Noip2005 4.火星人

(martian.pas/dpr/c/cpp)

【问题描述】

人类终于登上了火星的土地并且见到了神秘的火星人。人类和火星人都无法理解对方的语言，但是我们的科学家发明了一种用数字交流的方法。这种交流方法是这样的，首先，火星人把一个非常大的数字告诉人类科学家，科学家破解这个数字的含义后，再把一个很小的数字加到这个大数上面，把结果告诉火星人，作为人类的回答。

火星人用一种非常简单的方式来表示数字——掰手指。火星人只有一只手，但这只手上有成千上万的手指，这些手指排成一列，分别编号为1，2，3……。火星人的任意两根手指都能随意交换位置，他们就是通过这方法计数的。

一个火星人用一个人类的手演示了如何用手指计数。如果把五根手指——拇指、食指、中指、无名指和小指分别编号为1，2，3，4和5，当它们按正常顺序排列时，形成了5位数12345，当你交换无名指和小指的位置时，会形成5位数12354，当你把五个手指的顺序完全颠倒时，会形成54321，在所有能够形成的120个5位数中，12345最小，它表示1；12354第二小，它表示2；54321最大，它表示120。下表展示了只有3根手指时能够形成的6个3位数和它们代表的数字：

三进制数

123

132

213

231

312

321

代表的数字

1

2

3

4

5

6

现在你有幸成为了第一个和火星人交流的地球人。一个火星人会让你看他的手指，科学家会告诉你要加上去的很小的数。你的任务是，把火星人用手指表示的数与科学家告诉你的数相加，并根据相加的结果改变火星人手指的排列顺序。输入数据保证这个结果不会超出火星人手指能表示的范围。

【输入文件】

输入文件martian.in包括三行，第一行有一个正整数N，表示火星人手指的数目（1 <= N <= 10000）。第二行是一个正整数M，表示要加上去的小整数（1 <= M <= 100）。下一行是1到N这N个整数的一个排列，用空格隔开，表示火星人手指的排列顺序。

【输出文件】

输出文件martian.out只有一行，这一行含有N个整数，表示改变后的火星人手指的排列顺序。每两个相邻的数中间用一个空格分开，不能有多余的空格。

【样例输入】

5

3

1 2 3 4 5

【样例输出】

1 2 4 5 3

【数据规模】

对于30%的数据，N<=15；

对于60%的数据，N<=50；

对于全部的数据，N<=10000；

方法：排列生成法，直接从指定序列用排列产生方法顺序生成到后面M个。

程序：

Noip2006 **3.Jam的计数法**

**（count.pas/c/cpp）**

【问题描述】

Jam是个喜欢标新立异的科学怪人。他不使用阿拉伯数字计数，而是使用小写英文字母计数，他觉得这样做，会使世界更加丰富多彩。在他的计数法中，每个数字的位数都是相同的（使用相同个数的字母），英文字母按原先的顺序，排在前面的字母小于排在它后面的字母。我们把这样的“数字”称为Jam数字。在Jam数字中，每个字母互不相同，而且从左到右是严格递增的。每次，Jam还指定使用字母的范围，例如，从2到10，表示只能使用{b,c,d,e,f,g,h,i,j}这些字母。如果再规定位数为5，那么，紧接在Jam数字“bdfij”之后的数字应该是“bdghi”。（如果我们用U、V依次表示Jam数字“bdfij”与“bdghi”，则U<V< span>，且不存在Jam数字P，使U<P<V< span>）。你的任务是：对于从文件读入的一个Jam数字，按顺序输出紧接在后面的5个Jam数字，如果后面没有那么多Jam数字，那么有几个就输出几个。

【输入文件】

输入文件counting.in 有2行，第1行为3个正整数，用一个空格隔开：

s t w

（其中s为所使用的最小的字母的序号，t为所使用的最大的字母的序号。w为数字的位数，这3个数满足：1≤s<T< span>≤26, 2≤w≤t-s ）

第2行为具有w个小写字母的字符串，为一个符合要求的Jam数字。

所给的数据都是正确的，不必验证。

【输出文件】

输出文件counting.out 最多为5行，为紧接在输入的Jam数字后面的5个Jam数字，如果后面没有那么多Jam数字，那么有几个就输出几个。每行只输出一个Jam数字，是由w个小写字母组成的字符串，不要有多余的空格。

【输入样例】

  2 10 5

  bdfij

【输出样例】

bdghi

bdghj

bdgij

bdhij

befgh

*题二： 选数*

**[问题描述]**：

已知n（1<=n<=20）个整数x1,x2,…,xn（1<=xi<=5000000），以及一个整数k（k<n）。从n个整数中任选k个整数相加，可分别得到一系列的和。现在，要求你计算出和为素数共有多少种。

**[问题分析]**：

本题动态规划无从下手，也无数学公式可寻，看来只能搜索（组合的生成算法），其实1<=n<=20这个约束条件也暗示我们本题搜索是有希望的，组合的生成可用简单的DFS来实现，既搜索这k个整数在原数列中的位置，由于组合不同于排列，与这k个数的排列顺序无关，所以我们可以令**a[I]<a[I+1]**（a[I]表示第I个数在原数列中的位置），这个组合生成算法的复杂度大约为**C(n,k)**，下面给出递归搜索算法的框架：

|  |
| --- |
| Proc Search(dep)  Begin  for i <- a[dep - 1] + 1 to N - (M - dep) do  1：a[dep] <- i  2：S <- S + x[i]  3：if dep < k then Search(dep + 1) else 判断素数  4：S <- S - x[i]  End |

接下来的问题就是判断素数，**判断一个整数P(P>1)是否为素数最简单的方法就是看是否存在一个素数a(a<=sqrt(P))是P的约数**，如果不存在，该数就为素数，由于在此题中1<=xi<=5000000,n<=20,所以要判断的数P不会超过100000000，sqrt(p)<=10000，因此，为了加快速度，我们可以用筛选法将2…10000之间的素数保存到一个数组里（共1229个），这样速度估计将提高5~6倍。

**特别注意**：本题是要求使和为素数的情况有多少种，并不是求有多少种素数，比赛时就有很多同学胡乱判重而丢了12分；还有1不是素数，在判素数时要对1做特殊处理。

**[参考程序]**

