遇到的问题：

两个等长的数列a、b，要把所有可能的映射关系都得到

并把他们存在一个hashmap的数组里面

第一反应是写一个BFS遍历这个树，

一个选择树，选择条件是 选一个还没用到的数组中的元素

第一层代表（depth=1）a中的第一个元素的选择，依次类推

但是问题出现在怎么实现： 选择没用到的元素

反正很复杂啊，然后还得写javadoc，不想创建太多的类和方法了

直接一个递归方法算了吧

怎么递归呢：

递归弄清楚几个东西就好了

结束条件

传递条件

递归式

一个一个整理：

结束条件：设b长度为l，当选择了不重复的l的时候，就可以结束了，即可选择的元素为空

传递条件：传递的为 需要递归处理的数组，已经选择了的序列

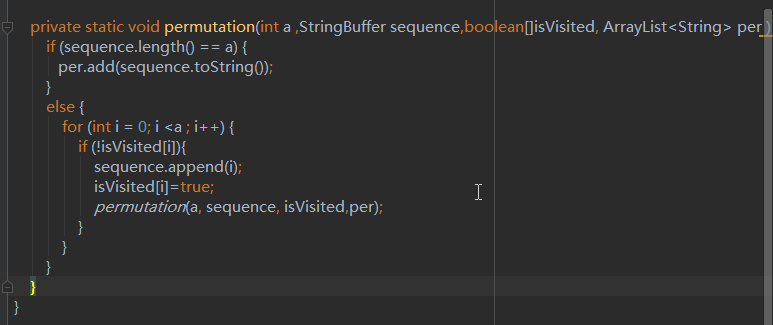
递归式: {f(0)=0,f(n)=f(n-1)+1(v属于f(n)-f(n-1))

F(t)表示长度为t的序列的长度（好像是个废话。）

真递归式：  
f(n)=n\*f（n-1）

f(t)表示包含的情况

}



问题：判断是否访问没有达到想要的效果，以及传递参数问题太大了。

怎么不能简单的把他们当参数传递，这样它们的作用域将是整个递归。

问题很大的

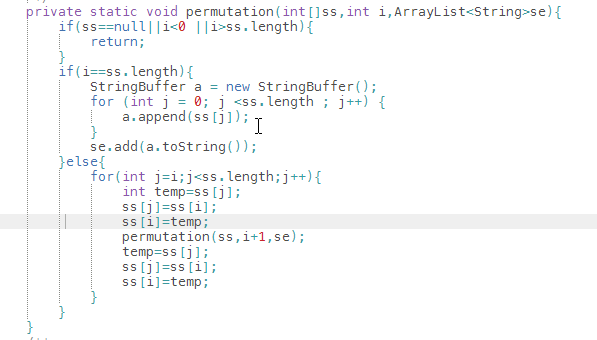
改不下去了。

三天后：  
之前的想法并没有得到具体的实现，

然后刚好在学分治，

决定用分治的方法递归实现它

分治的最低端就是两个元素的排列，只需要简单的把顺序交换就好了



这个时候，终止条件情况之外或者：设b长度为l，当选择了不重复的l的时候，就可以结束了，即可选择的元素为空

传递条件为分治

递归式为母问题=子+子