

指標字典

Description

有一個 N 行字的字典，請用一個二階指標陣列來交換它們的位置。我們用 (i, w) 來代表第 i 行的第 w 個字。兩者都從 0 開始編號。例如在下面的範例中 $(0, 0)$ 代表 apple、 $(1, 0)$ 代表 bird。

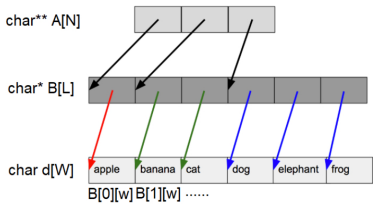
```
apple
bird cat
dog elephant frog
```

現在我們定義交換的指令。 (i_1, w_1, i_2, w_2) 代表交換 (i_1, w_1) 和 (i_2, w_2) 這兩個字的位置。注意我們不會有無效的指令。

例如如果指令為 $(0,0,2,2)$ ，那麼我們交換 apple 和 frog 的位置。指令結束後，按照順序輸出每行的內容。

```
frog
bird cat
dog elephant apple
```

一個二階指標陣列如下圖，第一個陣列有 N 個代表 N 行字，第二個指標陣列有 L 個代表共 L 個字。請注意如果不用二階指標陣列可能會造成某些測資 **TLE** 或 **MLE**。且因為字與字的長度可能差異很大，請用一個長度為 W 的一維陣列儲存所有字以節省空間。



Input

N 代表共 N 行字， L_i 代表第 i 行的字數， L 代表總共的字數， w 代表字的最長長度， W 代表總共的字元個數。

- $N < 1024$
- $L_i < 1024$
- $L < 16384$
- $w < 1024$
- $W < 10000000$

每個測資只有一組測試。第一行為 N 代表共有 N 行字。之後 N 行每行一開始有一個數字 L_i 代表此行有 L_i 個字，之後有 L_i 個字。然後會有一個正整數 M 代表有 M 個指令，之後再 M 行為指令的內容。

Output

請輸出 N 行字，每行的字與字之間以空白相隔，每行的結尾沒有空白。

Sample Input 1

```
3
1 apple
2 bird cat
3 dog elephant frog
5
0 0 1 0
1 0 1 1
1 1 2 0
2 0 2 1
2 1 2 2
```

Sample Output 1

```
bird
cat dog
elephant frog apple
```

Hint

```
char *d;
d = new char[W];
```

請用 new 方法建立所有字的一維陣列，否則可能會造成 segmentation fault。

```
std::cin >> &d[p];
```

可將輸入的字存在陣列 d 的第 p 個位置。

注意存的時候不能將 `'\0'` 覆蓋掉，否則將無法得知字的結尾。

Problems

Announcements

Submissions

Rankings

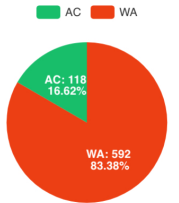
View Contest

Information

ID	3
Time Limit	8000MS
Memory Limit	15MB
Created By	kuruwa
Level	Low
Score	20
Tags	Show

Statistic

Details



Language: C++

Theme: Solarized Light

🔔 Contest has ended

✎ Submit

