IOITC 2021

Island Hopping

The nation of Islandia can be represented as an $n \times n$ grid. Let (i, j) denote the square at the i-th row from the top and the j-th column from the left. A cell (i, j) is 1 if it is land and 0 if it is water. Any two cells of the grid which share a side, and are both land are said to belong to the same island.

Islandia holds two contests a year, the first at (x_a, y_a) and the second at (x_b, y_b). Ashish will need to travel from cell (x_a, y_a) to (x_b, y_b). To do so he can perform a sequence of zero or more moves. In each move if he is at (x_i, y_i), he can perform one of two actions:

1. Move from (x_i, y_i) to some land cell (x_j, y_j) that lies on a different island at a cost of $\mid x_i - x_j \mid + \mid y_i - y_j \mid$
2. Move from (x_i, y_i) to some land cell (x_j, y_j) that lies on the same island at a cost of zero.

Let the beauty of a sequence of moves be the number of moves of type 1 while traveling from (x_a, y_a) to (x_b, y_b). We define the simple cost as the minimum cost of a sequence with $beauty \leq 1$. Ashish wants to figure out the maximum beauty possible without exceeding the simple cost.

Ashish knows (x_a, y_a) and wants to figure out the maximum beauty for each possible (x_b, y_b).

Input

- The first line contains T, the number of testcases. Each testcase contains $N + 2$ lines
- The first line of each test case contains N.
- The following N lines of each testcase contains a string $S = s_1 s_2 \ldots s_N$. The string on the i-th line representing the i-th row of the grid $-$ s_j is 1 if the cell (i, j) of the grid is a land cell, and 0 if it is a water cell.
- The last line of each test case containing two integers x_a and y_a $(1 \leq x_a, y_a \leq n)$ where (x_a, y_a) represents the cell where the first contest will be held. It is guaranteed that this cell will be a land cell.

Output

Print N lines, each containing the N space seperated integers. The j-th value of the i-th line represents the maximum beauty possible on a journey from (x_a, y_a) to (i, j) satisfying the mentioned conditions. If (i, j) is a water cell, or belongs to the same island as (x_a, y_a), the value to be printed is 0.

Test Data

In all inputs,

- $1 \leq T \leq 50$
- $1 \leq N \leq 300$
- The sum of N^2 over all cases doesn't exceed 10^5 $(\sum N^2 \leq 10^5)$

Subtask 1 (10 Points): $N \leq 20$, the sum of N^2 over all testcases doesn't exceed 500 and each island consists of exactly 1 land cell.

Subtask 2 (10 Points): $N \leq 20$ and the sum of N^2 over all testcases doesn't exceed 500.

Subtask 3 (10 Points): $N \leq 60$ and the sum of N^2 over all testcases doesn't exceed 4000.

Subtask 4 (30 Points): $N \leq 100$ and the sum of N^2 over all testcases doesn't exceed 20000.

Subtask 5 (20 Points): $N \leq 180$ and the sum of N^2 over all testcases doesn't exceed 36000.

Subtask 6 (20 Points): No additional constraints

Sample Input 1

```
2
2
00
01
2 2
5
00001
10010
00001
00000
01010
2 4
```

Sample Output 1

```
0 0
0 0
0 0 0 0 1
1 0 0 0 0
0 0 0 0 1
0 0 0 0 0
0 2 0 1 0
```

Sample Input 2

```
2
3
011
010
001
2 2
9
110000011
001100001
100100111
011000010
000110011
000010100
000011111
111101001
101110110
3 1
```

Sample Output 2

```
0 0 0
0 0 0
```

```
0 0 1
1 1 0 0 0 0 0 3 3
0 0 1 1 0 0 0 0 3
0 0 0 1 0 0 3 3 3
0 1 1 0 0 0 0 3 0
0 0 0 2 2 0 0 3 3
0 0 0 0 2 0 2 0 0
0 0 0 0 2 2 2 2 2
1 1 1 1 0 2 0 0 2
1 0 1 1 1 0 6 6 0
```

Sample Input 3

```
2
7
1011101
0000000
0000000
0000000
0000000
0000000
0000000
1 1
9
111111101
100000000
100000000
100000000
100000000
100000000
100000000
100000000
111111101
9 9
```

Sample Output 3

```
0 0 1 1 1 0 2
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
0 0 0 0 0 0 0
1 1 1 1 1 1 1 0 4
1 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 0 0
```

Explanation

In the second case of the first sample, the simple cost from $(2, 4)$ to $(5, 2)$ is $\mid 2 - 5 \mid + \mid 4 - 2 \mid = 3 + 2 = 5$. We can achieve a beauty of 2 without exceeding the simple cost in the following manner:

- Perform a move of type 1 from $(2, 4)$ to $(5, 4)$ at a cost of 3.
- Perform a move of type 1 from $(5, 4)$ to $(5, 2)$ at a cost of 2.

In the first case of third sample, the simple cost from $(1, 1)$ to $(1, 7)$ is $\mid 1 - 1 \mid + \mid 1 - 7 \mid = 0 + 6 = 6$. However we can achieve a beauty of 2 without exceeding the simple cost in the following manner:

- Perform a move of type 1 from $(1, 1)$ to $(1, 3)$ at a cost of 2.
- Perform a move of type 2 from $(1, 3)$ to $(1, 5)$ at a cost of 0.
- Perform a move of type 1 from $(1, 5)$ to $(1, 7)$ at a cost of 2.

Note that the cost of some sequence with beauty 1 might be lower than the simple cost.
In the second case of third sample, the simple cost from $(9, 9)$ to $(1, 9)$ is $\mid 9 - 1 \mid + \mid 9 - 9 \mid = 8 + 0 = 8$. However we can achieve a beauty of 4 without exceeding the simple cost in the following manner:

- Perform a move of type 1 from $(9, 9)$ to $(9, 7)$ at a cost of 2.
- Perform a move of type 2 from $(9, 7)$ to $(1, 7)$ at a cost of 0.
- Perform a move of type 1 from $(1, 7)$ to $(1, 9)$ at a cost of 2.
- Perform a move of type 1 from $(1, 9)$ to $(1, 7)$ at a cost of 2.
- Perform a move of type 1 from $(1, 7)$ to $(1, 9)$ at a cost of 2.

Note that we **DO NOT need to minimize the cost of the trip**, only maximize the beauty without exceeding the simple cost.
Also note that in the same sample the answer for $(1, 7)$ will be 1, not 3 as the simple cost from $(9, 9)$ to $(1, 7)$ is 2. This can be achieved by performing a move of type 1 from $(9, 9)$ to $(9, 7)$ at cost of 2 and then a move of type 2 from $(9, 7)$ to $(1, 7)$ at a cost of 0.

Limits

Time: 3 seconds

Memory: 1024 MB