**2D Array Quiz**

**Part I. General Concepts** - For this part assume you have 2D array of integers called *nums*.

1. If you have an n x n 2D array (same number of rows and columns), where n is an odd integer value, write a statement to change the center element of the array to the value of 1.

XXX XXXXX

X1X XXXXX

XXX XX1XX

 XXXXX

 XXXXX

2. If I wanted to output the value stored in the top right corner of the nums array, but I didn't know the size of the array, what code would accomplish this?

3. Declare a 2D boolean array called *checks* that has x columns and y rows. Write brief code that will fill this array with alternating columns of true and false. For example, if the array size was 4 columns and 3 rows, the array should look like: TFTF

 TFTF

 TFTF

5. Write a method, *makeThemInteger,* that receives a 2D array of double values. The method will build an array of integers of the same size. The values in this array will be the double values, rounded off to the nearest integer. (.5 or more is rounded up, less than .5 is rounded down) This array of integers is returned from the method.

**Part II. Calculate** - Calculate the contents of *nums* after the following code has run.

**int**[][] nums = {{6,4,8}, {7,9,1}, {5,2,6}, {4,0,3}};

 **for**(**int** c = 1; c < nums[0].length;c++)

 **for**(**int** r = nums.length-c; r >= 0; r--)

 {

 **if**(r \* c % 3 == 0) nums[r][c]= c - r;

 **else** **if**(r > c) nums[c][r] = 5;

 }

 *outputMatrix*(nums);

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**Part III. LiveLab Program** - P\_AdjustLetters

There is a prebuilt 2D array of characters called letters. To begin with, all letters in the array are capitalized. The main method is already written and posted on the program in LiveLab. Your job is to write the *adjust* method, which will receive the letters 2D array, as well as 2 integer values from the user. The first integer value received specifies a specific row in the array and the second integer value specifies a column in the array. The method will simply change all of the letters in that column and row to lower case. If either the integer value is not valid for the array, the method should simply output "invalid input".

Your program should process the rows first, then the columns. When processing the columns, make sure you don't change the value of the column to anything but the lower case version of the letter contained in the original array.

Sample run:

3 4

Before After

ABCDEFG ABCdEFG

HIJKLMN hijklmn

OPQRSTU OPQrSTU

VWXYZAB VWXyZAB

CDEFGHI CDEfGHI

On the ASCII table, 'A' has a value of 65. 'a' has a value of 97, giving them a difference of 32 (hint).