Recursion

递归

问题：

序列有个成员，每个成员可以选取这种值。

例如当，时，序列有如下排列组合：

，，，…

求的所有排列组合。（与本节的BruteForce问题一样）

解法：

本节中BruteForce存在一个问题，在BruteForce函数的代码中，外围for循环的数量是固定的：

void BruteForce(int s[4], int m) {

for (int i = 0; i < m; i++)

for (int j = 0; j < m; j++)

for (int p = 0; p < m; p++)

for (int q = 0; q < m; q++) {

s[0] = i;

s[1] = j;

s[2] = p;

s[3] = q;

Output(s);

}

}

在上面的代码中，共4层嵌套的for循环。若值改变，则for循环也必须改变。显然BruteForce的代码不能适应动态变化的值，只能满足动态变化的值。因此我们用递归方程来解决这个问题。

第1轮操作中，将序列初始化为长度为1的数组。对于唯一的元素它有种选择：

...

第2轮操作中，将序列的长度加1，得到数组，元素的选择可以看作在第1轮的每个选择的基础上继续选择。对于可以得到种选择：

...

因此第2轮操作后共有种选择。这样重复次操作，可以得到个结果。

实际编写代码中，在递归方程中传入一个参数，从0开始，序列中的成员可以取值，然后，继续考虑序列中的下一个成员。这样直到当个成员都选择了一个值时，即产生序列的一种排列组合。通过递归可以退回上一个函数栈，从而让每个成员都可以重新选择。

对于成员数量为，每个成员有种值的序列，遍历所有排列组合的时间复杂度。