

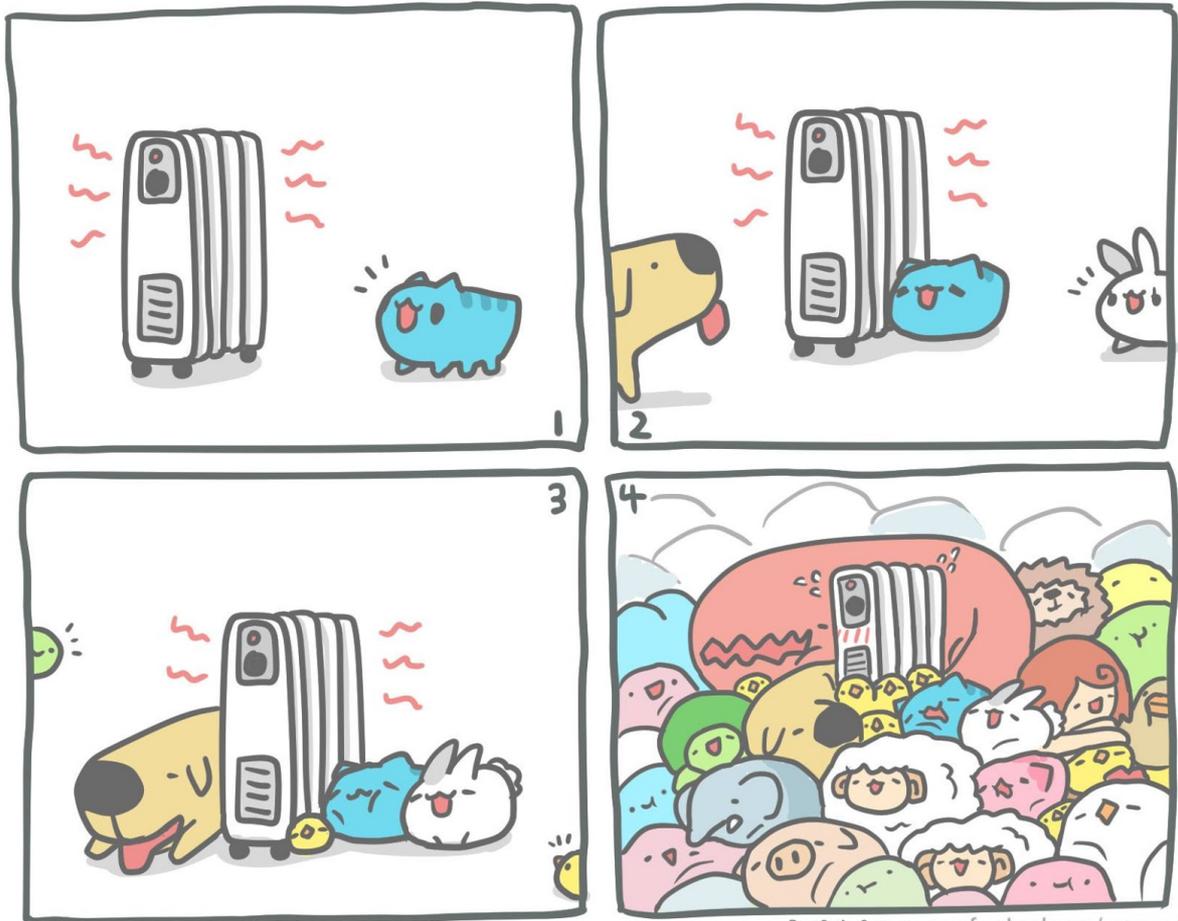
第三屆大台南高一生程式設計排名賽 賽前練習

說明與注意事項

1. 為了讓參賽者熟悉競賽的題目形式，提供3題賽前測試題目，正式競賽將有7題，作答方式與賽前測試類似。
2. 競賽採用線上自動評分，請先上網註冊練習系統：
<http://cms.tfcis.org/contests/fgc3/>，將與正式競賽使用相同系統。
3. 競賽系統使用與全國學科能力競賽/資訊奧林匹亞相同的CMS系統，評測伺服器為Ubuntu 14.04，編譯器版本為g++ 4.9.2 / gcc 4.9.2
4. 賽前測試時間：即日起至5月30日
5. 請使用C/C++程式語言作答，作答時請上傳原始程式碼 (副檔名為.cpp或.c)
6. 如對題目有疑問可使用競賽系統中的訊息詢問。

A. 電暖器

heater



BugCat-Capoo www.facebook.com/capooocat

咖波是一隻喜歡夏天的貓貓蟲，因為冬天真的太冷了。
今年年假的時候只有一臺電暖器，咖波每天都擠得很辛苦。他決定明年要買很多很多電暖器，而且要掛在天花板上才不會太擠。
但是聰明的咖波才不會讓自己吃虧呢！

在一個長寬 $N * M$ 大小的房間裡，有 K 個電暖器，掛在高度為 H 的天花板上。
已知咖波的電暖器非常神奇，周遭提升的溫度與距離平放成反比，中心提升的溫度都是 E 。

假定地板為 X - Y 平面

從 $(0,0) \sim (n,m)$

給你他們的座標 (X_i, Y_i) 、高度 H 、提升溫度 E

請找出最冷的地方在哪裡吧，咖波才不去呢！
站在牆邊的位置也算喔。

輸入說明

一開始有2個正整數 N, M
表示房間長寬
再來3個整數 K, H, E
 K = 電暖器的數量
 H = 高度
 E = 提升的溫度
再來有 N 組整數座標 X_i 與 Y_i

所有的數字之間都只以空白(' ')或換行('\n')來分隔

輸出說明

輸出一對正整數 x, y
表示房間中最冷的整數座標點
以逗號(',')分隔
例如最冷的座標是(2,3)
輸出為 "2,3" (不含雙引號)
保證最冷的地方只有一個

輸入限制

$1 \leq K \leq 500$
 $1 \leq H \leq 5$
 $1 \leq E \leq 1000$
 $1 \leq N \leq 300$
 $1 \leq M \leq 300$

子任務

| 子任務 | 分數 | 額外輸入限制 |
|-----|----|----------------|
| 1 | 10 | $N = 0, M = 0$ |
| 2 | 30 | $K = 1$ |
| 3 | 30 | $H = 1$ |
| 4 | 30 | 無 |

範例一

| INPUT | OUTPUT |
|------------------------------|--------|
| 0 0 2 10 10 0 0 0 0 | 0,0 |

範例二

| INPUT | OUTPUT |
|----------------------------|---------|
| 100 100 1 5 50 13 37 | 100,100 |

B. 咖波的責任

responsibility



年假結束了，咖波的主人也開始工作了。

咖波知道，自己有責任讓主人發現生命中有些事比工作更重要，例如陪咖波玩。

但是他一直失敗，這樣就不是一隻好的貓貓蟲了。

聰明的咖波決定趁主人不在的時候阻止她工作。但是時間有限，他必須先把搗亂用的程式寫好。

有一個字串works，代表主人的一連串工作，每項工作使用一個英文字母表示（大小寫視為不同），請你幫咖波寫一個程式支援修改其中的某一個工作和詢問指定工作出現的次數。

例如找出上班的次數，然後把其中一次改成回家。

請實作以下三個函式：

```
void Init (int length, const char *works);
```

只會被呼叫一次，而且會是第一個被呼叫的函式(Change和Ask一定會在Init呼叫後才被呼叫)

length 主人總共有多少時間，也就是字串work的長度

works 為程式給定的字串，代表主人工作

```
void Change (int index, char value);
```

把第index(由0開始編號)項工作(也就是works字串中的第index個字元)改為value

```
int Ask (char value);
```

總共有幾次value這項工作，也就是在works字串中字元value出現了幾次

限制

100%的測試資料保證:

$1 \leq \text{length} \leq 10^5$

works中的字元、value一律為英文字母

Change的呼叫次數 + Ask的呼叫次數 $\leq 2 \cdot 10^7$

50%的測試資料保證:

Change的呼叫次數 + Ask的呼叫次數 $\leq 10^5$

10%的測試資料保證:

$1 \leq \text{length} \leq 20$

Change的呼叫次數 + Ask的呼叫次數 ≤ 100

範例程式

以下是範例空白的程式模板，請你完成此3個函式，本程式以範例測試資料傳入你的函式，並且會自動檢查你對於範例測資的回答是否正確。

下載網址：<http://cms.fcis.org/contests/fgc3/responsibility.cpp>

```
#ifndef EVAL
#include <iostream>
void Init (int length, const char *works);
void Change (int index, char value);
int Ask (char value);
int main()
{
    char works[] = "abcde";
    Init(5, works);
    int ans1 = Ask('c');
    Change(2, 'a');
    int ans2 = Ask('c');
    int ans3 = Ask('a');
    if(ans1 == 1 && ans2 == 0 && ans3 == 2)
        std::cout<<"Your answers are correct!"<<std::endl;
    else
    {
        if(ans1 != 1)
            std::cout<<"Your answer to the 1st question is wrong. The
correct answer is 1, but you returns "<<ans1<<". "<<std::endl;
        if(ans2 != 0)
            std::cout<<"Your answer to the 2nd question is wrong. The
correct answer is 0, but you returns "<<ans2<<". "<<std::endl;
        if(ans3 != 2)
            std::cout<<"Your answer to the 3rd question is wrong. The
correct answer is 2, but you returns "<<ans3<<". "<<std::endl;
    }
}
```

```
}
#endif

/*=====*/
/*=====*/
/*=====*/
/*=====*/
/*Write Your Program Here*/
/*Notice: You should include header files yourself*/

void Init (int length, const char *works)
{
}
void Change (int index, char value)
{
}
int Ask (char value)
{
}
}
```

範例測試資料

以下為範例的函式呼叫順序

```
char str[] = "abcde";
Init(5, str);
Ask('c'); //應回傳1
Change(2, 'a');
Ask('c'); //應回傳0
Ask('a'); //應回傳2
```

C. 簡單的工作

work



BugCat-Capoo www.facebook.com/capooocat

咖波搗亂之後很開心，決定在旁邊等主人放棄工作。

沒想到懶惰的主人居然讓咖波代替她工作，完全在咖波的預料之外！

但是聰明的咖波才不會這樣就慌了。他決定好好研究主人的工作內容：

給一數列 $a_1, a_2, a_3, \dots, a_n$

數列的前綴 S_i 定義為 $a_1 + a_2 + a_3 + \dots + a_i$

數列的後綴 S'_i 定義為 $a_n + a_{n-1} + a_{n-2} + \dots + a_{n-i+1}$

給定一個數 T ，試問是否有一組 $S_a + S'_b = T$ ($1 \leq a, b \leq n$)

這對咖波來說太簡單了，趕快寫個程式解決之後去和主人玩吧。

輸入說明

有多筆測資以EOF作為結束

每筆測資第一行有2個數 N, Q ($N < 10^5, Q < 10^4$) 代表有 N 個數字， Q 筆詢問，

下一行有 N 個數為 $a_1, a_2, a_3, \dots, a_n$ ($a_i < 100$)

接下來有 Q 行，每行有一個數字 T ($T < 10^7$)，為查詢的數字。

輸出說明

對於每一個詢問輸出一個Y or N

範例

| INPUT | OUTPUT |
|----------------------------|--------|
| 5 2 2 1 6 3 4 7 8 | Y N |