

CS100J Spring 2007. Lab 13: Implementing rectangular arrays

Introduction

The goal of this lab is to show you how a rectangular array with r rows and c columns can be implemented in a one-dimensional array. This should be a short lab. Feel free to work on assignment A7 when you have finished the lab.

Step 1. Start a new project

Create a new folder on your hard drive. Download from the lab part of the course website the three files `TestRectArray.java`, `RectArray.java`, and `JLiveRead.java`. Put them into the new folder and load them into DrJava. Execute `TestRectArray.main(null)`; to start the program. Step 4 tells you what it does.

Step 2. Mapping a two-dimensional array on a one-dimensional array

Look at class `RectArray`. An instance of this class is supposed to contain a two-dimensional array. But this array will actually be stored in one-dimensional array variable `b`, as shown below, assuming that there are r rows and c columns.

$0*c$	$1*c$	$2*c$		$(r-1)*c$
elements of row 0	elements of row 1	elements of row 2	...	elements of row $r-1$

Thus, the elements of the first row are placed first in `b`, then the elements of the second row, etc. The elements of each row are stored in ascending order of their indices —first, element 0, then element 1, etc.

This arrangement is called *row-major order*. You can see how easy it is to store the elements of a two-dimensional array in a one-dimensional array.

Step 3. Completing class `RectArray`

An instance of class `RectArray` is supposed to

implement a two-dimensional array. This class is complete, except for three places:

- (a) You have to fix the body of the constructor so that it creates an array of the right size (= the number of elements in the two-dimensional array that it creates).
- (b) You have to fix the body of method `elementAt` so that it returns the right value. For this, you have to figure out the formula for the index in array `b` of an element of a two-dimensional array element `c[i][j]`.
- (c) You have to fix the body of method `setElementAt` so that it stores the right value in the right array element.

Step 4. Testing your work

Take a look at class `TestRectArray`. It creates a two-dimensional array `d` and fills it with values. It then prints the array both in row-major order and column-major order. If you have done Step 3 correctly, you'll be able to see that it is correct by looking at this output. So, run the program and test it.

Method `main` also asks for an integer from the keyboard. This is used only to stop execution so that you can see whether you wrote the methods `elementAt` and `setElementAt` correctly. But type in a few letters and see what happens. Then look at method `JLiveRead.readLineInt` to see how the `NumberFormatException` is caught in a loop so that the user won't get further until an integer is typed in.