

[◀ 戻る](#)[leftrightarrow サンプルコード](#)[leftrightarrow 問題FAQ](#)

「故障パートを探せ」 / 難易度: 3

問題タイプ: コーディング問題 目標タイム: 30分 アルゴリズム / データ処理

問題文

N 個のパートのうち、いくつかが故障しているようです。

過去 M 回の動作結果から、正常に動作しない可能性のあるパート i を全て、半角スペースで区切り昇順で出力してください。

動作結果

M 回の動作結果は以下の形式で与えられます。

```
N M
C_1 S_1
C_2 S_2
.
.
.
C_M S_M
```

- 整数 N はパートの数
- 整数 M は動作結果の回数
- 文字列 C_i は '**o**' または '**x**' のいずれかからなる長さ N の文字列
- S_i は '**T**' または '**F**' のいずれかの文字

C_i の j 番目の文字が '**o**' の時は j 番目のパートを使用したことを意味し、'**x**' の時は使用していないことを意味します。

S_i が '**T**' の時は正常に動作したことを意味し、'**F**' の場合は動作しなかったことを意味します。

正常に動作しない可能性のあるパートとは、そのパートを使用した時に一度も正常に動作しなかったパートのこととします。

また、過去に一度も使用されていないパートも正常に動作しない可能性のあるパートとします。

具体例

例えば、以下のような動作結果が与えられたとします。

```
3 2
xox F
oxx T
```

1回目の動作結果は、2番目のパートを使用し、正常に動作しなかったことを表します。

2回目の動作結果は、1番目のパートを使用し、正常に動作したことを表します。

この場合、正常に動作しない可能性のあるパートは 2, 3 番目のパートです。

入力される値

サンプルケース1

入力値 行数: 4

```
3 2
xox T
xxo F
```

出力値 行数: 2

```
1 3
```

サンプルケース2

入力値 行数: 3

```
7 1
xxxxxxxx T
```

出力値 行数: 2

```
1 2 3 4 5 6 7
```

サンプルケース3

入力値 行数: 9

```
10 7
xxxxxxxxxo F
xoxxxoxox T
ooooooooo F
```

出力値 行数: 2

```
1 3 4 9 10
```

テストする

```
N M
C_1 S_1
C_2 S_2
.
.
.
C_M S_M
```

- 1行目に、パーツの数 N 、動作結果の回数 M が半角スペースで区切られ与えられます。
- 2行目から $M+1$ 行にかけて、 i 番目の動作結果のパーツ構成 C_i 、動作の可否 S が半角スペースで区切られ与えられます。

期待される出力値

過去 M 回の動作結果から、正常に動作しない可能性のあるパーツ i を全て、半角スペースで区切り昇順で出力してください。

制約

- N, M : 整数
- $1 \leq N \leq 10^3$
- $0 \leq M \leq 10^3$
- $|C_i| = N$
- C_i は **o** または **x** から構成される。
- S_i は **T** または **F** からなる。
- 正常に動作しない可能性のあるパーツが 1 つ以上含まれる

サンプルケース1

入力値

```
3 2
xox T
xxo F
```

期待される出力値

```
1 3
```

説明

1番目のパーツは1度も使用されていないため、正常に動作しない可能性があるパーツです。
 2番目のパーツは1回目の動作結果で正常に動作しているため、2番目のパーツは正常です。
 3番目のパーツは2回目の動作結果でのみ使用されていて、この結果では動作しなかったため、正常に動作しない可能性のあるパーツです。
 以上より、正常に動作しない可能性のあるパーツは1,3番目のパーツとなります。

サンプルケース2

入力値

7 1
xxxxxx T

期待される出力値

1 2 3 4 5 6 7

説明

パートを使用した動作結果がないため、全てのパートが正常に動作しない可能性があります。
このようにパートを全く使わないような動作結果も存在します。

サンプルケース3

入力値

10 7
xxxxxxxxo F
xoxxoxxx T
xooooooooo F
xxxxxoxox T
xxoxxxooxo F
xoxxxxoooo T
xxxxxxxxo F

期待される出力値

1 3 4 9 10

配置変更

C++ ▾



```
1  /* CやC++などシェルに実行結果コード返却を明示する言語を利用する場合 基本的に0を返却してください。 */  
2  #include <iostream>  
3  #include <string>  
4  using namespace std;  
5  
6  int main()  
7  {  
8      string s;  
9      cin >> s;  
10     cout << s << endl;  
11     return 0;  
12 }
```

コードを提出する

2017 444 Inc. all rights reserved