Lab 4 – Basic Rectangles involving Malloc

Learning Goals

1) Develop your ability to use malloc to request memory on the heap.
2) Develop your ability to use “struct” and “typedef” to define memory structures.
3) Develop your ability to use pointers and their dereferencing notations.

Your Task

Implement **rect.c**.

Note: The interface **rect.h** has been provided for you. In it, the structs have been defined in **rect.h**. You do not need to define any structs.

The test program **rectTest.c** has also been provided for you.

Note that both **createRect** and **copyRect** create new rectangles. Therefore, both are required to call malloc. **copyRect** performs what is commonly referred to as a deep copy.

Sample Outputs

Your output should match the following exactly.

```
R1 : Rectangle: (7.7,9.2); (0.4,12.4)
R1 Area = 23.4
R1 Perimeter = 21.0

R2 : Rectangle: (15.6,14.4); (-6.3,24.0)
R2 Area = 210.2
R2 Perimeter = 63.0

R3 : Rectangle: (11.1,18.6); (16.8,3.9)
R3 Width = 5.7
R3 Height = 14.7
R3 Area = 83.8
R3 Perimeter = 40.8
```

Grading

This assignment will be scored out of 100 points.

80 pts – One point each for correctly implemented each of the following: createRect, copyRect, move, dilate, getWidth, getHeight, getArea, getPerimeter.

20 pts – For correctly implementing toString and getting the output to match exactly as shown.