

# I. Loader

time limit per test: 4 seconds  
memory limit per test: 256 megabytes  
input: standard input  
output: standard output

A loader works in a warehouse, which is a rectangular field with size  $n \times m$ . Some cells of this field are free, others are occupied by pillars on which the roof of the warehouse rests.

There is a load in one of the free cells, and the loader in another. At any moment, the loader and the load can not be in the cells with columns, outside the warehouse or in the same cell.

The loader can move to the adjacent cell (two cells are considered adjacent if they have a common side), or move the load. To move the load, the loader should reach the cell adjacent to the load and push the load. In this case the load advances to the next cell in the direction in which the loader pushes it and the loader ends up in the cell in which the load was.

Your task is to determine a sequence of pushes and loader's movements after which the load will reach the given cell (it is guaranteed that this cell is free). The load is rather heavy, so you need to minimize first the number of pushes and second the number of loader's movements.

## Input

The first line contains two positive integers  $n$  and  $m$  ( $1 \leq n, m \leq 40, n \cdot m \geq 3$ ) — the number of rows and columns in the rectangular field.

Each of the next  $n$  lines contains  $m$  characters — the description of the warehouse. If there is a character in the next cell of the warehouse:

- "X", it means, that the current cell contains the column;
- ".", it means, that the current cell is free;
- "Y", it means, that the loader is in the current cell;
- "B", it means, that the load is in the current cell;
- "T", it means, that the load should be moved to this cell.

It is guaranteed that there is exactly one load, one loader and one cell to which the load should be moved.

## Output

If the loader is not able to move the load to the given cell, print "NO" (without the quotes) in the first line of the output.

Otherwise, print "YES" (without the quotes) in the first line of the output, and in the second line — the sequence of characters that determines movements and pushes of the loader. Characters w, e, n, s shall denote loader's moves to the west, east, north and south, respectively. Characters W, E, N, S must denote loader's pushes in the corresponding directions. First of all you need to minimize the number of pushes of the load and second, the number of movements of the loader. If there are several answers, you are allowed to print any of them.

## Examples

input
3 3 ..Y .BX ..T
output
YES wSwSE

input

3 3  
.BY  
...  
TXX

output

NO