк в которых содержится число 11 минимум 3 раза.  
Числа 11 обязательно должны быть разделены другими символами, нужно подсчитать вхождение последовательности символов "11", т.е. например в строке "111" содержится одна такая последовательность, в то время как в "1111" их уже две.

```
n = int(input())
count = 0
for i in range(n):
    s = input()
    if s.count('11') >= 3:
        count += 1
print(count)
```
186. На вход программе подается строка текста. Напишите программу, которая подсчитывает количество цифр в данной строке.  
На вход программе подается строка текста.  
Программа должна вывести количество цифр в данной строке.

```
s = input()
count = 0
for i in range(len(s)):
    if '0' <= s[i] <= '9':
        count += 1
print(count)
```
187. На вход программе подается строка текста. Напишите программу, которая проверяет, что строка заканчивается подстрокой `.com` или `.ru`.  
На вход программе подается строка текста.  
Программа должна вывести «YES» если введенная строка заканчивается подстрокой `.com` или `.ru` и «NO» в противном случае.

```
s = input()
if s.endswith('.com') or s.endswith('.ru'):
    print('YES')
else:
    print('NO')
```
188. На вход программе подается строка текста. Напишите программу, которая выводит на экран символ, который появляется наиболее часто.  
На вход программе подается строка текста. Текст может содержать строчные и заглавные буквы английского и русского алфавита, а также цифры.  
Программа должна вывести символ, который появляется наиболее часто.  
Если таких символов несколько, следует вывести последний по порядку символ.  
Следует различать заглавные и строчные буквы, а также буквы русского и английского алфавита.

```
s = input()
count1 = 0
count2 = 0
for i in s:
    if s.count(i) >= count1:
        count1 = s.count(i)
        count2 = i
print(count2)
```
189. На вход программе подается строка текста. Если в этой строке буква «f» встречается только один раз, выведите её индекс. Если она встречается два и более раз, выведите индекс её первого и последнего вхождения на одной строке, разделенных символом пробела. Если буква «f» в данной строке не встречается, следует вывести «NO».



```
s = input()
index1 = 0
index2 = 0
for i in s:
    if s.count('f') == 1:
        index1 = s.find('f')
        print(index1)
        break
    if s.count('f') > 1:
        index1 = s.find('f')
        index2 = s.rfind('f')
        print(index1, index2)
        break
    if s.count('f') == 0:
        print('NO')
        break
```

190. На вход программе подается строка текста, в которой буква «h» встречается минимум два раза. Напишите программу, которая удаляет из этой строки первое и последнее вхождение буквы «h», а также все символы, находящиеся между ними.

```
s = input()
s1 = s.find('h')
s2 = s.rfind('h') + 1
print(s[s1] + s[s2:])
```

191. Дополните приведенный код, используя форматирование строк с помощью метода `format`, так чтобы он вывел текст:

«In 2010, someone paid 10k Bitcoin for two pizzas.» (без кавычек).

```
year = '2010'
price = '10k'
coin = 'Bitcoin'
s = 'In {0}, someone paid {1} {2} for two pizzas.'.format(year, price, coin)
print(s)
```

192. Дополните приведенный код, используя форматирование строк с помощью f-строк, так чтобы он вывел текст:

«In 2010, someone paid 10K Bitcoin for two pizzas.» (без кавычек).

```
year = 2010
amount = '10K'
currency = 'Bitcoin'
print(f'In {year}, someone paid {amount} {currency} for two pizzas.')
```

193. На вход программе подаются два числа *a* и *b*. Напишите программу, которая для каждого кодового значения в диапазоне от *a* до *b* (включительно), выводит соответствующий ему символ из таблицы символов Unicode.

```
a = int(input())
b = int(input())
result = ""
for i in range(a, b + 1):
    result += chr(i) + ' '
print(result)
```

194. На вход программе подается строка текста. Напишите программу, которая переводит каждый ее символ в соответствующий ему код из таблицы символов Unicode.  
На вход программе подается строка текста.  
Программа должна вывести кодовые значения символов строки разделенных одним символом пробела.

```
s = input()
for i in s:
    print(ord(i), end=' ')
```

195. Легион Цезаря, созданный в 23 веке на основе Римской Империи не изменяет древним традициям и использует шифр Цезаря. Это их и подвело, ведь данный шифр очень простой. Однако в постапокалипсисе люди плохо знают все тонкости довоенного мира, поэтому ученые из НКР не могут понять как именно нужно декодировать данные сообщения. Напишите программу для декодирования этого шифра.  
В первой строке дается число *n* ( $1 \leq n \leq 25$ ) – сдвиг, во второй строке даётся закодированное сообщение в виде строки со строчными латинскими буквами.  
Программа должна вывести одну строку – декодированное сообщение. Обратите внимание, что нужно декодировать сообщение, а не закодировать.

```
n = int(input())
s = input()
for i in range(len(s)):
    x = ord(s[i]) - n
    if x < 97:
        x = 122 - (96 - x)
    print(chr(x), end="")
```

196. Вставьте пропущенный фрагмент кода, так чтобы в результате выполнения программы была выведена длина строки `s`.

```
s = 'Python rocks!'
print(len(s))
```

197. Вставьте пропущенный фрагмент кода, так чтобы в результате выполнения программы был выведен четвертый символ строки `s`.

```
s = 'Python rocks!'
print(s[3])
```

198. Вставьте пропущенный фрагмент кода, так чтобы в результате выполнения программы были выведены символы строки `s` со 2 по 5 включительно.

```
s = 'Python rocks!'
print(s[1:5])
```

199. Вставьте пропущенный фрагмент кода, так чтобы в результате выполнения программы была выведена строка `s` без ведущих и замыкающих пробельных символов.

```
s = ' Python rocks! '
print(s.strip())
```

200. Вставьте пропущенный фрагмент кода, так чтобы в результате выполнения программы была выведена строка `s` заглавными буквами (в верхнем регистре).

```
s = 'Python rocks!'
print(s.upper())
```

201. Вставьте пропущенный фрагмент кода, так чтобы в результате выполнения программы была выведена строка `s` в которой символ «o» заменен на символ «@».

```
s = 'Python rocks!'
print(s.replace('o', '@'))
```

202. На вход программе подается строка текста. Напишите программу, которая удаляет из нее все символы с индексами кратными 3, то есть символы с индексами `0, 3, 6, ...`.

```
s = input()
for i in range(len(s)):
    if i % 3 == 0:
        continue
```

```
print(s[i], end="")
```

**203. На вход программе подается строка текста. Напишите программу, которая заменяет все вхождения цифры 1 на слово «one».**

```
s = input()
for i in s:
    if i == '1':
        s = s.replace('1', 'one')
print(s)
```

**204. На вход программе подается строка текста. Напишите программу, которая удаляет все вхождения символа «@».**

```
s = input()
for i in s:
    if i == '@':
        s = s.replace('@', "")
print(s)
```

**205. На вход программе подается строка текста. Напишите программу, которая выводит индекс второго вхождения буквы «f». Если буква «f» встречается только один раз, выведите число -1, а если не встречается ни разу, выведите число -2.**

```
s = input()
if s.count('f') == 1:
    print(-1)
elif s.count('f') == 0:
    print(-2)
else:
    s = s.replace('f', '', 1)
    s = s.find('f')
    print(s)
```

**206. На вход программе подается строка текста в которой буква «h» встречается как минимум два раза. Напишите программу, которая возвращает исходную строку и переворачивает последовательность символов, заключенную между первым и последним вхождением буквы «h».**

```
s = input()
first_entry = s.find('h')
second_entry = s.rfind('h')
print(s[first_entry], s[second_entry:first_entry:-1], s[second_entry:], sep="")
```