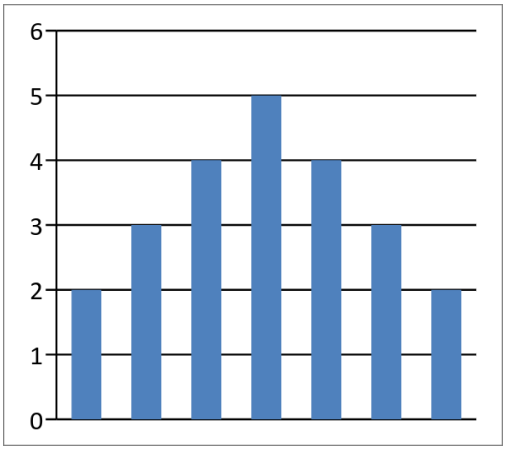
Recursion – "Mountain List"

Lab Description : Use recursion to find the sum of numbers in a "Mountain List."



A Mountain List is a set of numbers with a lower bound and an upper bound, such that the list begins at the lower bound, increases by one unit until it reaches the upper bound, and then returns to the lower bound.

In the example to the right, the list is [2, 3, 4, 5, 4, 3, 2], giving a sum of 2 + 3 + 4 + 5 + 4 + 3 + 2 = 23.

Write the helper method that will recursively find the sum of any Mountain List, given the lower and upper bounds as parameters. The method should not use any multiplication to calculate the sum, but should instead add the numbers one at a time starting at either the front or the back of the mountain.

The main recursive call is provided below.
Decide how you want to expand the parameter list in your helper method.
Remember: this must be a recursive method – iterative methods will not receive credit.

```
// precondition:  lower and upper are both positive integers,
//                and lower < upper
public static int sum(int lower, int upper){

}
```

```
// you should use the testers below
public static void main(String[] args) {
System.out.println(sum(2,5));   // the example shown above  - result is 23
System.out.println(sum(10,15)); // result is 135
System.out.println(sum(2,35));  // result is 1223
System.out.println(sum(1,500)); // result is 250000
}
```