```
/**
	Saket Bakshi. 4/14/19. Period 6. This is used for the Tumor Detection Lab.
	Takes in organism samples and potential tumors and sees if the samples can have the tumor, regardless of orientation of the potential tumor.
*/
public class TumorDetection
{
	private int[][] tumor, tumorRotated90, tumorRotated180, tumorRotated270;
	private double middle;
	private int tumors;

	/**
		Creates a tumor detection object.
		@param sample the sample of skin to check
	*/
	public TumorDetection(int[][] sample)
	{
		this.tumor = sample;
		middle = tumor.length / 2;
		if(tumor.length % 2 == 0)
		{
			middle -= 0.5;
		}
		tumorRotated90 = rotate(tumor);
		tumorRotated180 = rotate(tumorRotated90);
		tumorRotated270 = rotate(tumorRotated180);
		tumors = 0;
	}

	/**
		Returns the normal tumor sample.
		@return normal tumor
	*/
	public int[][] getTumor() {return tumor;}

	/**
		Returns the normal tumor sample, rotated 90 degrees.
		@return normal tumor, rotated 90 degrees
	*/
	public int[][] getTumor90() {return tumorRotated90;}

	/**
		Returns the normal tumor sample, rotated 180 degrees.
```

```
		@return normal tumor, rotated 180 degrees
	*/
	public int[][] getTumor180(){return tumorRotated180;}

	/**
		Returns the normal tumor sample, rotated 270 degrees.
		@return normal tumor, rotated 270 degrees
	*/
	public int[][] getTumor270(){return tumorRotated270;}

	/**
		Rotates a tumor array 90 degrees.
		@param tempTumor the tumor to rotate
		@return the rotated tumor
	*/
	private int[][] rotate(int[][] tempTumor)
	{
		int[][] result = new int[tempTumor.length][tempTumor.length];
		for(int i = 0; i < tempTumor.length; i++)
		{
			for(int j = 0; j < tempTumor.length; j++)
			{
				double xDistance = j - middle;
				double yDistance = i - middle;
				if(xDistance < 0 && yDistance > 0)
					result[(int) (middle + xDistance)][(int) (middle - yDistance)]
= tempTumor[i][j];
				else if(xDistance > 0 && yDistance > 0)
					result[(int) (middle + xDistance)][(int) (middle - yDistance)]
= tempTumor[i][j];
				else if(xDistance > 0 && yDistance < 0)
					result[(int) (middle + xDistance)][(int) (middle - yDistance)]
= tempTumor[i][j];
				else if(xDistance < 0 && yDistance < 0)
					result[(int) (middle + xDistance)][(int) (middle - yDistance)]
= tempTumor[i][j];
				else if((xDistance == 0) && (yDistance != 0))
					result[(int) middle][(int) (middle - yDistance)] =
tempTumor[i][j];
				else if((xDistance != 0) && (yDistance == 0))
					result[(int) (middle + xDistance)][(int) middle] =
tempTumor[i][j];
				else
```

```
                                        result[(int) middle][(int) middle] = tempTumor[i][j];
                        }
                }
                return result;
        }

        /**
                Prints using System.out the possible tumor locations.
                @param scan the skin sample to scan for
        */
        public void checkScan(int[][] scan)
        {
                tumors = 0;
                for(int i = 0; i <= (scan.length - tumor.length); i++)
                {
                        for(int j = 0; j <= (scan.length - tumor.length); j++)
                        {
                                int[][] possibleTumor = scanPartition(scan, i, j);
                                if(matching(possibleTumor, tumor))
                                {
                                        System.out.println("Possible tumor at (" + i + ", " + j + ", " +
0 + ")");
                                        tumors++;
                                }
                                else if(matching(possibleTumor, tumorRotated90))
                                {
                                        System.out.println("Possible tumor at (" + i + ", " + j + ", " +
270 + ")");
                                        tumors++;
                                }
                                else if(matching(possibleTumor, tumorRotated180))
                                {
                                        System.out.println("Possible tumor at (" + i + ", " + j + ", " +
180 + ")");
                                        tumors++;
                                }
                                else if(matching(possibleTumor, tumorRotated270))
                                {
                                        System.out.println("Possible tumor at (" + i + ", " + j + ", " +
90 + ")");
                                        tumors++;
                                }
                        }
```

```
            }
            if(tumors == 0)
                System.out.println("No tumors detected.");
        }

        /**
            Offsets an array.
            @param array the original array
            @param i offsets by i
            @param j offsets by j
            @return the new array
        */
        private int[][] scanPartition(int[][] array, int i, int j)
        {
            int[][] tempArray = new int[tumor.length][tumor.length];
            for(int a = 0; a < tempArray.length; a++)
            {
                for(int b = 0; b < tempArray.length; b++)
                    tempArray[a][b] = array[i+a][j+b];
            }
            return tempArray;
        }

        /**
            Sees if a skin sample matches a tumor.
            @param scan the sample to check
            @param tempTumor the tumor samples
            @return if a possible tumor is found
        */
        private static boolean matching(int[][] scan, int[][] tempTumor)
        {
            for(int i = 0; i < scan.length; i++)
            {
                for(int j = 0; j < scan.length; j++)
                {
                    if(scan[i][j] != tempTumor[i][j])
                        return false;
                }
            }
            return true;
        }
}
```

```
/**
	Saket Bakshi. 4/14/19. Period 6. This is used for the Tumor Detection Lab.
	Tests the TumorDetection class.
*/
import java.util.ArrayList;
import java.util.Scanner;
import java.io.File;
import java.io.FileNotFoundException;
public class TumorTester
{
	public static void main(String[] args) throws FileNotFoundException
	{
		File inFile = new File("input.txt");
		Scanner input = new Scanner(inFile);
		input.useDelimiter("");
		ArrayList<Integer> tempScan = new ArrayList<>();
		ArrayList<Integer> tempTumor = new ArrayList<>();

		while(input.hasNextInt())
			tempScan.add(input.nextInt());

		int[][] scan = new int[tempScan.size()][tempScan.size()];

		// Fills in the skin sample
		for(int j = 0; j < tempScan.size(); j++)
			scan[0][j] = tempScan.get(j);

		for(int i = 1; i < tempScan.size(); i++)
		{
			input.nextLine();
			for(int j = 0; j < tempScan.size(); j++)
				scan[i][j] = input.nextInt();
		}

		input.nextLine();

		// Determines the size of the tumor
		while(input.hasNextInt())
			tempTumor.add(input.nextInt());

		int[][] tumor = new int[tempTumor.size()][tempTumor.size()];

		// Fills in tumor object
```

```
		for(int j = 0; j < tempTumor.size(); j++)
			tumor[0][j] = tempTumor.get(j);
		for(int i = 1; i < tempTumor.size(); i++)
		{
			input.nextLine();
			for(int j = 0; j < tempTumor.size(); j++)
				tumor[i][j] = input.nextInt();
		}

		TumorDetection tester = new TumorDetection(tumor);
		tester.checkScan(scan);
	}

	public static void print2DArray(int[][] array)
	{
		for(int i = 0; i < array.length; i++)
		{
			for(int j = 0; j < array.length; j++)
				System.out.print(array[i][j]);

			System.out.println();
		}
	}
}

/**
	Saket Bakshi. 4/14/19. Period 6. This is used for the Tumor Detection Lab.
	Tests the TumorDetection class.
*/
import java.util.ArrayList;
import java.util.Scanner;
import java.io.File;
import java.io.FileNotFoundException;
public class Main
{
	public static void main(String[] args) throws FileNotFoundException
	{
		File inFile = new File("input.txt");
		Scanner input = new Scanner(inFile);
		input.useDelimiter("");
		ArrayList<Integer> tempScan = new ArrayList<>();
		ArrayList<Integer> tempTumor = new ArrayList<>();
```

```
		//determine scan size
		while(input.hasNextInt())
			tempScan.add(input.nextInt());

		int[][] scan = new int[tempScan.size()][tempScan.size()];

		//fill scan
		for(int j = 0; j < tempScan.size(); j++)
			scan[0][j] = tempScan.get(j);

		for(int i = 1; i < tempScan.size(); i++)
		{
			input.nextLine();
			for(int j = 0; j < tempScan.size(); j++)
				scan[i][j] = input.nextInt();
		}

		input.nextLine();
		//determine tumor size
		while(input.hasNextInt())
			tempTumor.add(input.nextInt());

		int[][] tumor = new int[tempTumor.size()][tempTumor.size()];

		//fill tumor
		for(int j = 0; j < tempTumor.size(); j++)
			tumor[0][j] = tempTumor.get(j);
		for(int i = 1; i < tempTumor.size(); i++)
		{
			input.nextLine();
			for(int j = 0; j < tempTumor.size(); j++)
				tumor[i][j] = input.nextInt();
		}

		TumorDetection tester = new TumorDetection(tumor);
		tester.checkScan(scan);
	}

	public static void print2DArray(int[][] array)
	{
		for(int i = 0; i < array.length; i++)
		{
			for(int j = 0; j < array.length; j++)
```

```
                System.out.print(array[i][j]);

            System.out.println();
        }
    }
}
```

```
PS C:\Users\saket\Git\CSWork\JAVA\Labs\TumorDetectionP6BakshiSaket> java Main
Possible tumor at (3, 5, 0)
PS C:\Users\saket\Git\CSWork\JAVA\Labs\TumorDetectionP6BakshiSaket>
```

I TRIED TO GET AUTOGRADR ALL IN ONE SCREENSHOT BUT I'D ALWAYS GET TIMED OUT. AFTER 45 MINUTES, I STOPPED TRYING BECAUSE I WAS WASTING TIME.