Codeforces Round #817 (Div. 4)

A. Spell Check

time limit per test
1 second
memory limit per test
256 megabytes
input
standard input
output
standard output

Timur likes his name. As a spelling of his name, he allows any permutation of the letters of the name. For example, the following strings are valid spellings of his name: Timur, miurT, Trumi, mriTu. Note that the correct spelling must have uppercased T and lowercased other letters.

Today he wrote string ss of length nn consisting only of uppercase or lowercase Latin letters. He asks you to check if ss is the correct spelling of his name.

Input

The first line of the input contains an integer tt ($1 \le t \le 10^3$1≤t≤103) — the number of test cases.

The first line of each test case contains an integer nn ($1 \le n \le 10$)(1≤n≤10) — the length of string ss.

The second line of each test case contains a string ss consisting of only uppercase or lowercase Latin characters.

Output

For each test case, output "YES" (without quotes) if ss satisfies the condition, and "NO" (without quotes) otherwise.

You can output the answer in any case (for example, the strings "yEs", "yes", "Yes" and "YES" will be recognized as a positive answer).

Example

input

Copy

```
10
5
Timur
5
miurT
5
Trumi
5
mriTu
5
timur
4
Timr
6
Timuur
10
codeforces
10
TimurTimur
5
TIMUR
```

output

Copy

```
YES
YES
YES
YES
NO
NO
NO
NO
NO
NO
```

B. Colourblindness

time limit per test
1 second
memory limit per test
256 megabytes
input
standard input
output
standard output

Vasya has a grid with 2 rows and n columns. He colours each cell red, green, or blue.

Vasya is colourblind and can't distinguish green from blue. Determine if Vasya will consider the two rows of the grid to be coloured the same.

Input

The input consists of multiple test cases. The first line contains an integer t ($1 \le t \le 100$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains an integer n ($1 \le n \le 100$) — the number of columns of the grid.

The following two lines each contain a string consisting of n characters, each of which is either R, G, or B, representing a red, green, or blue cell, respectively — the description of the grid.

Output

For each test case, output "YES" if Vasya considers the grid's two rows to be identical, and "NO" otherwise.

You can output the answer in any case (for example, the strings "yEs", "yes", "Yes" and "YES" will be recognized as a positive answer).

Example

input

Copy

```
6
2
RG
RB
4
GRBG
GBGB
5
GGGGG
BBBBB
7
BBBBBBB
RRRRRRR
8
RGBRRGBR
RGGRRBGR
1
G
G
```

output

Copy

```
YES
NO
YES
NO
YES
YES
```

Note

In the first test case, Vasya sees the second cell of each row as the same because the second cell of the first row is green and the second cell of the second row is blue, so he can't distinguish these two cells. The rest of the rows are equal in colour. Therefore, Vasya will say that the two rows are coloured the same, even though they aren't.

In the second test case, Vasya can see that the two rows are different.

In the third test case, every cell is green or blue, so Vasya will think they are the same.

C. Word Game

time limit per test
1 second
memory limit per test
256 megabytes
input
standard input
output
standard output

Three guys play a game: first, each person writes down n distinct words of length 3. Then, they total up the number of points as follows:

- if a word was written by one person — that person gets 3 points,
- if a word was written by two people — each of the two gets 1 point,
- if a word was written by all — nobody gets any points.

In the end, how many points does each player have?

Input

The input consists of multiple test cases. The first line contains an integer t ($1 \le t \le 100$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains an integer n ($1 \le n \le 1000$) — the number of words written by each person.

The following three lines each contain n **distinct** strings — the words written by each person. Each string consists of 3 lowercase English characters.

Output

For each test case, output three space-separated integers — the number of points each of the three guys earned. You should output the answers in the same order as the input; the i-th integer should be the number of points earned by the i-th guy.

Example

input

Copy

```
3
1
abc
def
abc
3
orz for qaq
qaq orz for
cod for ces
5
iat roc hem ica lly
bac ter iol ogi sts
bac roc lly iol iat
```

output

Copy

```
1 3 1
2 2 6
9 11 5
```

Note

In the first test case:

- The word abcabc was written by the first and third guys — they each get $1$1 point.
- The word defdef was written by the second guy only — he gets $3$3 points.

D. Line

time limit per test
2 seconds
memory limit per test
256 megabytes

input
standard input
output
standard output

There are n people in a horizontal line, each looking either to the left or the right. Each person counts the number of people in the direction they are looking. The **value** of the line is the sum of each person's count.

For example, in the arrangement LRRLL, where L stands for a person looking left and R stands for a person looking right, the counts for each person are $[0,3,2,3,4]$, and the value is $0+3+2+3+4=12$.

You are given the initial arrangement of people in the line. For each k from 1 to n, determine the maximum value of the line if you can change the direction of **at most** k people.

Input

The input consists of multiple test cases. The first line contains an integer t ($1 \le t \le 100$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains an integer n ($1 \le n \le 2 \cdot 10^5$) — the length of the line.

The following line contains a string consisting of n characters, each of which is either L or R, representing a person facing left or right, respectively — the description of the line.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Please note that the answer for some test cases won't fit into 32-bit integer type, so you should use at least 64-bit integer type in your programming language (like long long for C++).

Output

For each test case, output n space-separated non-negative integers — the maximum value of the line if you can change the direction of **at most** k people for each k from 1 to n, inclusive.

Example

input

Copy

```
6
3
LLR
5
LRRLL
1
L
12
LRRRLLLRLLRL
10
LLLLLRRRRR
9
LRLRLRLRL
```

output

Copy

```
3 5 5
16 16 16 16 16
0
86 95 98 101 102 102 102 102 102 102 102 102
29 38 45 52 57 62 65 68 69 70
44 50 54 56 56 56 56 56 56
```

Note

In the first test case:

- $k=1$: change the direction of 1 person to make the line RLR. The total value is $2+1+0=3$.
- $k=2$: change the direction of 2 people to make the line RLL. The total value is $2+1+2=5$.
- $k=3$: change the direction of 2 people to make the line RLL. The total value is $2+1+2=5$. Note that you have to change the direction of **at most** k people.

In the second test case, it is optimal to only change the direction of the first person for all k from 1 to 5 (that is, make the line RRRLL).

E. Counting Rectangles

time limit per test

6 seconds
memory limit per test
256 megabytes
input
standard input
output
standard output

You have n rectangles, the i-th rectangle has height h_i and width w_i.
You are asked q queries of the form h_s w_s h_b w_b.
For each query output, the total area of rectangles you own that **can fit** a rectangle of height h_s and width w_s while also **fitting in** a rectangle of height h_b and width w_b. In other words, print $\sum h_i \cdot w_i$ for i such that $h_s < h_i < h_b$ and $w_s < w_i < w_b$.

Please note, that if two rectangles have the same height or the same width, then they *cannot* fit inside each other. Also note that you **cannot** rotate rectangles.

Please note that the answer for some test cases won't fit into 32-bit integer type, so you should use at least 64-bit integer type in your programming language (like long long for C++).

Input

The first line of the input contains an integer t ($1 \le t \le 100$) — the number of test cases.
The first line of each test case two integers n, q ($1 \le n \le 10^5$; $1 \le q \le 10^5$) — the number of rectangles you own and the number of queries.
Then n lines follow, each containing two integers h_i, w_i ($1 \le h_i, w_i \le 1000$) — the height and width of the i-th rectangle.
Then q lines follow, each containing four integers h_s, w_s, h_b, w_b ($1 \le h_s < h_b$, $w_s < w_b \le 1000$) — the description of each query.
The sum of q over all test cases does not exceed 10^5, and the sum of n over all test cases does not exceed 10^5.

Output

For each test case, output q lines, the i-th line containing the answer to the i-th query.

Example

input

Copy

```
3
2 1
2 3
3 2
1 1 3 4
5 5
1 1
2 2
3 3
4 4
5 5
3 3 6 6
2 1 4 5
1 1 2 10
1 1 100 100
1 1 3 3
3 1
999 999
999 999
999 998
1 1 1000 1000
```

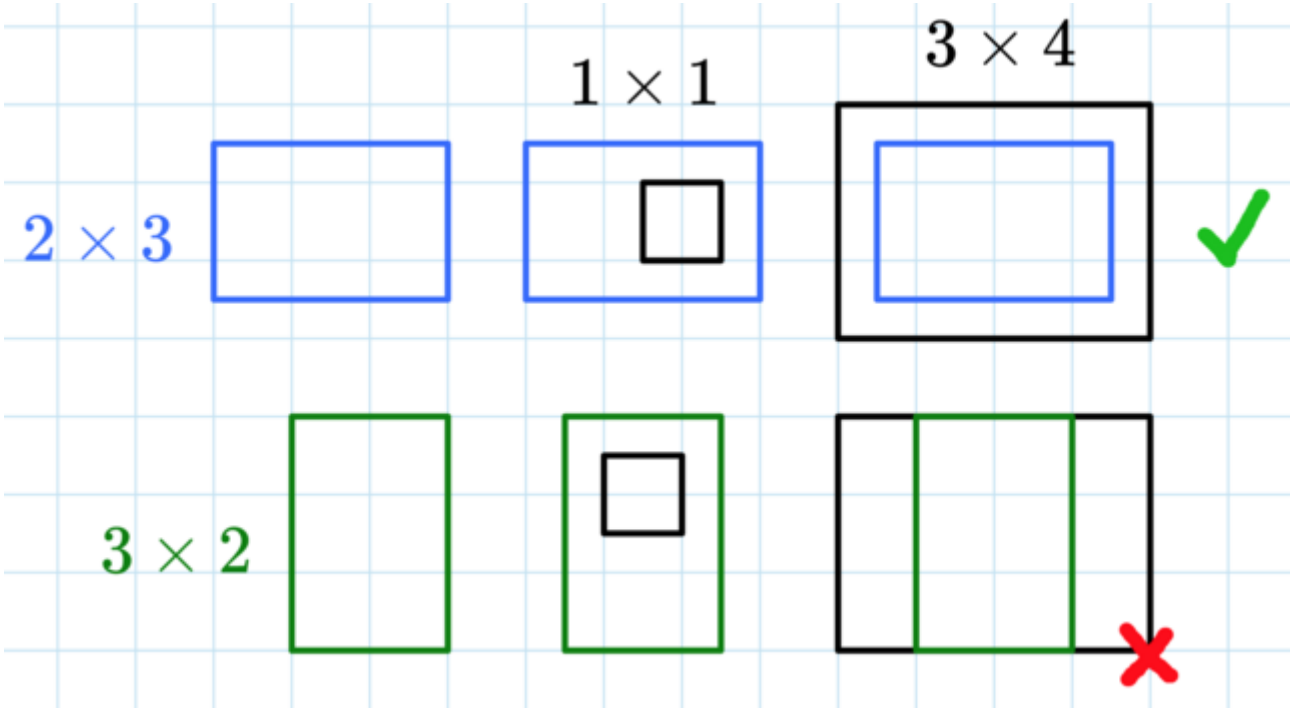
output

Copy

```
6
41
9
0
54
```

```
4
2993004
```

Note



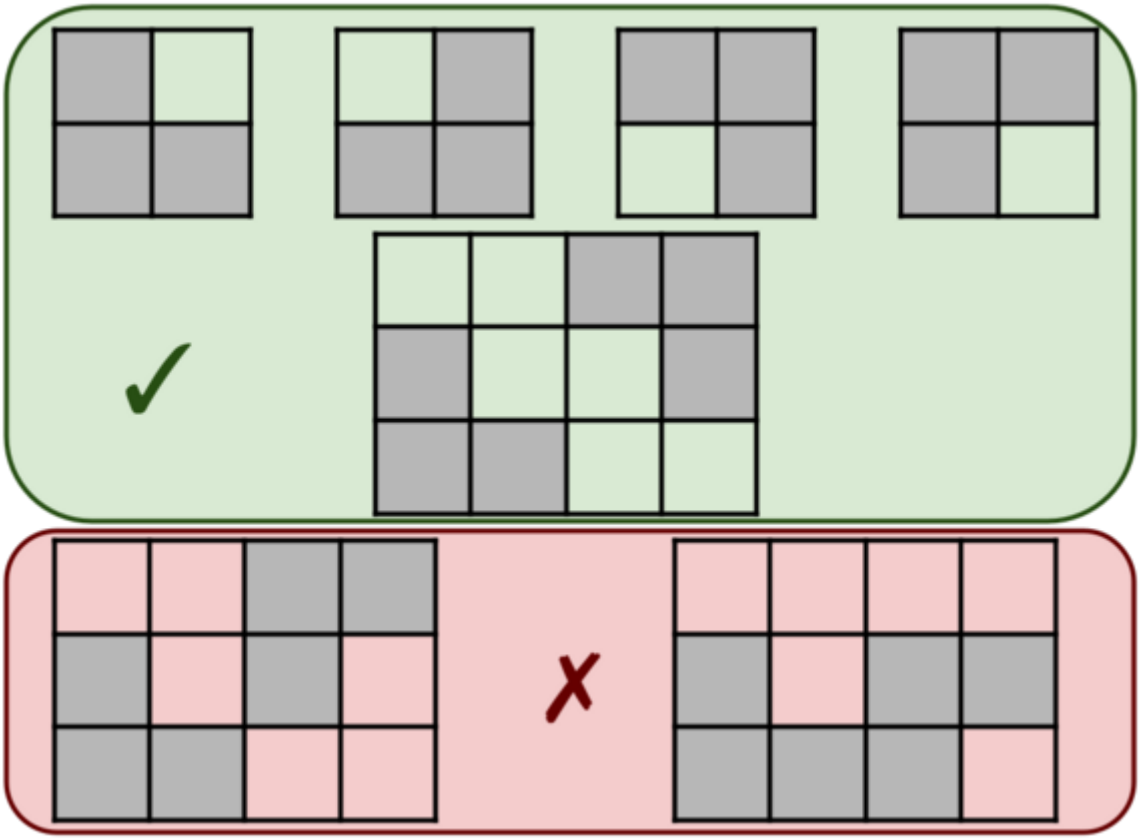
In the first test case, there is only one query. We need to find the sum of areas of all rectangles that can fit a 1×1 rectangle inside of it and fit into a 3×4 rectangle.

Only the 2×3 rectangle works, because $1<2$ (comparing heights) and $1<3$ (comparing widths), so the 1×1 rectangle fits inside, and $2<3$ (comparing heights) and $3<4$ (comparing widths), so it fits inside the 3×4 rectangle. The 3×2 rectangle is too tall to fit in a 3×4 rectangle. The total area is $2\cdot3=6$.

F. L-shapes

time limit per test
1 second
memory limit per test
256 megabytes
input
standard input
output
standard output

An L-shape is a figure on gridded paper that looks like the first four pictures below. An L-shape contains exactly three shaded cells (denoted by *), which can be rotated in any way.



You are given a rectangular grid. Determine if it contains L-shapes only, where L-shapes can't touch an edge or corner. More formally:

- Each shaded cell in the grid is part of exactly one L-shape, and
- no two L-shapes are adjacent by edge or corner.

For example, the last two grids in the picture above **do not** satisfy the condition because the two L-shapes touch by corner and edge, respectively.

Input

The input consists of multiple test cases. The first line contains an integer t ($1 \le t \le 100$) — the number of test cases. The description of the test cases follows.

The first line of each test case contains two integers n and m ($1 \le n, m \le 50$) — the number of rows and columns in the grid, respectively.

Then n lines follow, each containing m characters. Each of these characters is either '.' or '*' — an empty cell or a shaded cell, respectively.

Output

For each test case, output "YES" if the grid is made up of L-shape that don't share edges or corners, and "NO" otherwise.

You can output the answer in any case (for example, the strings "yEs", "yes", "Yes" and "YES" will be recognized as a positive answer).

Example

input

Copy

```
10
6 10
.......**
.**......*
..*..*....
.....**...
...*.....*
..**....**
6 10
....*...**
```

```
.**.....*
..*..*....
.....**...
...*.....*
..**....**
3 3
...
***
...
4 4
.*..
**..
..**
..*.
5 4
.*..
**..
....
..**
..*.
3 2
.*
**
*.
2 3
*..
.**
3 2
..
**
*.
3 3
.**
*.*
**.
3 3
..*
.**
..*
```

output

Copy

```
YES
NO
NO
NO
YES
NO
NO
YES
NO
NO
```

G. Even-Odd XOR

time limit per test
1 second
memory limit per test
256 megabytes
input
standard input
output
standard output

Given an integer n, find any array a of n **distinct** nonnegative integers less than 2^{31} such that the bitwise XOR of the elements on odd indices equals the bitwise XOR of the elements on even indices.

Input

The first line of the input contains an integer t ($1 \le t \le 629$) — the number of test cases.

Then t lines follow, each containing a single integer n ($3 \le n \le 2 \cdot 10^5$) — the length of the array.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output one line containing n distinct integers that satisfy the conditions.

If there are multiple answers, you can output any of them.

Example

input

Copy

```
7
8
3
4
5
6
7
9
```

output

Copy

```
4 2 1 5 0 6 7 3
2 1 3
2 1 3 0
2 0 4 5 3
4 1 2 12 3 8
1 2 3 4 5 6 7
8 2 3 7 4 0 5 6 9
```

Note

In the first test case the XOR on odd indices is $4 \oplus 1 \oplus 0 \oplus 7 = 2$ $4 \oplus 1 \oplus 0 \oplus 7 = 2$ and the XOR on even indices is $2 \oplus 5 \oplus 6 \oplus 3 = 2$ $2 \oplus 5 \oplus 6 \oplus 3 = 2$.