Declaration | Coursera

coursera.org/learn/pointers-arrays-recursion/supplement/37Ada/declaration

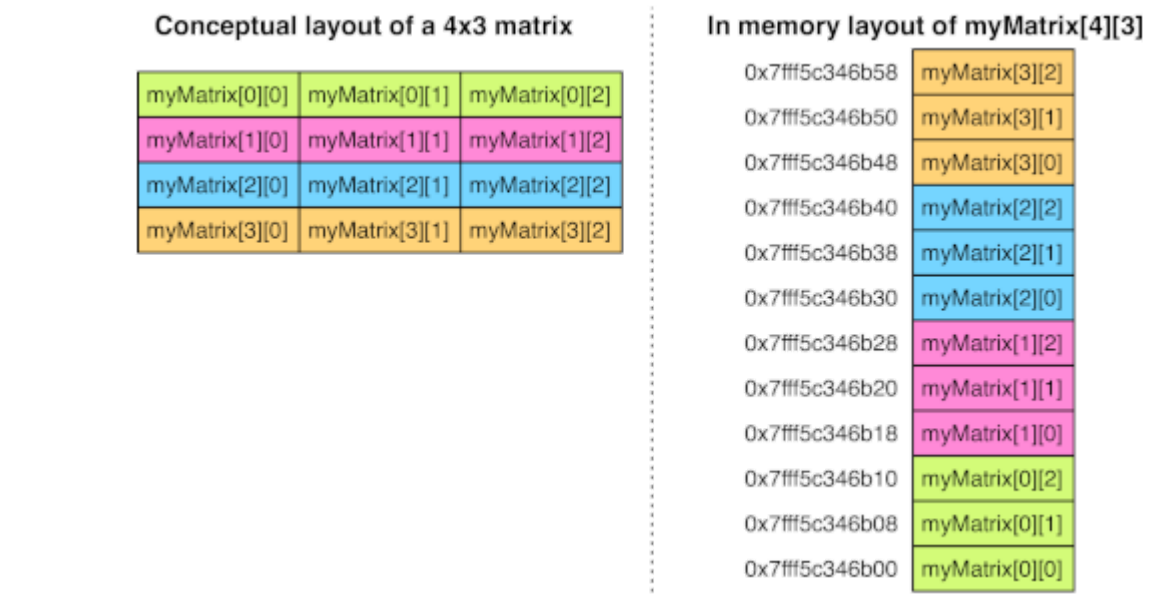
Declaration

Declaration

In C, multidimensional arrays are arrays of arrays—a two-dimensional array is an array whose elements are one-dimensional arrays; a three-dimensional array is an array whose elements are two-dimensional arrays; and so on. Accordingly, we declare them with multiple sets of square brackets, each indicating the size of the corresponding dimension. For example, we might declare a 2-dimensional array of doubles that is 4 elements by 3 elements (*e.g.*, to use as a 4x3 mathematical matrix) like this:

```
1

double myMatrix[4][3];
```



Declaring *myMatrix* in this fashion results in an array with four elements. Each of the elements of *myMatrix* is an array of 3 doubles. Accordingly, *myMatrix* occupies (4 * 3 * **sizeof**(**double**)) bytes of memory, with the three elements of *myMatrix*[*0*] appearing together, followed by the three elements of *myMatrix*[*1*], and so on. The figure above depicts this layout on the right, as well as the conceptual (*i.e.*, rectangular) layout of the matrix on the left. In both side of the figure, the 0th element ("row") of myMatrix is

colored green, the 1st is colored pink, the 2nd blue, and the 3rd orange, so that you can easily see how the data is laid out. The particular addresses are not important (and are just examples—they would change from program to program), but are intended to show how the elements are all consecutive in memory.