



第十二届全国青少年信息学奥林匹克 联赛复赛试题

(NOIP2006 普及组)

竞赛时间：2006 年 11 月 18 日 下午 1:30-4:30

试题名称	random	happy	count	sequence
目录	random	happy	count	sequence
输入文件名	random.in	happy.in	count.in	sequence.in
输出文件名	random.out	happy.out	count.out	sequence.out
试题类型	非交互式程序题	非交互式程序题	非交互式程序题	非交互式程序题
附加文件	无	无	无	无
时限	1 秒	1 秒	1 秒	1 秒

关于竞赛中不同语言使用限制的说明

一. 关于使用 Pascal 语言与编译结果的说明

1. 对于 Pascal 语言的程序，当使用 IDE 和 fpc 编译结果不一致时，以 fpc 的编译结果为准。
2. 允许使用数学库(uses math 子句)，以及 ansistring。但不允许使用编译开关（最后测试时 pascal 的范围检查开关默认关闭：{\$R-,Q-,S-}），也不支持与优化相关的选项。

二. 关于 C++语言中模板使用的限制说明

1. 允许使用的部分：

标准容器中的布尔集合，迭代器，串，流。

相关的头文件：<bitset> <iterator> <string> <iostream>

2. 禁止使用的部分：

序列：vector，list，deque

序列适配器：stack，queue，priority_queue

关联容器：map，multimap，set，multiset

拟容器：valarray

散列容器：hash_map，hash_set，hash_multimap，hash_multiset

所有的标准库算法

相关头文件：<vector> <list> <deque> <stack> <map> <set> <algorithm>



1. 明明的随机数

(random.pas/c/cpp)

【问题描述】

明明想在学校中请一些同学一起做一项问卷调查，为了实验的客观性，他先用计算机生成了 N 个 1 到 1000 之间的随机整数 ($N \leq 100$)，对于其中重复的数字，只保留一个，把其余相同的数去掉，不同的数对应着不同的学生的学号。然后再把这些数从小到大排序，按照排好的顺序去找同学做调查。请你协助明明完成“去重”与“排序”的工作。

【输入文件】

输入文件 random.in 有 2 行，第 1 行为 1 个正整数，表示所生成的随机数的个数：

N

第 2 行有 N 个用空格隔开的正整数，为所产生的随机数。

【输出文件】

输出文件 random.out 也是 2 行，第 1 行为 1 个正整数 M ，表示不相同的随机数的个数。第 2 行为 M 个用空格隔开的正整数，为从小到大排好序的不相同的随机数。

【输入样例】

10

20 40 32 67 40 20 89 300 400 15

【输出样例】

8

15 20 32 40 67 89 300 400

2. 开心的金明

(happy.pas/c/cpp)

【问题描述】

金明今天很开心，家里购置的新房就要领钥匙了，新房里有一间他自己专用的很宽敞的房间更让他高兴的是，妈妈昨天对他说：“你的房间需要购买哪些物品，怎么布置，你说了算，只要不超过 N 元钱就行”。今天一早金明就开始做预算，但是他想买的东西太多了，肯定会超过妈妈限定的 N 元。于是，他把每件物品规定了一个重要度，分为 5 等：用整数 1~5 表示，第 5 等最重要。他还从因特网上查到了每件物品的价格（都是整数元）。他希望在不超过 N 元（可以等于 N 元）的前提下，使每件物品的价格与重要度的乘积的总和最大。

设第 j 件物品的价格为 $v[j]$ ，重要度为 $w[j]$ ，共选中了 k 件物品，编号依次为 j_1, j_2, \dots, j_k ，则所求的总和为：

$$v[j_1] * w[j_1] + v[j_2] * w[j_2] + \dots + v[j_k] * w[j_k]。 \text{ (其中 } * \text{ 为乘号)}$$

请你帮助金明设计一个满足要求的购物单。

【输入文件】

输入文件 happy.in 的第 1 行，为两个正整数，用一个空格隔开：

$N \quad m$

(其中 N (< 30000) 表示总钱数， m (< 25) 为希望购买物品的个数。)

从第 2 行到第 $m+1$ 行，第 j 行给出了编号为 $j-1$ 的物品的的基本数据，每行有 2 个非负整数

$v \quad p$



(其中 v 表示该物品的价格 ($v \leq 10000$), p 表示该物品的重要度 ($1 \sim 5$))

【输出文件】

输出文件 happy.out 只有一个正整数, 为不超过总钱数的物品的价格与重要度乘积的总和的最大值 (< 100000000)。

【输入样例】

```
1000 5
800 2
400 5
300 5
400 3
200 2
```

【输出样例】

```
3900
```

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 $U < V$, 且不存在 Jam 数字 P , 使 $U < P < V$)。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: $1 \leq s < t \leq 26, 2 \leq w \leq t - s$)

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
```

4. 数列

(sequence.pas/c/cpp)

【问题描述】

给定一个正整数 k ($3 \leq k \leq 15$), 把所有 k 的方幂及所有有限个互不相等的 k 的方幂之和构成一个递增的序列, 例如, 当 $k=3$ 时, 这个序列是:

1, 3, 4, 9, 10, 12, 13, ...

(该序列实际上就是: $3^0, 3^1, 3^0+3^1, 3^2, 3^0+3^2, 3^1+3^2, 3^0+3^1+3^2, \dots$)

请你求出这个序列的第 N 项的值 (用 10 进制数表示)。

例如, 对于 $k=3, N=100$, 正确答案应该是 981。

【输入文件】

输入文件 sequence.in 只有 1 行, 为 2 个正整数, 用一个空格隔开:

k N

(k, N 的含义与上述的问题描述一致, 且 $3 \leq k \leq 15, 10 \leq N \leq 1000$)。

【输出文件】

输出文件 sequence.out 为计算结果, 是一个正整数 (在所有的测试数据中, 结果均不超过 2.1×10^9)。(整数前不要有空格和其他符号)。

【输入样例】

```
3 100
```

【输出样例】

```
981
```