Lab 2

Deadline: Due in Lab in the week of Jan 29, 2024

Implement a function **rotateArray** in the provided **lab2.c** skeleton file that rotates a 2D array of integers by an angle specified in input file. The rotation is in the clockwise or anti-clockwise direction based on sign of angle e.g 90 or -90.

The signature of the function is:

```
void rotateArray(int arr[10][10], int n, int rows, int cols);
```

Requirements

- The input array size is limited to 10x10.
- You may receive an input smaller than 10x10, and it will always fit within a 10x10 array.
- The input array will be a square containing only digits 1-9.
- Ensure that the rotation angle is always a multiple of 90 degrees.
- The main function reads the input from a file specified in the command line arguments and prints the rotated array to the console.

Restrictions

- You are not allowed to write any additional **printf** statement anywhere in the file.
- You are not allowed to modify any part of the code except the **rotateArray** function and the **a_num** variable.
- If you have any doubts, ask during the lab session.

Example

Here is a 5x5 example of the input file where 90 is rotation angle, 5 is dimension of array and E at the end specifies the end of file:

```
90
5
11111
11111
22222
22222
33333
E
```

Here is the output of running the input file through your program when it is rotated 90 degrees. In this case, integer n in the function parameter is 90.

```
32211
32211
32211
32211
32211
```

How to Compile and Run

- The Makefile for lab2 is provided.
- The Makefile is supposed to work with lab2.c and input.txt files so, make sure to modify the skeleton files only.
- Run the following command in vs code Terminal.
  `make`
  It should compile the code without any errors.
  `make convert`
  It should convert the input.txt file to unix encoding.
  `make run`
  It should display the rotated array.
- Run the following command to delete the out file.
  `make clean`
- You are not supposed to make any changes in the Makefile.
- Make sure to install dos2unix utility using the following command:
  `sudo apt-get install dos2unix`
  For Mac
  `brew install dos2unix`

Grading

The lab must done in Unix environment. Any grading failure due to not following instructions will result in 0.

- (1 point) All files are submitted correctly using the instructions below.
- (3 point) Generate a correct solution to the problem(s) in this lab. Three test inputs will be used.

Submission

- You must submit only one .c file named: lab2.c (case sensitive) to learning hub.
- Make sure to update your A number. Look at the top of lab2.c and write your A number including leading 0's.