Задача 1.

Лёгкая

**Снежинка**

Дано нечетное число n. Создайте двумерный массив из n×n элементов, заполнив его символами "." (каждый элемент массива является строкой из одного символа). Затем заполните символами "\*" среднюю строку массива, средний столбец массива, главную диагональ и побочную диагональ. В результате единицы в массиве должны образовывать изображение звездочки. Выведите полученный массив на экран, разделяя элементы массива пробелами.

ПРИМЕРЫ

ВХОДНЫЕ ДАННЫЕ

5

ВЫХОДНЫЕ ДАННЫЕ

\* . \* . \*

. \* \* \* .

\* \* \* \* \*

. \* \* \* .

\* . \* . \*

Начало формы

Задача 2.

Лёгкая

**Побочная диагональ**

Дано число n. Создайте массив размером n×n и заполните его по следующему правилу:

Числа на диагонали, идущей из правого верхнего в левый нижний угол равны 1.

Числа, стоящие выше этой диагонали, равны 0.

Числа, стоящие ниже этой диагонали, равны 2.

Полученный массив выведите на экран. Числа в строке разделяйте одним пробелом.

ПРИМЕРЫ

ВХОДНЫЕ ДАННЫЕ

4

ВЫХОДНЫЕ ДАННЫЕ

0 0 0 1

0 0 1 2

0 1 2 2

1 2 2 2

Начало формы

Задача 3.

Лёгкая

**Ходы ферзя**

На шахматной доске стоит ферзь. Отметьте положение ферзя на доске и все клетки, которые бьет ферзь. Клетку, где стоит ферзь, отметьте буквой “Q”, клетки, которые бьет ферзь, отметьте символами “\*”, остальные клетки заполните точками.

ВХОДНЫЕ ДАННЫЕ

Программа получает на вход координаты коня на шахматной доске в шахматной нотации (то есть в виде “e4”, где сначала записывается номер столбца (буква от “a” до “h”, слева направо), затем номеру строки (цифра от 1 до 8, снизу вверх).

ВЫХОДНЫЕ ДАННЫЕ

Выведите на экран изображение доски.

ПРИМЕРЫ

ВХОДНЫЕ ДАННЫЕ

c4

ВЫХОДНЫЕ ДАННЫЕ

. . \* . . . \* .

. . \* . . \* . .

\* . \* . \* . . .

. \* \* \* . . . .

\* \* Q \* \* \* \* \*

. \* \* \* . . . .

\* . \* . \* . . .

. . \* . . \* . .

Начало формы

Задача 4.

Лёгкая

**Транспонировать прямоугольную матрицу**

Дан двумерный массив размером n×m (n, m < 1000). Симметричный ему относительно главной диагонали массив называется транспонированным к данному. Он имеет размеры mn: строки исходного массива становятся столбцами транспонированного, столбцы исходного массива становятся строками транспонированного.

Для данного массива постройте транспонированный массив и выведите его на экран. Решение оформите в виде функции Transpose (A).

ПРИМЕРЫ

ВХОДНЫЕ ДАННЫЕ

3 4

11 12 13 14

21 22 23 24

31 32 33 34

ВЫХОДНЫЕ ДАННЫЕ

11 21 31

12 22 32

13 23 33

14 24 34

Начало формы

1.

Решение задачи №1

n = int(input())
a = [['.']\*n for j in range(n)]
for i in range(n):
   a[n // 2][i] = '\*'
   a[i][n // 2] = '\*'
   a[i][i] = '\*'
   a[-i-1][i] = '\*'
for i in range(n):
   print(\*a[i])

Решение задачи №2

n=int(input())

a = [print(\*[0 if n- i-1 > j else 1 if n-i-1 == j else 2 for j in range(n)]) for i in range(n)]

Решение задачи №3

a = input()

k = ord(a[0]) - 97

l = 8 - int(a[1])

f = k - l

for i in range(8):

    c = []

    for j in range(8):

        if i == l and j == k:

            c.append("Q")

        elif f == j - i or i == l or j == k or i + j == k + l:

            c.append("\*")

        else:

            c.append(".")

    print(" ".join(c))

    Решение задачи №4

def Transpose(a):

    m = len(a[0])

    n = len(a)

    sp = []

    for i in range(m):

        subsp = []

        for o in range(n):

            subsp.append(a[o][i])

        sp.append(subsp)

    return sp

n, m = map(int, input().split())

sp = []

for i in range(n):

    sp.append(input().split())

newsp = Transpose(sp)

for i in range(m):

    print(' '.join(newsp[i]))

Конец формы

Конец формы

Конец формы

Конец формы