Hi there,

Please see below some questions for you to do

- Think well on your answer and try to build a clean elegant solution and code
- Make sure your code is tested and provide the relevant unit tests with it
- You can send the project via mail or, better, send a link to a github repository
- Just skip any questions that are in an unfamiliar area
- Enjoy!

1. Given the following interface “RWNode”:

```
public interface RWNode {
      public String getName();
      public RWNode getNext();
}
```

RWNode can be thought of as defining a list. Assume RWNode list is null terminated (i.e. that getNext() invoked on the last RWNode in a list will return null)

Implement a function with the following signature:

```
public void printList(RWNode node);
```

Which prints out the names of all of the RWNodes in the node list defined by the 'node' argument. The idea here is to walk the list, printing each node name as you encounter it.

2. Implement a function with the following signature:

```
public void printListReverse (RWNode node);
```

which behaves the same as printList except that it prints the names of the nodes in reverse. This time, do not use any looping constructs (i.e., for, while, etc loops are not allowed). Also, this function should only visit each RWNode once.

3. There are some arrays that I like very much. These are arrays that have some “pivot” index, so that if you sum all the elements to the right of this pivot, and you sum all the elements to the left of the pivot, you get the same result.

Here is an example of such an array:

5	9	7	17	6	5	4	6
Sum = 21			Pivot	Sum = 21			

Implement a function that receives an array and returns true if it meets the criteria and false if it doesn’t. Make sure your code is as efficient as possible in terms of time complexity.