**IOI Team Training 2013 – Online Test 1, 6–7 June, 2013**

## Snowman

Your school is organizing a computer-based art contest in which you need to draw something and display it using your own computer. The theme for the contest is *Winter is coming*, but you, being unaware of the Game of Thrones reference, decide that you would like to draw a Snowman.

The snowman you draw will be of a very simple shape. It will consist of two rectangles, with one representing the head sitting on top of the one representing the body. The rectangles should be such that the lower edge of the top rectangle should be on the same horizontal line as the upper edge of the bottom rectangle, and should be completely contained in it (not necessarily properly contained). You decide that the aesthetic beauty of your snowman is equal to the product of the two rectangles' interior area.

Unfortunately, your computer screen has some bad pixels. If your snowman's drawing passes through such bad pixels, it will not look like a snowman any more. You thus wish to maximize the beauty of a snowman that can be drawn using only good pixels. For this problem, consider your computer screen to be a square $N \times N$ grid of pixels, where the value $N$ will also be given to you.

For example, if your screen is shaped as shown on the left, with bad pixels marked by `*`, the optimally drawn snowman is the one shown on the right whose outline is described by `x`.

```
..............        ..XXXXXXXXXX..
..............        ..X........X..
...*******.....        ..X*******..X..
.*....*.......*        .*X...*.....X.*
.*......*....*.        .*X.....*...X*.
....*.........        ..X.*.......X..
...*...****....        ..X*...****.X..
..............        .XXXXXXXXXXXXXX
..**.*..*..*...        .X**.*..*..*..X
...*...**.*....        .X.*...**.*...X
*..*...*.......        *X.*...*.....X
..............        .X...........X
.....*..*......        .X...*..*....X
.........*.....        .X.......*....X
..............        .XXXXXXXXXXXXX
```

In the example above, the rectangles' interior areas are $6 \times 9$ and $12 \times 6$, giving a total beauty value of 3888.

## Program and Grader

Implement one function in the file `snowman.cpp`.

```
int beauty(int N, char** screen);
```

that, given the value $N$ and the 2d character array describing the screen (`'*'` is a bad pixel, `'.'` is a good one), returns the maximum value of a snowman's beauty.

### Input and Output

The program `grader_snowman.cpp` takes input in the following format:

- The first line has a single integer, $N$.
- This is followed by $N$ lines having $N$ characters each, describing the screen.

`grader_snowman.cpp` then calls your function `beauty()` and outputs the value that your function returns.

### Test data

- Subtask 1 (30 marks) : $N \leq 80$
- Subtask 2 (70 marks) : $N \leq 300$

### Sample input

```
...............
...............
...*******.....
.*....*.......*
.*......*....*.
....*..........
...*...****....
...............
..**.*..*..*...
...*...**.*....
*..*...*.......
...............
.....*..*......
.........*.....
...............
```

### Sample output

```
3888
```

### Limits

- *Memory limit* : 128 MB
- *Time limit* : 5s