2017 網際網路程式設計全國大賽
國中組決賽

- 本次比賽共 7 題，含本封面共 18 頁。
- 全部題目的輸入都來自**標準輸入**。輸入中可能包含多組輸入，以題目敘述為主。
- 全部題目的輸出皆輸出到螢幕 (**標準輸出**)。
  輸出和裁判的答案必須完全一致，英文字母大小寫不同或有多餘字元皆視為答題錯誤。
- 比賽中上傳之程式碼，使用 C 語言請用 `.c` 為副檔名；使用 C++ 語言則用 `.cpp` 為副檔名。
- 使用 `cin` 輸入速度遠慢於 `scanf` 輸入，若使用需自行承擔 Time Limit Exceeded 的風險。
- 部分題目有浮點數輸出，會採容許部分誤差的方式進行評測。一般來說「相對或絕對誤差小於 ϵ 皆視為正確」，ϵ 值以題目敘述為主。

  舉例來說，假設 $\epsilon = 10^{-6}$ 且 a 是正確答案，b 是你的答案，如果符合 $\frac{|a-b|}{\max(|a|,|b|,1)} \leq 10^{-6}$，就會被評測程式視為正確。

2017 網際網路程式設計全國大賽
輸入輸出範例

C 程式範例：

```c
#include <stdio.h>
int main()
{
    int cases;
    scanf("%d", &cases);
    for (int i = 0; i < cases; ++i)
    {
        long long a, b;
        scanf("%lld %lld", &a, &b);
        printf("%lld\n", a + b);
    }
    return 0;
}
```

C++ 程式範例：

```cpp
#include <iostream>
int main()
{
    int cases;
    std::cin >> cases;
    for (int i = 0; i < cases; ++i)
    {
        long long a, b;
        std::cin >> a >> b;
        std::cout << a + b << std::endl;
    }
    return 0;
}
```

A. 線上遊戲

Problem ID: game

瀚瀚最近迷上了線上遊戲，一天二十四小時有十六小時都在玩遊戲，這真是太不應該了。

但最近瀚瀚找到了一個絕妙的練功地點，為了讓一天二十四小時被完善的利用，身為一個會寫程式的人，瀚瀚就開發了一個外掛程式。於是，在外掛程式的幫助下，瀚瀚就可以好好練功，並且不花太多時間在上面。

人算不如天算，正當瀚瀚以為人生美滿時，迎來了一個難題。本來瀚瀚可以很巧妙地在每一單位時間恰好擊殺一隻小怪物，但這個練功地點每 X 單位時間會重生出一隻大魔王艾迪，當大魔王艾迪出現時，瀚瀚必須先花費 Y 單位時間去擊殺他，否則瀚瀚的角色就會死去，很不方便。

以第一筆範例測試資料為例（$X = 5, Y = 2$），至第一隻艾迪大魔王重生出來前，瀚瀚可以擊殺五隻小怪物，接著瀚瀚必須花費兩單位時間擊殺艾迪大魔王；在下一隻艾迪大魔王重生出來前，瀚瀚可以擊殺三隻小怪物，接著瀚瀚必須花費兩單位時間擊殺艾迪大魔王；如此持續地進行下去。

身為一個專業玩家，瀚瀚想知道在時間點 T 時，瀚瀚已經擊殺了幾隻小怪物以及幾隻大魔王艾迪，你能夠幫助他計算嗎？

Input

測試資料恰有一行，包含三個整數 T, X, Y。

- $1 \leq T \leq 10^3$
- $1 \leq Y \leq X \leq 10^3$

Output

輸出兩個以空白隔開的數字於一行，其中第一個數字為瀚瀚擊殺的小怪物數量，第二個數字為瀚瀚擊殺的艾迪大魔王數量。

Sample Input 1	Sample Output 1
12 5 2	8 2

Sample Input 2	Sample Output 2
11 5 2	8 1

Sample Input 3	Sample Output 3
10 5 2	8 1

Sample Input 4	Sample Output 4
9 5 2	7 1

Sample Input 5	Sample Output 5
8 5 2	6 1

Sample Input 6	Sample Output 6
7 5 2	5 1

Sample Input 7	Sample Output 7
6 5 2	5 0

Sample Input 8	Sample Output 8
5 5 2	5 0

Sample Input 9	Sample Output 9
4 5 2	4 0

B. 密密麻麻密碼鎖

Problem ID: locker

琪露諾，霧之湖畔最強的妖精，長久以來早已感到高手寂寞。因此她決定要環遊世界，向各地的強者發起挑戰來砥礪自己。為了保護心愛的行李箱，她使用了高強度的六位數密碼鎖，每位有 0 到 9 共十種可能，且每位數可以獨立轉動，如 9 轉一下可以變成 8 或 0。這個密碼鎖總共有 000000 到 999999 整整一百萬種可能的密碼。

但很不幸的，出發沒多久後她就忘了自己行李箱的密碼。看著密碼鎖回（ㄈㄚ）憶（ㄉㄞ）許久的她，只記得當初設定的密碼六位數均相異，而且出門的時候只有隨意稍微撥亂，所以正確的密碼很可能跟當前顯示的密碼相去不遠。

身為「⑨章算術」的作者，聰明的她想到了一個辦法：「先來試試最有可能的密碼吧！也就是所有跟當前密碼最相近且六位數均相異的密碼」，所謂的最相近也就是所需轉動次數最少的意思。假設當前密碼為 991234，則有四個只需轉動一下的可能密碼：091234、891234、901234 及 981234。

轉著轉著，琪露諾的頭就暈了。好心的妳決定寫個程式幫幫她找出最有可能的密碼列表。

Input

測試資料恰有一行，包含一個六位數字表示當前密碼鎖上顯示的密碼。

Output

請先輸出一行，包含以空白隔開的兩個整數 n, d，代表有 n 個最有可能的密碼，以及最少需要轉動 d 下。接下來請輸出一行，包含 n 個六位數字，代表最有可能的密碼列表。密碼列表中的密碼請依序由小到大輸出，密碼間恰由一個空白隔開。

Sample Input 1	Sample Output 1
991234	4 1 091234 891234 901234 981234

Sample Input 2	Sample Output 2
123456	1 0 123456

Sample Input 3	Sample Output 3
118827	4 2 018927 019827 108927 109827

C. 測試資料系統

Problem ID: shell

你知道網際網路程式設計全國大賽（NPSC）嗎？沒錯，就是你正在比的這個比賽！

每年，NPSC 的題目都是由若干位裁判分別負責出題而成的，其中，最重要也是最困難的環節就是把所有裁判產生出的測試資料整合起來。每位裁判都有自己產生測試資料的習慣以及偏好，如果沒有好好協調，肯定是無法讓每位裁判產生的測試資料好好的儲存起來。

因此，爲了讓之後 NPSC 出題的過程能更順暢，現在想請你寫一支測試資料的管理系統！

這個系統在一開始，必須提供一個檔案目錄供所有裁判使用 (/)，裁判們可以在這個檔案目錄底下，新增或刪除一個目錄、新增或刪除一個檔案，或是列出一個目錄所包含的資訊，嚴格來說，必須有以下幾個功能：

- `mkdir DIR` : 新增路徑爲 DIR 的目錄
- `rmdir DIR` : 刪除路徑爲 DIR 的目錄
- `ls DIR` L : 列出路徑爲 DIR 這個目錄下，最多 L 層的資訊。
- `touch FILE` : 新增 FILE 這個檔案
- `rm FILE` : 刪除 FILE 這個的檔案

其中，DIR 爲 /D1/D2/.../Dk 的形式；而 FILE 爲 /D1/D2/.../Dk/F 的形式。

1. 對於 mkdir 必須滿足在最一開始所提供的目錄 (/) 底下有 D1 這個目錄，且 /D1 這個目錄底下必須有 D2 這個目錄，以此類推。如果 Dk 已經存在、中途有任何一個目錄不存在、Dk 是一個檔案或是 DIR 恰爲 /，便回報錯誤；否則，便建立 Dk 這個目錄。
2. 對於 rmdir 必須滿足在最一開始所提供的目錄 (/) 底下有 D1 這個目錄，且 /D1 這個目錄底下必須有 D2 這個目錄，以此類推。如果 Dk 並不存在、中途有任何一個目錄不存在、Dk 是一個檔案、Dk 這個目錄底下還有存在其他目錄或檔案、或是 DIR 恰爲 /，便回報錯誤；否則，便刪除 Dk 這個目錄。
3. 對於 ls 必須滿足在最一開始所提供的目錄 (/) 底下有 D1 這個目錄，且 /D1 這個目錄底下必須有 D2 這個目錄，以此類推。如果 Dk 並不存在、中途有任何一個目錄不存在、或是 Dk 是一個檔案便回報錯誤；否則，請輸出 Dk 這個目錄往下最多 L 層的資訊。如果第 L 層當中有目錄，且該目錄底下還有其他目錄或檔案，請再目錄名稱後多輸出一個 '+'，請以字典序的順序依序輸出**該目錄下的目錄及檔案**，如果該目錄底下還有目錄或檔案，且還未達到 L 層，請先進入該目錄，並輸出其中的內容。舉例來說，如果 / 底下有一個

目錄 tmp ，tmp 底下有兩個目錄分別爲 tmpdir1 及 tmpdir2，tmpdir1 底下有一個檔案爲 tmpfile：

指令	輸出
ls / 0	/+
ls / 1	/ /tmp/+
ls / 2	/ /tmp/ /tmp/tmpdir1/+ /tmp/tmpdir2/
ls / 3	/ /tmp/ /tmp/tmpdir1/ /tmp/tmpdir1/tmpfile /tmp/tmpdir2/

4. 對於 touch 必須滿足在最一開始所提供的目錄 (/) 底下有 D1 這個目錄，且 /D1 這個目錄底下必須有 D2 這個目錄，以此類推。如果 Dk 不存在、中途有任何一個目錄不存在、Dk 這個目錄底下已經存在 F、或是 FILE 恰爲 /，便回報錯誤；否則，便建立 F 這個檔案。
5. 對於 rm 必須滿足在最一開始所提供的目錄 (/) 底下有 D1 這個目錄，且 /D1 這個目錄底下必須有 D2 這個目錄，以此類推。如果 Dk 並不存在、中途有任何一個目錄不存在、F 是一個目錄、Dk 這個目錄底下不存在 F 這個檔案、或是 FILE 恰爲 /，便回報錯誤；否則，便刪除 F 這個檔案。

Input

測試資料第一行包含一個整數 N，代表總共輸入的指令數。接下來 N 行，分別包含以下其中一種形式：

- mkdir DIR : 新增路徑爲 DIR 的目錄
- rmdir DIR : 刪除路徑爲 DIR 的目錄
- ls DIR L : 列出路徑爲 DIR 這個目錄下，最多 L 層的資訊。
- touch FILE : 新增 FILE 這個檔案
- rm FILE : 刪除 FILE 這個的檔案

- $1 \leq N \leq 500$
- $1 \leq |\texttt{DIR}|, |\texttt{FILE}| \leq 100$
- $1 \leq L \leq 100$
- DIR 滿足題目敘述中的格式
- FILE 滿足題目敘述中的格式
- Di, F 僅包含大小寫英文字母數字 (a-zA-Z0-9) 及點 ('.')
- $|\texttt{D}i|, |\texttt{F}| \geq 1$

Output

對於每個指令如果產生錯誤，請輸出 "ERROR" 於一行（不包含引號）。對於每個正確執行的 ls DIR L 指令，輸出相對應的內容。

Sample Input 1	Sample Output 1
8 mkdir /tmp mkdir /tmp/tmpdir1 mkdir /tmp/tmpdir2 touch /tmp/tmpdir1/tmpfile ls / 0 ls / 1 ls / 2 ls / 3	/+ / /tmp/+ / /tmp/ /tmp/tmpdir1/+ /tmp/tmpdir2/ / /tmp/ /tmp/tmpdir1/ /tmp/tmpdir1/tmpfile /tmp/tmpdir2/

Sample Input 2

```
18
touch /c
mkdir /
mkdir /a/b
mkdir /a
touch /a/b
mkdir /a
mkdir /c
rmdir /c/d
rmdir /c
rmdir /a
mkdir /a/e
touch /a/e
rm /a/e
rm /a/b
ls / 0
ls / 1
ls / 2
ls / 3
```

Sample Output 2

```
ERROR
ERROR
ERROR
ERROR
ERROR
ERROR
ERROR
ERROR
ERROR
/+
/
/a/+
/c
/
/a/
/a/e/
/c
/
/a/
/a/e/
/c
```

Sample Input 3

```
9
mkdir /npsc
mkdir /npsc/shell
touch /npsc/shell/s1.in
touch /npsc/shell/s2.in
touch /npsc/shell/s3.in
ls /npsc 0
ls /npsc/shell 0
ls /npsc/shell/s1.in 0
ls /npsc/shell 100
```

Sample Output 3

```
/npsc/+
/npsc/shell/+
ERROR
/npsc/shell/
/npsc/shell/s1.in
/npsc/shell/s2.in
/npsc/shell/s3.in
```

D. 發獎品

Problem ID: equal

全國網際網路程式設計大賽（NPSC）決賽的頒獎就快要開始了！爲了獎勵第一名隊伍出色的表現，主辦單位特地準備了 N 個獎品，準備要發給第一名的隊伍！

在還沒公佈比賽結果以前，主辦單位不可能知道獲勝的隊伍會是由一個人還是兩個人組成，因此，主辦單位準備的獎品個數一定是偶數個（N 是偶數）。如果最後獲勝的是兩個人的隊伍，其中前 $\frac{N}{2}$ 個獎品會給一個人，而後 $\frac{N}{2}$ 個獎品會給另一個人。爲了頒獎流程的順利，這些獎品之間的順序並不能任意調換。

很不幸地，主辦單位與選手的價值觀不同，並無法預知每個獎品對於選手的價值！所以有可能發生其中 $\frac{N}{2}$ 個獎品對於選手來說都相當於有無限大的價值，而另外 $\frac{N}{2}$ 個獎品對於選手來說都相當於沒有任何價值，這樣一來，便有可能發生兩個人沒辦法獲得相等價值的獎品！

現在主辦單位已經知道這 N 個獎品中若干個獎品對於選手的價值，以及一些獎品之間的等價關係。主辦單位想知道是否有可能，兩人獲得的獎品對於選手來說是獲得相等價值的。也就是說，主辦單位想知道是否存在一種方法，將未知價值的獎品賦予相對應的價值，使得兩個選手獲得的價值相等。

舉例來說，假設主辦單位準備了 6 個獎品，依序的價值爲 $2, -1, 8, -1, -2, 6$，也就是說，主辦單位已經知道第一個獎品價值 2、第三個獎品價值 8、第六個獎品價值 6，而如果價值爲負數，代表主辦單位還不知道這些獎品的價值，但相等的負數代表主辦單位已經知道這些獎品的價值會相等，也就是說，主辦單位知道第二個及第四個的價值相等（第五個獎品也可能跟第二個及第四個獎品價值相等）。

如果我們把 -1 換成 2、把 -2 換成 4，則第一個選手會拿到前 $\frac{6}{2} = 3$ 個獎品總價值爲 12，而第二個選手會拿到後 $\frac{6}{2} = 3$ 個獎品總價值爲 12，兩者相等。

需要特別注意的是，雖然主辦單位不確定一些獎品的價值，但主辦單位能確定所有獎品的價值都介於 1 到 10 之間（包含 1 及 10）。

Input

測試資料包含兩行。第一行包含一個正整數 N，代表主辦單位準備了 N 個獎品。第二行包含 N 個整數 a_i，如果 $a_i > 0$，代表主辦單位已經知道第 i 個獎品價值 a_i，否則代表主辦單位

還不知道第 i 個獎品價值多少。但如果 $a_i = a_j$ 代表主辦單位已經知道第 i 個獎品的價值會等於第 j 個獎品的價值。

- $2 \leq N \leq 100$
- N 是偶數
- $-8 \leq a_i \leq 10$，**最多只有八種未知價值的獎品**
- $a_i \neq 0$

Output

如果存在一種滿足的方法，請輸出一行包含 N 個以空格隔開的正整數，代表可行的方案，如果有多種滿足的方法，請輸出任意一種。否則，請輸出 -1 於一行。

Sample Input 1	Sample Output 1
6 2 -1 8 -1 -2 6	2 2 8 2 4 6

Sample Input 2	Sample Output 2
4 1 2 2 1	1 2 2 1

Sample Input 3	Sample Output 3
2 -1 -2	1 1

Sample Input 4	Sample Output 4
8 -4 -3 -2 -1 1 2 3 4	4 3 2 1 1 2 3 4

Sample Input 5	Sample Output 5
4 10 10 1 -1	-1

E. 乘法問題

Problem ID: multiply

殿王是個天才兒童，他在六個月大的時候就已經學會了乘法跟除法。順帶一提，他在一歲時學會寫程式，兩歲時學會反組譯並分析一個執行檔來破解自己所玩的遊戲，不過那又是另一個故事了。

在六個月大學會乘法跟除法後，他又憑著自己的優異資質了解了「倍數關係」。在學會倍數關係之後，殿王覺得只要一個數字是自己當天的幸運數字的倍數（對於任意兩個整數 x, y，如果存在一個整數 z 使得 $x = y \times z$，那麼 x 就是 y 的倍數），那這個數字就是他今天喜歡的數字。

殿王已經知道自己今天的幸運數字是 K，他的家教老師瀚瀚在他面前擺了 N 個正整數。殿王想知道，有多少種連續的數字乘起來會是自己今天喜歡的數字（也就是 K 的倍數）？

更精確的說，假設這 N 個數字依序為 $a_1, a_2, \ldots a_N$，那麼殿王想知道有多少對數字 (L, R)，滿足 $1 \le L \le R \le N$，使得第 L 個數字一直乘到第 R 個數字（$a_L \times a_{L+1} \times a_{L+2} \times \cdots \times a_R$）會是 K 的倍數。

以範例測試資料第 1 筆為例（瀚瀚擺的數字為 $4, 5, 1, 3, 10$，而 $K = 6$）。若 $(L, R) = (2, 3)$，那麼這段數字乘起來就會是 $a_2 \times a_3 = 5 \times 1 = 5$，不是 $K = 6$ 的倍數。但若 $(L, R) = (3, 5)$，那麼這段數字乘起來就會是 $1 \times 3 \times 10 = 30$，是 6 的倍數。能夠讓一段數字乘起來為 6 的倍數的 (L, R) 有 $(1, 4), (1, 5), (2, 5), (3, 5), (4, 5)$ 共 5 個，所以這筆測試資料的答案是 5。

Input

測試資料總共有 2 行。第一行包含兩個整數 N, K，依序表示瀚瀚擺在殿王面前的數字數量以及殿王今天的幸運數字。第二行包含 N 個正整數，依序為 $a_1, a_2, \ldots a_N$，即為殿王面前的 N 個正整數。

- $1 \le N \le 10^5$
- $1 \le K \le 100$
- $1 \le a_i \le 100$

Output

請輸出一行包含一個整數代表殿王想知道的答案。輸出後請記得換行。

Sample Input 1	Sample Output 1
5 6 4 5 1 3 10	5

Sample Input 2	Sample Output 2
3 8 2 2 2	1

Sample Input 3	Sample Output 3
1 5 9	0

F. 水題

Problem ID: water

桌子上放著一杯半滿的水。

樂觀主義者說：「這杯水還有一半」。

悲觀主義者說：「這杯水少了一半」。

現實主義者說：「桌上放著一杯水」。

懷疑主義者說：「杯子裝得不是水」。

理想主義者說：「這杯水能創造無限的能量」。

資本主義者說：「爲這杯水成立品牌將創造無限的財富」。

共產主義者說：「我們要共享這杯水」。

物理學家說：「杯子裡有一半是氣體、有一半是液體」。

工程學家說：「杯子做的比真正需要的大一倍」。

瀚瀚連忙把那杯水從桌上拿起，他覺得這杯水就是一杯水，沒有那麼多艱澀困難的意義。

在瀚瀚把水喝完之後，突然想到，如果這杯子能裝 N 單位的水，當裝了 K 單位的水時，樂觀主義者會說：「這杯水還有 K 單位的水」，而悲觀主義者會說：「這杯水少了 $N-K$ 單位的水」。瀚瀚想知道樂觀主義者跟悲觀主義者說出來的數字相差多少，也就是說 K 跟 $N-K$ 相差多少，可以請你幫他計算出來嗎？

Input

測試資料恰有一行包含兩個整數 N, K，代表杯子能裝 N 單位的水，目前裝了 K 單位的水。

- $1 \leq N \leq 10^6$
- $0 \leq K \leq N$

Output

請輸出一行包含一個整數代表瀚瀚想知道的答案。輸出後請記得換行。

Sample Input 1	Sample Output 1
2 1	0

Sample Input 2	Sample Output 2
3 2	1

G. 發氣球

Problem ID: balloon

今天是網際網路程式設計全國大賽（NPSC）的決賽！

比賽中有一樣不可或缺的東西，就是「氣球」。

決賽中總共有 28 支隊伍，參賽的隊伍要在 240 分鐘內面對 7 道問題。

比賽中上傳的每一份程式碼，經過評測後會有下列資訊：Run ID（這份程式碼是整個比賽中第幾個上傳的，從 1 開始編號）、Submission Team（這份程式碼是由哪一隊上傳的，隊伍被編號為 1 至 28）、Submission Time（這份程式碼是在第幾分鐘上傳的，可能的資料是整數 0 至 239）、Problem ID（這份程式碼是針對哪一題上傳的，可能的資料會是大寫字母 A 到 G）、Verdict（這份程式碼的評測結果，可能的資料為 AC、WA、TLE 跟 RTE 其中之一）。

當一支隊伍，在比賽中**首次**解開某一題時(也就是上傳程式碼至該題並得到 AC 的 Verdict)，該隊伍就能在這個時間獲得一顆代表該題的氣球。但**若該次上傳的 Submission Time 大於或等於** 180**，那麼就不會拿到任何氣球**，因為此時計分版已經封鎖了。

現在請你幫忙撰寫一個氣球伺服器，我們將給你整場比賽中共 N 個上傳的資訊，請你印出指令指示現場的工作人員發送氣球。

Input

測試資料第一行包含一個整數 N，代表決賽中總共有 N 個上傳記錄。接下來 N 行，每行都包含三個整數 A_i, B_i, C_i 和兩個字串 P_i, V_i，依序表示這個上傳的 Run ID、Submission Team、Submission Time、Problem ID 和 Verdict。

- $1 \le N \le 10^5$
- $A_i = i$，該 N 行中的 A_i 依序為 $1, 2, 3, \ldots N$
- $1 \le B_i \le 28$
- $0 \le C_i \le 239$
- $C_i \le C_j \; \forall i \le j$，該 N 行每一個 C_j 都大於或等於前面每一行的 C_i
- P_i 為 A、B、C、D、E、F、G 其中一個
- V_i 為 AC、WA、TLE、RTE 其中一個

Output

對每一個應該獲得氣球的上傳，請輸出 "Send balloon of <Problem ID> to team <Team> at time <Time>." 於一行（不包含引號），其中的 <Problem ID> 為該隊解開的題目、<Team> 為獲得氣球的隊伍、<Time> 為該上傳的時間，來表示一個發送氣球的指令。

請注意發送氣球的指令要按照 Run ID 由小到大依序印出，如果一個獲得氣球的上傳是 Run ID 2，而另一個是 Run ID 5，那麼 Run ID 2 的發送氣球的指令一定要印在比較前面。

在輸出所有發送氣球的指令後，請輸出 "Go get snacks." 於一行（不包含引號），讓工作人員知道氣球發完，可以去吃點心了。

Sample Input 1	Sample Output 1
10 1 1 1 A TLE 2 1 10 A AC 3 2 10 A AC 4 2 100 A AC 5 1 155 G AC 6 1 166 A WA 7 1 177 A RTE 8 10 179 C AC 9 11 180 C AC 10 28 239 B AC	Send balloon of A to team 1 at time 10. Send balloon of A to team 2 at time 10. Send balloon of G to team 1 at time 155. Send balloon of C to team 10 at time 179. Go get snacks.