CountPairs specs

```
countpairs
We'll say that a "pair" in a string is two instances of a char
separated by a char. So "AxA" the A's make a pair. Pair's can overlap,
so "AxAxA" contains 3 pairs -- 2 for A and 1 for x. Recursively
compute the number of pairs in the given string.


a pair is two instances of a char separated by a char

countPairs("A")) = 0
countPairs("xA")) = 0
countPairs("AxA")) = 1
countPairs("AxxA")) = 0
countPairs("Ax0xA")) = 1
countPairs("AxAxA")) = 3
countPairs("xAxA")) = 2
countPairs("aaaa")) = 2

first find an integer n that identifies the problem.  For example: in
hanoi the n was the numnber of disks. in the addarray problem it was
the current index in the array.  what would be a good n for this problem?

Express the solution of the the problem for n using the solution for n-1.

Provide a solution for the base case.  That could be for the smallest
n you want to consider.

We'll say that a "pair" in a string is two instances of a char
separated by a char. So "AxA" the A's make a pair. Pair's can overlap,
so "AxAxA" contains 3 pairs -- 2 for A and 1 for x. Recursively
compute the number of pairs in the given string.


step1. find an n that defines the problem.  for example in hanoi the n is
the number of disks, in factorial the n is the number of which we are
computing the factorial.

step2. assume that we have a solution for n-1

step3. put the solution for n in terms of n-1

step4. solve the problem for the base case (e.g. n = 1)


lets apply the steps to the factorial.

here is the factorial function:

f(n) = n * n-1 * n-2 * n-3 ..... 1
f(2) = 2
f(3) = 6
f(4) = 24 ....


step 1: the n is the n from f(n)

step2: i have the solution for f(n-1)

step3: f(n) = n * f(n-1)

step4: f(1) = 1  this is the base case
```

```
static int f(int n) {
   if (n == 1)
       return 1;
   else
       return n * f(n-1);

}

for the pairs problem:

step1: the n is goung to be the length of the string

step2: if length str is n-1 then countPairs(str) returns the correct nr of pairs

step3: if I had the solution for a str of length n-1 could I use it
to build the solution for n

      given a String str of length n
      firstChar = str.charAt(0)
      strMinusTheFirstChar = str.substring(1)

      what is the relationship between    A  and   xAxAx
      A is equal to the second char of the strMinusTheFirstChar

      secondCharofShortStr = strMinusTheFirstChar.charAt(1)

      can I express the solution of countPairs("AxAxAx") in terms of countPairs("xAxAx")
      let pp =  countPairs(strMinusTheFirstChar)

      if firstChar == secondCharofShortStr then return pp + 1
      else
         return pp


step4: base case: length is 2

       if length == 2 return 0
```