

To print higher-resolution math symbols, click the
Hi-Res Fonts for Printing button on the jsMath control panel.

Задачи на генераторы двумерных массивов

Все задачи в этом разделе являются "Output only". В этих задачах нужно сдать однострочное выражение-генератор заданного массива. Внимательно читайте условия первой задачи.

Сайт: [Дистанционная подготовка](#)
Курс: Д. П. Кириенко. Программирование на языке Python (школа 179 г. Москвы)
Условия задач: Задачи на генераторы двумерных массивов
Printed by: maung myo
Date: Воскресенье 4 Март 2018, 01:29

Список задач

- [Задача А. Таблица умножения](#)
- [Задача В. Номера строк](#)
- [Задача С. Номера столбцов](#)
- [Задача D. Слева направо, сверху вниз](#)
- [Задача Е. Сверху вниз, слева направо](#)
- [Задача F. Снизу вверх, слева направо](#)
- [Задача G. Справа налево, снизу вверх](#)
- [Задача H. Квадраты - 1](#)
- [Задача I. Квадраты - 2](#)
- [Задача J. Шахматная доска](#)
- [Задача K. Нумерация диагоналей - 1](#)
- [Задача L. Нумерация диагоналей - 2](#)
- [Задача M. Расстояние до побочной диагонали](#)
- [Задача N. Каждая четвертая диагональ](#)
- [Задача O. Решетка](#)
- [Задача P. Выше и ниже главной диагонали](#)

- [Задача Q. Андреевский крест](#)
 - [Задача R. Квадранты](#)
 - [Задача S. Большие квадраты](#)
 - [Задача T. Змейка](#)
 - [Задача U. Заполнение в шахматном порядке](#)
 - [Задача V. Вложенные прямоугольники](#)
 - [Задача W. Крест](#)
 - [Задача X. Ромбы](#)
 - [Задача Y. Вдоль главной диагонали](#)
 - [Задача Z. Заполнение по диагоналям](#)
 - [Задача AA. Заполнение по спирали](#)
-

Таблица умножения

Задача А. Таблица умножения

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу в виде таблицы умножения.

Пример для $n = 5, m = 6$

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

Номера строк

Задача В. Номера строк

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу, "пронумеровав" строки.

Пример для $n = 5, m = 6$

```
0 0 0 0 0 0
1 1 1 1 1 1
2 2 2 2 2 2
3 3 3 3 3 3
4 4 4 4 4 4
```

Номера столбцов

Задача С. Номера столбцов

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами): 0 0 0 0 0 0 1 2 3 4 5 0 2 4 6 8 10 0 3 6 9 12 15 0 4 8 12 16 20 В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу, "пронумеровав" столбцы.

Пример для $n = 5, m = 6$

```
0 1 2 3 4 5
0 1 2 3 4 5
0 1 2 3 4 5
0 1 2 3 4 5
0 1 2 3 4 5
```

Слева направо, сверху вниз

Задача D. Слева направо, сверху вниз

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу.

Пример для $n = 5, m = 6$

```
0 1 2 3 4 5
6 7 8 9 10 11
12 13 14 15 16 17
18 19 20 21 22 23
24 25 26 27 28 29
```

Сверху вниз, слева направо

Задача Е. Сверху вниз, слева направо

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу.

Пример для $n = 5, m = 6$

```
0 5 10 15 20 25
1 6 11 16 21 26
2 7 12 17 22 27
3 8 13 18 23 28
4 9 14 19 24 29
```

Снизу вверх, слева направо

Задача F. Снизу вверх, слева направо

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу.

Пример для $n = 5, m = 6$

```
4 9 14 19 24 29
3 8 13 18 23 28
2 7 12 17 22 27
1 6 11 16 21 26
0 5 10 15 20 25
```


Справа налево, снизу вверх

Задача G. Справа налево, снизу вверх

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу.

Пример для $n = 5, m = 6$

```
29 28 27 26 25 24
23 22 21 20 19 18
17 16 15 14 13 12
11 10 9 8 7 6
5 4 3 2 1 0
```

Квадраты - 1

Задача Н. Квадраты - 1

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните квадратный массив целыми числами по образцу.

Пример для $n = 5$

```
0 1 2 3 4
1 1 2 3 4
2 2 2 3 4
3 3 3 3 4
4 4 4 4 4
```

Так как массив квадратный, то число столбцов в нем также равно n , а переменная m не определена и не должна использоваться в ответе.

Квадраты - 2

Задача I. Квадраты - 2

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните квадратный массив целыми числами по образцу.

Пример для $n = 5$

```
0 1 2 3 4
0 1 2 3 3
0 1 2 2 2
0 1 1 1 1
0 0 0 0 0
```

Шахматная доска

Задача J. Шахматная доска

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу в виде шахматной доски

Пример для $n = 5, m = 6$

```
1 0 1 0 1 0
0 1 0 1 0 1
1 0 1 0 1 0
0 1 0 1 0 1
1 0 1 0 1 0
```

Нумерация диагоналей - 1

Задача К. Нумерация диагоналей - 1

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу: нужно пронумеровать диагонали

Пример для $n = 5, m = 6$

```
0 1 2 3 4 5
1 2 3 4 5 6
2 3 4 5 6 7
3 4 5 6 7 8
4 5 6 7 8 9
```

Нумерация диагоналей - 2

Задача L. Нумерация диагоналей - 2

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните массив целыми числами по образцу.

Пример для $n = 5, m = 6$

```
0 1 2 0 1 2
2 0 1 2 0 1
1 2 0 1 2 0
0 1 2 0 1 2
2 0 1 2 0 1
```

Расстояние до побочной диагонали

Задача М. Расстояние до побочной диагонали

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните квадратный массив целыми числами по образцу.

Пример для $n = 5$

```
4 3 2 1 0
3 2 1 0 1
2 1 0 1 2
1 0 1 2 3
0 1 2 3 4
```

Каждая четвертая диагональ

Задача N. Каждая четвертая диагональ

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните **квадратный** массив целыми числами по образцу. На побочной диагонали стоят единицы, каждая четвертая по счету диагональ, параллельная побочной, также заполнена единицами.

Пример для $n = 10$

```
0 1 0 0 0 1 0 0 0 1
1 0 0 0 1 0 0 0 1 0
0 0 0 1 0 0 0 1 0 0
0 0 1 0 0 0 1 0 0 0
0 1 0 0 0 1 0 0 0 1
1 0 0 0 1 0 0 0 1 0
0 0 0 1 0 0 0 1 0 0
```



```
0 0 1 0 0 0 1 0 0 0
0 1 0 0 0 1 0 0 0 1
1 0 0 0 1 0 0 0 1 0
```

Решетка

Задача О. Решетка

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами по образцу (нули стоят на пересечении строк и столбцов с нечетными номерами, в остальных клетках стоят единицы).

Пример для $n = 5, m = 6$

```
1 1 1 1 1 1
1 0 1 0 1 0
1 1 1 1 1 1
1 0 1 0 1 0
1 1 1 1 1 1
```

Выше и ниже главной диагонали

Задача Р. Выше и ниже главной диагонали

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами по образцу. На главной диагонали стоят нули, выше главной диагонали — единицы, ниже — двойки

Пример для $n = 5, m = 6$

```
0 1 1 1 1 1
2 0 1 1 1 1
2 2 0 1 1 1
2 2 2 0 1 1
2 2 2 2 0 1
```

Андреевский крест

Задача Q. Андреевский крест

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните **квадратный** массив целыми числами по образцу. На главной и побочной диагоналях стоят единицы, остальные элементы равны нулю.

Пример для $n = 7$

```
1 0 0 0 0 0 1
0 1 0 0 0 1 0
0 0 1 0 1 0 0
0 0 0 1 0 0 0
0 0 1 0 1 0 0
0 1 0 0 0 1 0
1 0 0 0 0 0 1
```

Квадранты

Задача R. Квадранты

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните **квадратный** массив целыми числами по образцу. На главной и побочных диагоналях стоят нули, эти диагонали делят массив на четыре части. В верхней части записаны единицы, в правой записаны двойки, в нижней записаны тройки, в левой записаны четверки.

Пример для $n = 8$

```
0 1 1 1 1 1 1 0
4 0 1 1 1 1 0 2
4 4 0 1 1 0 2 2
4 4 4 0 0 2 2 2
4 4 4 0 0 2 2 2
4 4 0 3 3 0 2 2
```

4 0 3 3 3 3 0 2

0 3 3 3 3 3 3 0

Большие квадраты

Задача S. Большие квадраты

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами квадратами размера 2×2 из единиц, квадраты идут в шахматном порядке, начиная с левого верхнего угла.

Пример для $n = 7, m = 10$

```
1 1 0 0 1 1 0 0 1 1
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
0 0 1 1 0 0 1 1 0 0
1 1 0 0 1 1 0 0 1 1
1 1 0 0 1 1 0 0 1 1
0 0 1 1 0 0 1 1 0 0
```

Змейка

Задача Т. Змейка

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами начиная с нуля по строчкам “змейкой”: слева направо, затем справа налево. строчки обходятся сверху вниз.

Пример для $n = 5, m = 6$

```
0 1 2 3 4 5
11 10 9 8 7 6
12 13 14 15 16 17
23 22 21 20 19 18
24 25 26 27 28 29
```


Заполнение в шахматном порядке

Задача U. Заполнение в шахматном порядке

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами начиная с единицы по строчкам слева направо, строчки обходятся сверху вниз. При этом в левом верхнем углу записано число 1, далее заполняются только клетки, имеющие тот же цвет в шахматной раскраске, что и левый верхний угол, а в остальных клетках записано число 0.

Пример для $n = 5, m = 6$

```
1 0 2 0 3 0
0 4 0 5 0 6
7 0 8 0 9 0
0 10 0 11 0 12
13 0 14 0 15 0
```

Вложенные прямоугольники

Задача V. Вложенные прямоугольники

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив вложенными прямоугольниками. Клетки, находящиеся на краях прямоугольника, заполняются числом 0, соседние с ними клетки заполняются числом 1, соседние с ними — числом 2 и т.д.

Пример для $n = 6, m = 9$

```
0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 0
0 1 2 2 2 2 2 1 0
0 1 2 2 2 2 2 1 0
0 1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0
```

Крест

Задача W. Крест

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните **квадратный** массив целыми числами по образцу. В углах записан число 0, в соседних с ними с углами клетках записано число 1, в соседних с ними клетках записано число 2 и т.д. Клетки считаются соседними, если у них есть общая сторона или вершина.

Пример для $n = 8$

```
0 1 2 3 3 2 1 0
1 1 2 3 3 2 1 1
2 2 2 3 3 2 2 2
3 3 3 3 3 3 3 3
3 3 3 3 3 3 3 3
2 2 2 3 3 2 2 2
```

```
1 1 2 3 3 2 1 1  
0 1 2 3 3 2 1 0
```

Ромбы

Задача X. Ромбы

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните **квадратный** массив целыми числами по образцу. В углах записан число 0, в соседних с ними с углами клетках записано число 1, в соседних с ними клетках записано число 2 и т.д. Клетки считаются соседними, если у них есть общая сторона.

Пример для $n = 7$

```
0 1 2 3 3 2 1 0
1 2 3 4 4 3 2 1
2 3 4 5 5 4 3 2
3 4 5 6 6 5 4 3
3 4 5 6 6 5 4 3
2 3 4 5 5 4 3 2
```

```
1 2 3 4 4 3 2 1
0 1 2 3 3 2 1 0
```

Вдоль главной диагонали

Задача Y. Вдоль главной диагонали

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните **квадратный** массив целыми числами по образцу. Часть массива ниже главной диагонали заполнена нулями. На главной диагонали записаны числа $0, 1, \dots, n - 1$. На диагонали выше главной — числа $n, n + 1, \dots, 2n - 2$. Следующая диагональ заполняется, начиная с числа $2n - 1$ и т.д.

Пример для $n = 7$

```
0 7 13 18 22 25 27
0 1 8 14 19 23 26
0 0 2 9 15 20 24
0 0 0 3 10 16 21
0 0 0 0 4 11 17
```

```
0 0 0 0 0 5 12
0 0 0 0 0 0 6
```

Заполнение по диагоналям

Задача Z. Заполнение по диагоналям

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами начиная с нуля вдоль диагоналей, параллельных побочной диагонали. Заполнение проводится сверху вниз, диагонали обходятся слева направо.

Пример для $n = 7, m = 10$

```
0 1 3 6 10 15 21 28 35 42
2 4 7 11 16 22 29 36 43 49
5 8 12 17 23 30 37 44 50 55
9 13 18 24 31 38 45 51 56 60
14 19 25 32 39 46 52 57 61 64
20 26 33 40 47 53 58 62 65 67
27 34 41 48 54 59 63 66 68 69
```


Заполнение по спирали

Задача АА. Заполнение по спирали

В этой задаче нужно придумать генератор — однострочное выражение на языке Python, результатом вычисления которого будет двумерный массив (список вложенных списков), заполненный по некоторому правилу.

Например, пусть нужно задать следующий массив (для удобства добавлены дополнительные пробелы между элементами):

```
0 0 0 0 0 0
0 1 2 3 4 5
0 2 4 6 8 10
0 3 6 9 12 15
0 4 8 12 16 20
```

В этом массиве $n = 5$ строк, $m = 6$ столбцов, и элемент в строке i и столбце j вычисляется по формуле: $A[i][j] = i * j$.

Ответом на это задание будет следующее выражение-генератор:

`[[i * j for j in range(m)] for i in range(n)]` Вам нужно создать текстовый файл, записать в его первой строчке заданное выражение (только одно выражение в квадратных скобках, например, достаточно просто скопировать текст, записанный выше) и сдать на проверку данный файл. Не нужно писать инструкции вроде `A = [...]` или `print(...)`.

В выражении должны использоваться переменные n и m , означающие число строк и столбцов в массиве. Считывать эти переменные с клавиатуры не нужно, они уже будут автоматически определены на момент запуска вашего решения.

Если в задании сказано, что массив — квадратный, то число строк и столбцов в нем равно n , а значение m не определено и использовать его нельзя.

Проверка будет осуществляться при помощи интерпретатора языка Python версии 3, в частности, это означает, что в генераторах нужно использовать функции `range`, а не `xrange`.

Заполните прямоугольный массив целыми числами начиная с нуля спиралью, закручивающейся по часовой стрелке из левого верхнего угла к центру массива.

Пример для $n = 7, m = 10$

```
0 1 2 3 4 5 6 7 8 9
29 30 31 32 33 34 35 36 37 10
28 51 52 53 54 55 56 57 38 11
27 50 65 66 67 68 69 58 39 12
26 49 64 63 62 61 60 59 40 13
25 48 47 46 45 44 43 42 41 14
24 23 22 21 20 19 18 17 16 15
```