Mr. Jack very recently started gardening. He has a very specific way of planting new flowers. He denotes the position of his flowers with a single number. The number tells him the number of the flower.

Given a series of positions, he is able to draw the map of the entire garden.

For example, the positions 1 3 4 0 2 means his garden is like this:

* * * # *

* * * *

* * * * #

* # * * *

* * # * *

Where the * represents grass and # is a flower.

Write a program takes as input an integer array. The integer array has the positions and you have to print the garden. The number of rows in the garden is the same as the length of the input array.

Input Format

The first line contains n, the number of test cases.

Each test case has m, the size of the array, followed by m integers - the elements of the arrays.

Use the System.out.println() statement to print.

Example Input:

1

5

1 3 4 0 2

Example Output:

* * * # *

* * * *

* * * * #

* # * * *

* * # * *

Example Input:

1

3

0 0 0

Example Output

# # #

* * *

* * *

For example:

Input	Result
1 5 1 3 4 0 2	* * * # * # * * * * * * * * # * # * * * * * # * *
1 3 0 0 0	# # # * * * * * *

Answer: (penalty regime: 0 %)

```java
import java.util.*;
public class Flowers
{
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        int t=sc.nextInt();
        for(int m=0;m<t;m++)
        {
            int n=sc.nextInt();
            char a[][]=new char[n][n];
            for(int i=0;i<n;i++)
            {
                for(int j=0;j<n;j++)
                {
                    a[i][j]='*';
                }
            }

            for(int col=0;col<n;col++)
            {
                int val=sc.nextInt();
                a[val][col]='#';
            }


            for(int i=0;i<n;i++)
            {
                for(int j=0;j<n;j++)
                {
                    System.out.print(a[i][j]+" ");
                }
                System.out.println();
            }
        }
    }
}
```

	Input	Expected	Got	
✔	1 5 1 3 4 0 2	* * * # * # * * * * * * * * # * # * * * * * # * *	* * * # * # * * * * * * * * # * # * * * * * # * *	✔
✔	1 3 0 0 0	# # # * * * * * *	# # # * * * * * *	✔

Passed all tests! ✔

Mr. Jack has a new passion: farming. Has a simple request from you: given a farm map, find the longest sequence of consecutive vegetables in his farm, either in the same row, or vertically or diagonally.

Write a program that takes as input a 2D character array, which stores the map. You have to print out the value of the longest sequence.

Note: * is grass and V is vegetable

Example Input:

3 4

* * * *

* V V *

* * * *

Example Output:

2

Example Input:

3 4

* V V V

* V V *

* * * *

Example Output:

3

For example:

Input	Result
3 4 * * * * * V V * * * * *	2
3 4 * V V V * V V * * * * *	3

Answer: (penalty regime: 0 %)

```
 1   import java.util.*;
 2   public class Farming
 3   {
 4       public static void main(String[] args)
 5       {
 6           Scanner sc = new Scanner(System.in);
 7           int r = sc.nextInt();
 8           int c = sc.nextInt();
 9           int max=0,count=0;
10           char a[][] = new char[r][c];
11           for(int i=0;i<r;i++)
12           {
13               for(int j=0;j<c;j++)
14               {
15                   a[i][j] = sc.next().charAt(0);
16               }
17           }
18           for(int i=0;i<r;i++)
19           {
20               count=0;
21               for(int j=0;j<c;j++)
22               {
23                   if(a[i][j]=='V')
24                   {
25                       count++;
```

```
26                }
27            }
28            if(count>max)
29            {
30                max=count;
31                System.out.print(max);
32            }
33        }
34        for(int i=0;i<c;i++)
35        {
36            count=0;
37            for(int j=0;j<r;j++)
38            {
39                if(a[j][i]=='V')
40                {
41                    count++;
42                }
43            }
44            if(count>max)
45            {
46                max=count;
47                System.out.print(max);
48            }
49        }
50    }
51 }
```

	Input	Expected	Got	
✔	3 4 * * * * * V V * * * * *	2	2	✔
✔	3 4 * V V V * V V * * * * *	3	3	✔

Your code failed one or more hidden tests.
Your code must pass all tests to earn any marks. Try again.

Dr. CooCoo is going trekking. She wants to climb all the peaks possible in a given map. You have to tell her the number of peaks she has to climb.

A map contains the heights of the different points on the ground. A point is a peak if the immediate points to it's north, south, east and west all have a lesser height than it.

Write a program that takes as input a 2D array, containing the map, and prints the number of peaks in that map.

Example Input:

3 3

1 0 2

3 1 0

0 0 0

Example Output:

2

Example Input:

4 4

1 0 0 1

0 0 0 0

0 0 0 0

1 0 1 0

Example Output:

4

For example:

Input	Result
3 3 1 0 2 3 1 0 0 0 0	2
4 4 1 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0	4

Answer: (penalty regime: 0 %)

```java
import java.util.*;
public class Peaks
{
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        int r = sc.nextInt();
        int c = sc.nextInt();
        int sum=0;

        int a[][] = new int[r][c];

        for(int i=0;i<r;i++)
        {
            for(int j=0;j<c;j++)
            {
                a[i][j] = sc.nextInt();
            }
        }
        for(int i=0;i<r;i++)
        {
```

```
22              int flag=0;
23              for(int j=0;j<c;j++)
24              {
25                  if(j-1>=0 && j-1<=c-1 && a[i][j-1] >= a[i][j])
26                  {
27                      flag=1;
28                      break;
29                  }
30                  if(j+1>=0 && j+1<=c-1 && a[i][j+1] >= a[i][j])
31                  {
32                      flag=1;
33                      break;
34                  }
35                  if(i-1>=0 && i-1<=r-1 && a[i-1][j] >= a[i][j])
36                  {
37                      flag=1;
38                      break;
39                  }
40                  if(i+1>=0 && i+1<=r-1 && a[i+1][j] >= a[i][j])
41                  {
42                      flag=1;
43                      break;
44                  }
45                  if(flag==0)
46                  {
47                      sum++;
48                  }
49              }
50          }
51      System.out.println(sum);
52  }
```

	Input	Expected	Got	
✗	3 3 1 0 2 3 1 0 0 0 0	2	1	✗
✗	4 4 1 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0	4	2	✗

Some hidden test cases failed, too.
Your code must pass all tests to earn any marks. Try again.

Show differences

Mr. Jack has a problem in his garden. If any two flowers in his garden are adjacent either in row, column or diagonal, then they will attack each other.

Given a garden map, you have to tell Mr. Jack if the plants will attack or not.

Write a program that takes input a 2D character array. Print Attack if the map is meets the above conditions otherwise print Safe

Note: the * represents grass and # is a flower.

Example Input:
3

# # #

* * *

* * *

Example Output:

Attack

Example Input:

4

* * * #

* * # *

* * *

* # * *

Example Output:

Attack

Example Input:
4

* * * #

* # * *

* * * *

* # * *

Example Output:

Safe

For example:

Input	Result
3 3 # # # * * * * * *	Attack
4 4 * * * # * * # * # * * * * # * *	Attack
4 4 * * * # * # * * * * * * * # * *	Safe

Answer: (penalty regime: 0 %)

```
1  import java.util.*;
2  public class Garden
```

```
3 ▼ {
4       public static void main(String[] args)
5 ▼      {
6           Scanner sc = new Scanner(System.in);
7           int r = sc.nextInt();
8           int c = sc.nextInt();
9           int flag=0;
10
11          char a[][] = new char[r][c];
12
13          for(int i=0;i<r;i++)
14 ▼        {
15              for(int j=0;j<c;j++)
16 ▼            {
17                  a[i][j] = sc.next().charAt(0);
18              }
19          }
20          for(int i=0;i<r;i++)
21 ▼        {
22              for(int j=0;j<c;j++)
23 ▼            {
24
25                  if(a[i][j]=='#')
26 ▼                {
27                      if(j-1>=0 && a[i][j-1]=='#')
28 ▼                    {
29                          flag=1;
30                          break;
31                      }
32                      if(j+1>=0 && j+1<=c-1 && a[i][j+1]=='#')
33 ▼                    {
34                          flag=1;
35                          break;
36                      }
37                      if(i-1>=0 && a[i-1][j]=='#')
38 ▼                    {
39                          flag=1;
40                          break;
41                      }
42                      if(i+1>=0 && i+1<=r-1 && a[i+1][j]=='#')
43 ▼                    {
44                          flag=1;
45                          break;
46                      }
47                      if(i-1>=0 && j-1>=0 && a[i-1][j-1]=='#')
48 ▼                    {
49                          flag=1;
50                          break;
51                      }
52                      if(i-1>=0 && j+1>=0 && j+1<=c-1 && a[i-1][j+1]=='#')
```

	Input	Expected	Got	
✔	3 3 # # # * * * * * *	Attack	Attack	✔
✔	4 4 * * * # * * # * # * * * * # * *	Attack	Attack	✔
✔	4 4 * * * # * # * * * * * * * # * *	Safe	Safe	✔

Passed all tests! ✔