**第十届全国青少年信息学奥林匹克联赛复赛试题**

**(提高组 3小时完成)**

**一、津津的储蓄计划**   
(Save.pas／dpr／c／cpp)．   
【问题描述】   
    津津的零花钱一直都是自己管理。每个月的月初妈妈给津津300元钱，津津会预算这个月的花销，并且总能做到实际花销和预算的相同。   
    为了让津津学习如何储蓄，妈妈提出，津津可以随时把整百的钱存在她那里，到了年末她会加上20％还给津津。因此津津制定了一个储蓄计划：每个月的月初，在得到妈妈给的零花钱后，如果她预计到这个月的月末手中还会有多于100元或恰好100元，她就会把整百的钱存在妈妈那里，剩余的钱留在自己手中。   
    例如11月初津津手中还有83元，妈妈给了津津300元。津津预计11月的花销是180元，那么她就会在妈妈那里存200元，自己留下183元。到了11月月末，津津手中会剩下3元钱。   
    津津发现这个储蓄计划的主要风险是，存在妈妈那里的钱在年末之前不能取出。有可能在某个月的月初，津津手中的钱加上这个月妈妈给的钱，不够这个月的原定预算。如果出现这种情况，津津将不得不在这个月省吃俭用，压缩预算。   
    现在请你根据2004年1月到12月每个月津津的预算，判断会不会出现这种情况。如果不会，计算到2004年年末，妈妈将津津平常存的钱加上20％还给津津之后，津津手中会有多少钱。   
【输入文件】   
    输入文件save.in包括12行数据，每行包含一个小于350的非负整数，分别表示1月到12月津津的预算。   
【输出文件】   
    输出文件save.out包括一行，这一行只包含一个整数。如果储蓄计划实施过程中出现某个月钱不够用的情况，输出-X，X表示出现这种情况的第一个月；否则输出到2004年年末津津手中会有多少钱。   
【样例输入1】   
290  
230  
280  
200  
300  
170  
340  
50   
90   
80   
200  
60

【样例输出1】

-7   
【样例输入2】

290   
230   
280   
200   
300   
170   
330   
50   
90   
80   
200   
60   
【样例输出2】

1580

**二、合并果子**   
(fruit.pas／dpr／c／cpp)   
【问题描述】   
    在一个果园里，多多已经将所有的果子打了下来，而且按果子的不同种类分成了不同的堆。多多决定把所有的果子合成一堆。   
    每一次合并，多多可以把两堆果子合并到一起，消耗的体力等于两堆果子的重量之和。可以看出，所有的果子经过n-1次合并之后，就只剩下一堆了。多多在合并果子时总共消耗的体力等于每次合并所耗体力之和。   
    因为还要花大力气把这些果子搬回家，所以多多在合并果子时要尽可能地节省体力。假定每个果子重量都为1，并且已知果子的种类数和每种果子的数目，你的任务是设计出合并的次序方案，使多多耗费的体力最少，并输出这个最小的体力耗费值。   
    例如有3种果子，数目依次为1，2，9。可以先将1、2堆合并，新堆数目为3，耗费体力为3。接着，将新堆与原先的第三堆合并，又得到新的堆，数目为12，耗费体力为12。所以多多总共耗费体力=3+12=15。可以证明15为最小的体力耗费值。   
【输入文件】   
    输入文件fruit.in包括两行，第一行是一个整数n(1<＝n<=10000)，表示果子的种类数。第二行包含n个整数，用空格分隔，第i个整数ai(1<＝ai<=20000)是第i种果子的数目。   
【输出文件】   
    输出文件fruit.out包括一行，这一行只包含一个整数，也就是最小的体力耗费值。输入数据保证这个值小于231。   
【样例输入】   
3   
1 2 9   
【样例输出】   
15

【数据规模】   
对于30％的数据，保证有n<=1000：   
对于50％的数据，保证有n<=5000；   
对于全部的数据，保证有n<=10000。

**三、合唱队形**  
(chorus.pas／dpr／c／cpp)  
【问题描述】  
    N位同学站成一排，音乐老师要请其中的(N-K)位同学出列，使得剩下的K位同学排成合唱队形。  
    合唱队形是指这样的一种队形：设K位同学从左到右依次编号为1，2…，K，他们的身高分别为T1，T2，…，TK，  则他们的身高满足T1<...<Ti>Ti+1>…>TK(1<=i<=K)。  
    你的任务是，已知所有N位同学的身高，计算最少需要几位同学出列，可以使得剩下的同学排成合唱队形。  
【输入文件】  
    输入文件chorus.in的第一行是一个整数N(2<=N<=100)，表示同学的总数。第一行有n个整数，用空格分隔，第i个整数Ti(130<=Ti<=230)是第i位同学的身高(厘米)。  
【输出文件】  
    输出文件chorus.out包括一行，这一行只包含一个整数，就是最少需要几位同学出列。  
【样例输入】  
8  
186 186 150 200 160 130 197 220  
【样例输出】  
4  
【数据规模】  
对于50％的数据，保证有n<=20；  
对于全部的数据，保证有n<=100。

**四、虫食算**  
(alpha.pas/dpr/c/cpp)  
【问题描述】  
    所谓虫食算，就是原先的算式中有一部分被虫子啃掉了，需要我们根据剩下的数字来判定被啃掉的字母。来看一个简单的例子：  
       43#9865#045  
    +    8468#6633  
       44445509678  
    其中#号代表被虫子啃掉的数字。根据算式，我们很容易判断：第一行的两个数字分别是5和3，第二行的数字是5。  
    现在，我们对问题做两个限制：  
    首先，我们只考虑加法的虫食算。这里的加法是N进制加法，算式中三个数都有N位，允许有前导的0。  
   其次，虫子把所有的数都啃光了，我们只知道哪些数字是相同的，我们将相同的数字用相同的字母表示，不同的数字用不同的字母表示。如果这个算式是N进制的，我们就取英文字母表午的前N个大写字母来表示这个算式中的0到N-1这N个不同的数字：但是这N个字母并不一定顺序地代表0到N-1)。输入数据保证N个字母分别至少出现一次。  
            BADC  
       +    CBDA  
            DCCC  
    上面的算式是一个4进制的算式。很显然，我们只要让ABCD分别代表0123，便可以让这个式子成立了。你的任务是，对于给定的N进制加法算式，求出N个不同的字母分别代表的数字，使得该加法算式成立。输入数据保证有且仅有一组解，  
【输入文件】  
    输入文件alpha.in包含4行。第一行有一个正整数N(N<=26)，后面的3行每行有一个由大写字母组成的字符串，分别代表两个加数以及和。这3个字符串左右两端都没有空格，从高位到低位，并且恰好有N位。  
【输出文件】  
    输出文件alpha.out包含一行。在这一行中，应当包含唯一的那组解。解是这样表示的：输出N个数字，分别表示A，B，C……所代表的数字，相邻的两个数字用一个空格隔开，不能有多余的空格。  
【样例输入】  
5  
ABCED  
BDACE  
EBBAA  
【样例输出】  
1 0 3 4 2  
【数据规模】  
对于30％的数据，保证有N<＝10；  
对于50％的数据，保证有N<＝15；  
对于全部的数据，保证有N<＝26。