ЗАДАЧИ И РЕШЕНИЯ ДЛЯ МОДУЛЯ: ЦИКЛ FOR

1

Ряд

Даны два целых числа A и B (при этом $A \leq B$). Выведите все числа от A до B включительно.

Входные данные

Вводятся два целых числа, не превосходящие 200 по абсолютному значению.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
1 10	1 2 3 4 5 6 7 8 9 10

Решение

Код

```
1 A=int(input())
2 B=int(input())
3 for i in range(A,B+1):
4     print(i)
```

Код из файла

2

Ряд — 2

Даны два целых числа A и B. Выведите все числа от A до B включительно в порядке возрастания, если $A < B$, или в порядке убывания в противном случае.

Входные данные

Вводятся два целых числа, не превосходящих 200 по абсолютному значению.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
1 10	1 2 3 4 5 6 7 8 9 10

Решение

```
a = int(input())
b = int(input())
if a < b:
    for i in range(a, b + 1):
        print(i)
else:
    for i in range(a, b - 1, -1):
        print(i)
```

*

Цифры числа

Дано 10-значное число. Выведите все цифры этого числа в обратном порядке по одной.

Входные данные

На вход подаётся натуральное 10-значное число.

Выходные данные

Выведите ответ на задачу. В качестве разделителя между цифрами можно использовать переводы строки и пробелы.

Примеры

Ввод	Вывод
1234567890	0 9 8 7 6 5 4 3 2 1

Решение

```
1 a=int(input())
2 for i in range(10):
3     print(a//10**i%10)
```

3

Диофантово уравнение

Даны числа a, b, c, d. Выведите в порядке возрастания все целые числа от 0 до 1000, которые являются корнями уравнения $a * x^3 + b * x^2 + c * x + d = 0$.

Входные данные

Вводятся целые числа a, b, c и d. Все числа не превосходят по модулю 30000.

Выходные данные

Выведите ответ на задачу. Если в указанном промежутке нет корней уравнения, то ничего выводить не нужно.

Примеры

Ввод	Вывод
-1 1 -1 1	1
1 1 1 1	

Решение

```
a=int(input())
b=int(input())
c=int(input())
d=int(input())
for x in range(1001):
#     print(x, a * x**3 + b * x**2 + c*x + d)
    if a * x**3 + b * x**2 + c*x + d == 0:
        print(x)
```

4

Факториал

По данному целому неотрицательному n вычислите значение $n!$.

Входные данные

Вводится число n ($0 \le n \le 15$).

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
3	6

Решение

```
1 res = 1
2 n = int(input())
3 for i in range(1, n + 1):
4     res *= i
5 print(res)
```

5

Сумма квадратов

По данному натуральному n вычислите сумму $1^2 + 2^2 + 3^2 + \ldots + n^2$.

Входные данные

Вводится натуральное число, не превосходящее 100.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
3	14

Решение

```
n = int(input())
sum = 0
for i in range(1, n + 1):
    sum += i ** 2
print(sum)

```

*

Остатки

Даны целые неотрицательные числа a, b, c, d, при этом $0 \leq c < d$. Выведите в порядке возрастания все числа от a до b, которые дают остаток c при делении на d.

В этой задаче нельзя использовать инструкцию *if*, операторы сравнения (< и т.д.), должен быть только один цикл.

Входные данные

На вход подаются четыре строки, в каждой из которых написано по одному неотрицательному целому числу — a, b, c, d. Все числа не превосходят $2 * 10^9$, $0 \leq c < d$.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
2 5 0 2	2 4
5 5 0 5	5

Решение

```
a=int(input())
b=int(input())
c=int(input())
d=int(input())
start=(a-c+d-1)//d*d+c
for i in range(start,b+1,d):
    print(i)
```

6

Четные числа

По данным двум натуральным числам A и B ($A \leq B$) выведите все чётные числа на отрезке от A до B. В этой задаче нельзя использовать инструкцию `if`.

Входные данные

Вводятся два натуральных числа A и B.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
1 10	2 4 6 8 10

Решение

```
a=int(input())
b=int(input())
for i in range(a,b-(b%2)+1,2):
  print(i%2+i)

```

7

Количество нулей

Дано несколько чисел. Подсчитайте, сколько из них равны нулю, и выведите это количество.

Входные данные

Сначала вводится число n, затем вводятся ровно n целых чисел. Все числа во входных данных целые неотрицательные, не превосходящие 30000.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
5 0 7 0 2 2	2

Решение

```
num_zeroes = 0
for i in range(int(input())):
    if int(input()) == 0:
        num_zeroes += 1
print(num_zeroes)
```

*

Делители

По данному натуральному числу выведите все его натуральные делители в порядке возрастания.

Входные данные

На вход подаётся единственное натуральное число n ($n \leq 1000$).

Выходные данные

Выведите все делители числа в порядке возрастания. Делители можно выводить на одной строке, разделяя пробелом, или на разных строках.

Примеры

Ввод	Вывод
10	1 2 5 10

Решение

```
1 n=int(input())
2 for d in range(1,n+1):
3     if n%d==0:
4         print(d)
5
```

*

Лесенка

По данному натуральному числу n выведите лесенку из n ступенек, i-я ступенька состоит из чисел от 1 до i без пробелов.

Входные данные

На вход подаётся натуральное число n ($n \leq 9$).

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
3	1 12 123

Решение

```
n=int(input())
s=0
for i in range(1,n+1):
    s=s*10+i
    print(s)
```

*

Сумма произведений соседних чисел

По заданной последовательности $a_1, a_2, \ldots, a_n$ чисел вычислите сумму $a_1 * a_2 + a_2 * a_3 + \cdots + a_{n-1} * a_n$.

Входные данные

Первая строка входных данных содержит число $n \geq 2$. В следующих n строках вводится по одному числу. В $i + 1$ строке содержится значение i-того элемента последовательности. Все числа во входном файле натуральные, не превосходящие 100.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
4 2 3 1 5	14

Решение

```
n=int(input())
s=0
prev=int(input())
for i in range(n-1):
    a=int(input())
    s=s+a*prev
    prev=a
print(s)
```

8

Сумма факториалов

По данному натуральном n вычислите сумму $1! + 2! + 3! + \ldots + n!$. В решении этой задачи можно использовать только один цикл.

Входные данные

Вводится натуральное число n, не превосходящее 30.

Выходные данные

Выведите ответ на задачу.

Примеры

Ввод	Вывод
3	9

Решение

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