**Grid and phrase**

Max. Marks: 100

You are given an**n\*m**grid which contains lower case English letters. How many times does the phrase "saba" appear horizontally, vertically, and diagonally in the grid?

**Input format**

* First line: Two integer n and m, where n denotes (1 <= n,m <= 100) the number of rows and m denotes the number of columns in the grid
* Next n lines: Each line must contain a string of length m which contains lower-case English letters only

**Output format**

Print the number of times the word “saba” appears in the grid.

**SAMPLE INPUT**

5 5

safer

amjad

babol

aaron

songs

**SAMPLE OUTPUT**

2

**Explanation**

The phrase "saba" must look one of thease shapes :

|  |  |  |  |
| --- | --- | --- | --- |
| s |  |  |  |
|  | a |  |  |
|  |  | b |  |
|  |  |  | a |

|  |  |  |  |
| --- | --- | --- | --- |
| s | a | b | a |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| s |  |  |  |
| a |  |  |  |
| b |  |  |  |
| a |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  | a |
|  |  | b |  |
|  | a |  |  |
| s |  |  |  |

**Time Limit:**1.0 sec(s) for each input file.

**Memory Limit:**256 MB

**Source Limit:**1024 KB

**Marking Scheme:**Marks are awarded if any testcase passes.

**Allowed Languages:**Bash, C, C++, C++14, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, Racket, Ruby, Rust, Scala, Swift, Swift-4.1, TypeScript, Visual Basic

<https://www.hackerearth.com/challenges/competitive/april-circuits-19/algorithm/jadvaliioo-62280ff6/>

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp2

{

class Program

{

static int ContarSABA(string[] matriz)

{

int ans = 0;

for(int i =0; i<matriz.Length; i++)

{

for(int j =0; j<matriz[i].Length; j++)

{

if (matriz[i][j] == 's')

{

// if (i - 3 >= 0)

// {

// if (matriz[i - 1][j] == 'a' &&

// matriz[i - 2][j] == 'b' &&

// matriz[i - 3][j] == 'a')

// {

// ans++;

// }

// }

if (i + 3 < matriz.Length)

{

if (matriz[i + 1][j] == 'a' &&

matriz[i + 2][j] == 'b' &&

matriz[i + 3][j] == 'a')

{

ans++;

}

}

//else if (j - 3 >= 0)

// {

// if (matriz[i][j-1] == 'a' &&

// matriz[i][j-2] == 'b' &&

// matriz[i][j-3] == 'a')

// {

// ans++;

// }

// }

if (j + 3 < matriz[i].Length)

{

if (matriz[i][j + 1] == 'a' &&

matriz[i][j + 2] == 'b' &&

matriz[i][j + 3] == 'a')

{

ans++;

}

}

// //---------------

//else if (i - 3 >= 0 && j -3 >=0)

// {

// if (matriz[i - 1][j - 1] == 'a' &&

// matriz[i - 2][j-2] == 'b' &&

// matriz[i - 3][j-3] == 'a')

// {

// ans++;

// }

// }

if (i - 3 >= 0 && j + 3 < matriz[i].Length)

{

if (matriz[i - 1][j + 1] == 'a' &&

matriz[i - 2][j + 2] == 'b' &&

matriz[i - 3][j + 3] == 'a')

{

ans++;

}

}

//else if (i + 3 < matriz.Length && j - 3 >= 0)

// {

// if (matriz[i + 1][j - 1] == 'a' &&

// matriz[i + 2][j - 2] == 'b' &&

// matriz[i + 3][j - 3] == 'a')

// {

// ans++;

// }

// }

if (i + 3 < matriz.Length && j + 3 < matriz[i].Length)

{

if (matriz[i + 1][j + 1] == 'a' &&

matriz[i + 2][j + 2] == 'b' &&

matriz[i + 3][j + 3] == 'a')

{

ans++;

}

}

}

}

}

return ans;

}

static void Main(string[] args)

{

//string[] matriz = {

// "safer",

// "amjad",

// "babol",

// "aaron",

// "songs",

//};

string[] nm = Console.ReadLine().Trim().Split(' ');

int n = int.Parse(nm[0]);

int m = int.Parse(nm[1]);

string[] matriz = new string[n];

for(int i =0; i<n; i++)

{

matriz[i] = Console.ReadLine().Trim();

}

Console.WriteLine(ContarSABA(matriz));

//Console.WriteLine(ContarSABA(matriz));

Console.ReadLine();

}

}

}