

UVA10189

---

# 題目

Have you ever played Minesweeper? It's a cute little game which comes within a certain Operating System which name we can't really remember. Well, the goal of the game is to find where are all the mines within a  $M \times N$  field. To help you, the game shows a number in a square which tells you how many mines there are adjacent to that square. For instance, suppose the following  $4 \times 4$  field with 2 mines (which are represented by an '\*' character):

*	.	.	.
.	.	.	.
.	*	.	.
.	.	.	.

您玩過《踩地雷》嗎？這是一款可愛的小遊戲，遊戲的目標是找到所有 $M \times N$ 地圖內的地雷。為了幫助您，遊戲在一個正方形中顯示一個數字，告訴您該正方形附近有多少個地雷。

例如，假設下面的 $4 \times 4$ 的地圖內帶有2個地雷(以"\*"字元表示)。

# 題目

If we would represent the same field placing the hint numbers described above, we would end up with:

*	1	0	0
2	2	1	0
1	*	1	0
1	1	1	0

As you may have already noticed, each square may have at most 8 adjacent squares.

如果我們根據上述作法，將遊戲提示數字填入，則結果將為：

*	1	0	0
2	2	1	0
1	*	1	0
1	1	1	0

當然，您可能已經注意到，每個正方形內的數字最多為8(因為最多有8個正方形相鄰)。

# 輸入與輸出

Input : The input will consist of an arbitrary number of fields. The first line of each field contains two integers  $n$  and  $m$  ( $0 < n, m \leq 100$ ) which stands for the number of lines and columns of the field respectively.

The next  $n$  lines contains exactly  $m$  characters and represent the field.

Each safe square is represented by an '.' character (without the quotes) and each mine square is represented by an '\*' character (also without the quotes). The first field line where  $n = m = 0$  represents the end of input and should not be processed.

Output : For each field, you must print the following message in a line alone:

Field # $x$ :

Where  $x$  stands for the number of the field (starting from 1). The next  $n$  lines should contain the field with the '.' characters replaced by the number of adjacent mines to that square. There must be an empty line between field outputs.

輸入：輸入將包含多組測資。每組測資第一行包含兩個整數 $n$ 和 $m$  ( $0 < n, m \leq 100$ )，代表地圖大小。如果 $n = m = 0$ 代表輸入結束。接下來的 $n$ 行，每行 $m$ 個字元，代表整張地圖。每個安全方塊用"."字元表示，每個地雷方塊用"\*"字元表示。

輸出：對於每組測資。輸出第一行為"Field # $k$ :"， $k$ 代表測資編號。接下來輸出題示後的遊戲地圖。每筆測資間請用空白行分隔。

# 範例測資

*Input*

4 4

\*...

....

\*..  
.

....

3 5

\*\*..  
.

....

\*..  
.

0 0

*Output*

Field #1:

\*100

2210

1\*10

1110

Field #2:

\*\*100

33200

1\*100

# 程式碼說明

## Step 1：輸入測資

```
6      int n,m,k=0;
7      while (cin>>n>>m&&n!=0&&m!=0) {
8          if (k++!=0) {
9              cout<<endl;
10         }
```

已宣告變數	註解
n、m	地圖大小
k	第幾個測資

# 程式碼說明

## Step 2：輸入每行並將周圍的格子加一

```
12  vector<vector<int>> vec(n, vector<int>(m,0));
13  string s;
14  for(int i=0;i<n;i++){
15      cin>>s;
16      for(int j=0;j<m;j++){
17          if(s[j]=='*'){
18              vec[i][j]=-1;
19              for(int vec_i=max(0,i-1);vec_i<=min(i+1,n-1);vec_i++)
20                  for(int vec_j=max(0,j-1);vec_j<=min(j+1,m-1);vec_j++)
21                      if(vec[vec_i][vec_j]!=-1)
22                          vec[vec_i][vec_j]++;
23          }
24      }
25  }
```

已宣告變數	註解
n、m	地圖大小
k	第幾個測資
s	地圖每行
vec	處理後的地圖

# 程式碼說明

## Step 3：輸出

```
25      cout<<"Field #"<<k<<":\n";
26      for (auto i: vec) {
27          for (auto j: i) {
28              if (j==-1)
29                  cout<<"*";
30              else
31                  cout<<j;
32          }
33      cout<<endl;
```

已宣告變數	註解
n、m	地圖大小
k	第幾個測資
s	地圖每行
vec	處理後的地圖



# 完整程式碼

```
1  #include<iostream>
2  #include<vector>
3  #include<string>
4  using namespace std;
5  int main() {
6      int n,m,k=0;
7      while(cin>>n>>m&& n!=0&&m!=0) {
8          if(k++!=0) {
9              cout<<endl;
10         }
11         vector<vector<int>> vec(n, vector<int>(m,0));
12         string s;
13         for(int i=0;i<n;i++) {
14             cin>>s;
15             for(int j=0;j<m;j++) {
16                 if(s[j]=='*') {
17                     vec[i][j]=-1;
18                     for(int vec_i=max(0,i-1);vec_i<=min(i+1,n-1);vec_i++)
19                         for(int vec_j=max(0,j-1);vec_j<=min(j+1,m-1);vec_j++)
20                             if(vec[vec_i][vec_j]!=-1)
21                                 vec[vec_i][vec_j]++;
22                 }
23             }
24         }
25         cout<<"Field #"<<k<<":\n";
26         for (auto i: vec) {
27             for (auto j: i) {
28                 if(j== -1)
29                     cout<<"*";
30                 else
31                     cout<<j;
32             }
33             cout<<endl;
34         }
35     }
36 }
```



# 資料來源

英文題目：

<https://vjudge.net/problem/UVA-10189>

中文題目：

<https://zerojudge.tw/ShowProblem?problemid=e605>

*Thank You*

---