Recursion

递归

问题：

序列s有n个成员，每个成员可以选取这m种值。

例如当，时，序列s有如下排列组合：

，，，…

遍历序列s的可能排列组合的所有情况。（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[MAX], int n, int m)

{

for (int i\_0 = 0; i\_0 < m; i\_0++)

for (int i\_1 = 0; i\_1 < m; i\_1++)

for (int i\_2 = 0; i\_2 < m; i\_2++)

/\* ... \*/

for (int i\_n\_1 = 0; i\_n\_1 < m; i\_n\_1++) {

s[0] = i\_0;

s[1] = i\_1;

s[2] = i\_2;

/\* ... \*/

s[n - 1] = i\_n\_1;

BruteForceOutput(s, 0, n);

}

}

在上面的代码中，共4层嵌套的for循环。若n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n值，只能满足动态变化的m值。

考虑一个下标，从0开始，序列s中的成员可以取值，然后，继续考虑序列s中的下一个成员。这样直到当n个成员都选择了一个值时，即产生序列s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n，每个成员有m种值的序列s，遍历所有排列组合的时间复杂度。