

PROGRAMMING ASSIGNMENT 2

DUE: Tuesday July 12, 11:59 PM. DO NOT COPY. ACKNOWLEDGE YOUR SOURCES.

Please read <http://www.student.cs.uwaterloo.ca/~cs341> for general instructions and policies.

Popped Corn

Problem Statement

Farmer has fields that are modeled as squares on $n \times n$ two-dimensional grid. The grid of cells has n rows going from north to south and numbered from 1 to n , and n columns going from west to east numbered from 1 to n . So coordinates of north-west corner are $(1, 1)$, and coordinates of south-east corner are (n, n) . Each grid cell either contains corn or water. This data is represented as array F of characters, with F_{ij} equal to "c" or "w" (representing "corn" or "water" respectively). A fire starts in a corn cell (x, y) , $1 \leq x, y \leq n$ (x represents the row and y represents the column) at time 0, and simultaneously spreads one cell in each direction (North, South, East, West) in each unit of time (unless the cell was already burned or contains water). It also takes one unit of time for the cell of corn to be completely burned. The fire cannot spread to water (and thus can be blocked by water). Use graph representation and algorithms you have learnt in the class to answer the following 3 questions:

1. How long will it take for the fire to be over if it starts at corn cell (x, y) ?
2. If not all corn will burn, how many cells with corn will remain (considering that fire starts at cell (x, y) ?)
3. What is the minimum number of fires needed to burn all the corns? In other words, if cells with corn can be chosen to start fires and we choose those cells optimally in order to burn all the corns with minimum number of chosen cells, what would be the number of cells we have chosen for starting fires?

Input Format

In the first line of the input, you are given the integer n which is the size of the grid.

Next n lines, represents the $n \times n$ grid and what each cell contains. So, each line shows one row of the grid with n characters equal to "c" or "w" separated by one [SPACE]. Lines show the rows in order; so, first line represents the first row, second line represents the second row, and so on. Also, each line represent the type of the grid ("corn" or "water") of columns in order. So, the first character in a line represents the grid cell for the first column and the last character in the line shows the last or n -th column in that row.

Finally, in the last line of the input, x and y are given in one line separated by one space.

Output Format

The output should contain three lines. Each line is the answer to one of the three questions asked in the problem description.

Specifically, in the first line you should output the time when fire is over assuming that fire starts at grid (x, y) at time 0 (and will spread to applicable neighbouring cells at time 1). In the second line, your output should be the number of remaining corn cells if the fire starts at cell (x, y) . Finally, in the third line, you should output the minimum number of fires needed for burning all the corn cells. Thus, your output should consist of three lines, each of which contains an integer.

Constraints

For all test cases, $1 \leq n \leq 1000$, $F_{ij} = \text{"w"}$ or "c" for $1 \leq i, j \leq n$, and $1 \leq x, y \leq n$, $F_{xy} = \text{"c"}$. 30 points (out of 100) are allotted for secret test cases with large n values. Your solution should be as efficient as possible to solve these cases.

Sample Input 1

```
3
c c c
c c c
c c c
1 1
```

Sample Output 1

```
5
0
1
```

Sample Input 2

```
3
c w c
w c c
c c c
2 2
```

Sample Output 2

```
3
1
2
```

Sample Input 3

```
3
c w c
w c c
c c c
1 1
```

Sample Output 3

```
1
6
2
```

Submission Instructions

- Submit your solution on Marmoset.
- You can choose to code in either C++ or Python.
- Name your program `prog2.cpp/prog2.py`
- **Time limit:** 2 seconds (C++) / 4 seconds (Python) for each test case.
- Compilation command for C++: `g++ -std=c++14 prog2.cpp -O3 -o prog2`
- Execution command for Python: `python3 prog2.py`
- Read from standard input and write to standard output.
- There will be several test cases, worth a total of 100 points. The public tests are worth 25 points and the secret tests are worth 75 points. The public tests (input only, not the answer) will be made available under a separate file.
- We will take the submission with the highest score. Please, however, refrain from excessive submissions.
- General collaboration policy applies. Please acknowledge your collaborator(s) by adding a comment in the beginning of your code.
- FAQ and updates will be posted on Piazza when necessary