**Вложенные генераторы**

Научимся использовать вложенные генераторы для заполнения двумерных массивов по формулам, которые зависят от номера строки и номера столбца элемента.

Мы уже встречались с генератором, который позволяет заполнить двумерный массив нулями:

a = [[0] \* m **for** i **in** range(n)]

Можно модифицировать этот генератор таким образом, чтобы внутренний список, состоящий из m нулей, тоже создавался через генератор:

a = [[0 **for** j **in** range(m)] **for** i **in** range(n)]

Таким образом, мы получили два генератора, один из которых вложен в другой. Теперь, если в теле вложенного генератора вместо числа 0 записать какую-то формулу, зависящую от индексов i и j, то мы получим способ нетривиально заполнить наш двумерный массив. Ниже мы рассмотрим несколько примеров такого заполнения.

**Пример 1**

a = [[j **for** j **in** range(m)] **for** i **in** range(n)]

Для n=3, m=4 этот генератор заполнит двумерный массив следующим образом:

0 1 2 3

0 1 2 3

0 1 2 3

**Пример 2**

a = [[i **for** j **in** range(m)] **for** i **in** range(n)]

Для n=3, m=4 этот генератор заполнит двумерный массив так:

0 0 0 0

1 1 1 1

2 2 2 2

**Пример 3**

a = [[i + j **for** j **in** range(m)] **for** i **in** range(n)]

Для n=3, m=4 этот генератор заполнит двумерный массив следующим образом:

0 1 2 3

1 2 3 4

2 3 4 5

**Пример 4**

a = [[(int)(i == j) **for** j **in** range(m)] **for** i **in** range(n)]

В этом генераторе выражение (i == j) равно True, если элемент расположен на диагонали, и равно False в противном случае. Если преобразовать это выражение к целочисленному типу (int)(i == j), то в сгенерированной таблице на диагонали будут стоять единицы, а в остальных ячейках будут стоять нули.

Для n=3, m=4 этот генератор заполнит двумерный массив следующим образом:

1 0 0 0

0 1 0 0

0 0 1 0