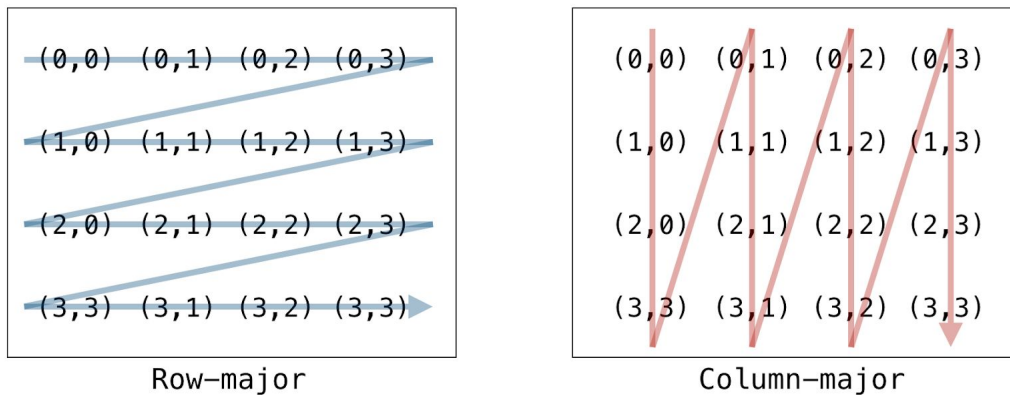


EXERCISE 1: A Grid Iterator

Download `Grid.zip` from the precepts page, unzip the project and open it using IntelliJ.

(a) Implement the `GridIterator` class in `Grid.java` to enable iterating over the elements in the grid in row-major order (as shown below). Test your program by running the given driver program.



(b) Create another iterator `ColMajorIterator` that returns elements in *column-major* order. Add code to `main` that prints the grid elements using this iterator.

(c) Convert `Grid.java` to an *Iterable*, where the default iteration is in row-major order. Test your code by converting the while loop in `main` to a for-each loop.

(d) Add code to `main` that prints `"Distinct"` if all the elements in the grid are distinct and `"Not Distinct"` if any element appears more than once.

EXERCISE 2: Memory Analysis

(a) Use tilde notation to describe as a function of n how much memory an object of type `Grid<Item>` requires right after the constructor finishes execution. Note that the grid is of size $n \times n$.

(b) Use tilde notation to describe as a function of n how much memory a `Grid<Integer>` object requires, assuming that there are no *null* items in the grid. Note that every object of type `Integer` requires 24 Bytes.