## **CECS 277 – Lab 4**

## **Drawing Rectangles**

Create a program that draws a rectangle on a grid that starts at (0,0) and then moves around the screen based on inputs from the user.

Create a Rect class (in its own file: Rect.java) with the following properties:

Instance Variables	<u>Methods</u>
1. x	1. Constructor – pass in w and h, sets width and height.
2. y	2. getX and getY – returns x and y value respectively.
3. width	3. getWidth and getHeight – returns the width and height.
4. height	4. translate – pass in dx and dy, then add to x and y.

In a separate file (Main.java), create a Main class and a main method. Then create a 20x20 2D array of characters (the grid), then prompt the user to enter the width and height of the rectangle (1-5). Construct a Rect object using the width and height entered.

Create the following methods:

- 1. displayGrid pass in the grid, then display the contents of the grid.
- 2. placeRect pass in the grid and the rectangle. Go to the location in the grid at the rectangle's x and y location, then, using '\*'s, fill in the grid using the rectangle's width and height to draw the rectangle on the grid.
- 3. resetGrid pass in the 2D array, and fill the grid with '.'s.
- 4. menu display the menu below, prompt the user for their input and return it.

Reset the grid, place the rectangle on it, display the grid, and then display the menu. Allow the user to choose which direction to move the rectangle. Do not allow the user to move any part of the rectangle outside of the bounds of the grid. If an edge of the rectangle reaches a wall, then it should not be able to move any farther in that direction. Repeat until the user decides to quit the program.

You can use the CheckInput class functions to make sure that all user inputs are valid and within the bounds of the input.

Make sure you're using the Rect class that you created for all rectangles constructed in your program (not the Rectangle class in java.awt). Document all methods using Javadocs style comments.

## **Example Output:**

						_		wi	Ldt	h	( -	L – 5	5) 3	? v	V				
ΤΙ.	lVa	ГТТ	_a	ΤΙ.	ıρι	ıt.													
Er	ite	er	re	ect	ar	ngl	.е	W	Ĺdt	:h	( _	L – 5	5) [	? 3	3				
En	ite	er	re	ect	ar	ngl	.e	h€	eig	ght	_	(1-	-5)	:	2				
*	*	*																	
*	*	*					_												_
			•	•	•	•	٠	•	•	•	•	•	•	•	•	•	•	•	٠
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
-					-	-				,					,	,			
•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•

You cannot go that way. 2																		
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2		•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2		•	•	•	•		•	•	•	•	•	•	•	•	•	•	•	•
Down Left Right Quit Cou cannot go that way.				•	•		•			•	•	•	•	•	•	•	•	
. Up . Down . Left . Right . Quit ou cannot go that way				•	•		•			•	•	•		•	•	•	•	
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2	•			•	•		•			•	•	•		•	•	•	•	
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2	•			•	•		•			•	•	•		•	•	•	•	
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2	•			•	•		•			•	•	•		•	•	•	•	•
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•	•
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2	•	•	•	•	•		•	•	•	•	•	•	•	•	•	•	•	•
L. Up  2. Down  3. Left  4. Right  5. Quit  4. Vou cannot go that way.  2																		•
2. Down 3. Left 4. Right 5. Quit L You cannot go that way. 2	Εr	nte	er	d:	ire	ect	cio	on	:									
2 · · · · · · · · · · · · · · · · · · ·	5.	. (	Qu i	Ĺt														
* * *		ou	Cá	anr	not	to	30	tŀ	nat	tι	var							
	1 Yo 2	ou	Cá	anı	not	t q	go	tl	nat	tτ	va <u>s</u>	y •						
	Y c 2		Сā	anr	not	t (	90	th	nat	t 1	wa <u>y</u>	<i>Y</i> •	•	•	•			
	Y c 2 •	•		anr •	not	t (		th .	nat	t 1	va <u>s</u>							
	Y c 2 •	•	•	anr	not	t (		th	na1	t 1	• • • • • • • • • • • • • • • • • • •				•			
	Y c 2 •	•	•	anr • •				th	nat		• • • • • • • • • • • • • • • • • • •							
	Y c 2 •	•	•	• • •				t1	nat		• • • • • • • • • • • • • • • • • • •							
	Y c 2 •	•	•	• • • • • • • • • • • • • • • • • • •				th	na1		• • • • • • • • • • • • • • • • • • •							
	Y c 2 •	•	•					th	nat		• • • • • • • • • • • • • • • • • • •							
	Y ( 2 •	•	•					th	na1									
	Y c 2 •	•	•					th	na1	· · · · · · · · · · · · · · · · · · ·	wa <u>y</u>							
	Y ( 2 •	•	•					th		· · · · · · · · · · · · · · · · · · ·	wa <u>y</u>							
	Y c 2 •	•	•					tl		· · · · · · · · · · · · · · · · · · ·	wa <u>y</u>							
	Y ( 2 •	•	•					tl	nat		wa <u>y</u> 							
	Y ( 2 •	•	•					tl	nat		wa <u>y</u>							
	Y c 2 •	•	•					tl	nat		wa <u>y</u>							
	Y ( 2 •	•	•					tl	na1	· · · · · · · · · · · · · · · · · · ·	wa <u>y</u>							
	Y ( 2 •	•	•				· · · · · · · · · · · · · · · · · · ·	tl	na1	· · · · · · · · · · · · · · · · · · ·	wa <u>r</u>							