长安大学第二届ACM-ICPC 程序设计竞赛校赛暨省赛选拔
正式赛试题

ACM association of CHD

April 23, 2017

Contents

A Count Circles

Description

Stupid Aguin feels confused while reading. The book shows following equations:

$$6 = 9\ ,\ 8 = 1010\ ,\ 144 = 75\ ,\ 690 = 801$$

Stupid Aguin doesn't know why and he asks RoyYuan for help. RoyYuan tells Aguin that he only needs to count circles in each number. Notice that 0 , 6 and 9 have one circle, and 8 has two circles. For example, both 690 and 801 have 3 circles, so 690 = 801.

However, Aguin is too stupid to count circles in each number, please help him.

Input

The first line contains an integer number T, the number of test cases.

i-th of each next T lines contains an integer number $x (0 \le x \le 10^9)$.

Output

For each test case print a number, the number of circles in x.

Sample Input

```
8
6
9
8
1010
144
75
690
801
```

Sample Output

```
1
1
2
2
0
0
3
3
```

4 5 6 7
1 3 5 7
3 2 1

B Boys and Girls

Description

AA is a kindergarten teacher, there are n boys and n girls in her class. Chairs in the classroom are put in a row, children select their seats according to their own preferences. However, AA wants to make boys and girls sit alternately. That is to say, if $n = 2$, 'B' said a boy and 'G' said a girl, she wants her children sit like "BGBG" or "GBGB". So she decides to make some changes, each time she chooses two adjacent children and swap their seats. Now she wants to know how many times she needs to swap at least.

BBGGGGBB
01234567

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains an integer $n(1 \le n \le 10^5)$, the number of boys and girls.

The next line contains a string s consists of 'B' and 'G', representing the initial seats of children.

It's guaranteed that $|s| = 2n$ and both 'B' and 'G' appear exactly n times in the string s.

Output

For each test case print a number, the minimum number of times required.

Sample Input

2
1
GB
2
BBGG

BBBGGG

2 BGBBGG

BBBGGG 3
BBGBGG 3
BGBBGG 3
BGBGBG

BBBGGG 4

BBGBGG

BBGG
0123

Sample Output

0
1

012345
BBBGGG
BGGBBG
012345

BGBBGG 2

BBBGGG

GBBGGB

BBGGGGBB
BGBGGGBB

BBBGGG
012345

gg bb

C Roses

Description

The Little Prince has n roses in his garden, which can be described as a 2-dimensional plane. The i^{th} of the roses has the coordinate (x_i, y_i).It's raining recently, to protect the roses, the Little Prince plans to arrange two umbrellas(overlapping is allowed) to cover all the roses. The area covered by one umbrella is a circle, the Little Prince can determine the sizes of two umbrellas(not necessarily the same) and he can also choose any positions to place these two umbrellas. Meanwhile, in order not to block too much sunshine after the rain, he wants to minimize the covered areas.

Help the Little Prince to calculate the minimized covered areas that satisfy the condition.

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains an integer $n(1 \le n \le 50)$, the number of roses.

The following n lines, each contains two integers x_i, $y_i(|x_i|, |y_i| \le 10^4)$, the coordinate of the i^{th} rose.

It's guaranteed that no coordinates coincide.

Output

For each test case print the minimum covered areas, round to two decimal places.

Sample Input

```
2
3
0 0
1 1
2 2
8
-3 -4
-4 -3
-3 4
4 -3
1 8
8 1
8 9
9 8
```

Sample Output

```
1.57
142.81
```

D Arrays

Description

A[n] B[n]

Given two arrays A and B, both of them have n elements. Then there are m queries, each of which contains four integers: L, R, x, y. For each query, you are required to count different indexes $k \in [L, R]$, that satisfy $A[k] \geq x$ and $B[k] \geq y$.

A[k]≥x B[k]≥y

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains two integers n, $m (1 \leq n, m \leq 10^5)$.

The second line contains n integers $A[i] (1 \leq A[i] \leq 10^5)$.

The third line contains n integers $B[i] (1 \leq B[i] \leq 10^5)$.

i-th of each next m lines contains four integers L, R, x, $y (1 \leq L \leq R \leq n, 1 \leq x, y \leq 10^5)$.

Output

For each query print the answer.

Sample Input

```
1
5 5
1 2 3 4 5
7 6 5 4 3
1 2 1 1
1 5 1 1
2 4 3 4
2 2 2 6
3 3 3 3
```

length.
A
B

[1,2] x=1 y=1 { [1,2] [7,6] k=2 k=3

[1,5] x=1 y=1 { [1,5] [7,3] k=1 k=2 k=3, 4, 5

Sample Output

```
2
5
2
1
1
```

[1,3,4] 3
[6,5,4] 4 >1

for (k=L; k<=R. k++)
if (A[k-1]>=x&&B[k-1]>=y) ans++

E Colorful Ribbon

Description

Lizishu has a colorful ribbon, which can be expressed as a string consists of only lowercase letters, and each letter represents a color. Now she wants to divide the ribbon into several parts so that no color appears more than one time in each part.

Tell her how many different ways can she divide the ribbon, output the answer modulo $10^9 + 7$.

Input

The first line contains an integer number T, the number of test cases.

i-th of each next T lines contains a string s consists of lowercase letters ($1 \le |s| \le 10^5$).

Output

For each test case print the answer modulo $10^9 + 7$.

Sample Input

```
2
a
abcba
```

$4\times3\times2\times1$ $3\times2=6$ $48 - 6 = 42$

Sample Output

```
1
12
```

$A_5^5 = 5! = 120$

~~5! = ...3~~ $3\times$

[aa][bb] $\cancel{2A_4^4 - A_3^3 = 2\times24 - 6} = 42$

$2A_4^4A_2^2 - A_3^3A_2^2A_2^2$

$= 2\times24\times2 - 6\times2\times2$

$= 96 - 24 = 72$

$C_2^2C_2^2$

a/bc/ba

a/b/cba

~~a/bc~~

ab/c/ba

ab cb/a

abc/b/a

ab/cba 2

abcba

2×a 2×b 1×c

$\cancel{C_2^1C_2^1}$

$\frac{C_2^1C_2^1}{A_2^2}\cdot C_2^1$

F XOR-mean

Description

As we all know, the mean of two integers A and B is $\frac{A+B}{2}$.Similarly, let's define XOR-mean of two non-negative integers A and B as $\left\lfloor \frac{A \oplus B}{2} \right\rfloor$($\oplus$ is the XOR operation, $\lfloor x \rfloor$ is the integer part of x).Given an array C of length n, count different triples (i, j, k) that satisfy $1 \le i < j < k \le n$ and $C[j] = \left\lfloor \frac{C[i] \oplus C[k]}{2} \right\rfloor$.

$C(n)$ 小于大多.

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains an integer $n (1 \le n \le 50000)$.

The second line contains n integers $C[i] (0 \le C[i] \le 50000)$.

Output

For each query print the answer.

Sample Input

```
1
10
1 2 3 4 5 6 7 8 9 10
```

Sample Output

```
23
```

G K-partite Graph

Description

We are all familiar with bipartite graph, actually it can be extended to multipartite graph.

If vertices of an undirected graph G can be divided into exactly $k(k \geq 2)$ non-empty sets, and for each pair of vertices u and v, there exists an edge (u, v) if and only if they are from different sets, then G is defined as a complete k-partite graph.

Given an undirected graph G with n vertices and m edges, judge whether it is a complete k-partite graph.

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains two integers n and $m(1 \leq n \leq 1000, 0 \leq m \leq \frac{n \times (n-1)}{2})$, the number of vertices and edges.

i-th of each next m lines contains two integers u_i ans v_i, which means there exists an undirected edge between u_i and v_i $(1 \leq u_i, v_i \leq n, u_i \neq v_i)$.

It's also guaranteed that no duplicated edges in the input.

Output

For each test case :

print a number k if G is a complete k-partite graph$(k \geq 2)$, print "0"(without quotation) otherwise.

Sample Input

```
3
1 0
3 3
1 2
2 3
3 1
6 6
1 5
1 6
2 4
2 6
3 4
3 5
```

Sample Output

```
0
3
0
```

H Boxes

Description

OldWC is a warehouseman, the warehouse can be described as a $n \times m$ grid, there are several(maybe none) boxes on each grid.In order to prevent dumping, on each grid, boxes can not be stacked for more than h layers.OldWC needs to move some boxes to meet the requirement, every time, he can move a box from one grid to an adjacent grid,two grids are adjacent if and only if they share an edge.

Calculate how many times he needs to move at least to make each pile of boxes within h layers.

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains three integers n, m, $h(1 \le n, m, h \le 50)$, the size of the warehouse and the limited number of layers.

The following n lines, each contains m integers $a_{ij}(0 \le a_{ij} \le 50)$, the initial number of boxes on the i^{th} row j^{th} column.

It's guaranteed that $\sum a_{ij} \le n \times m \times h$.

Output

For each test case print the answer.

Sample Input

```
1
2 3 5
1 2 9
6 5 4
```

Sample Output

```
5
```

I Annual Party

Description

There are n cities in the Rainbow Island, connected by $n-1$ bidirectional roads. In the i^{th} of each next m years, k_i of the cities would like to hold a party and they would choose one from these k_i cities as the place for the party. For convenience, every year, the sum of distance from the chosen city to all of these k_i cities should be minimized.

Now you are given the map of the Rainbow Island and the k_i cities which would hold the party, calculate the minimized sum of distance from the chosen city to all of the k_i cities every year.

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains two integers n, $m(1 \leq n, m \leq 100000)$, number of cities and years.

The following $n-1$ lines, each contains three integers $u, v, w(1 \leq u, v \leq n,\ u \neq v,\ 1 \leq w \leq 10^9)$, which means there is a bidirectional road between u and v, the length of road is w.

It's guaranteed that the graph is connected.

i-th of each next m lines contains k_i+1 integers, first is the number of cities for the party $k_i(1 \leq k_i \leq n)$, then come k_i different integers a_1, a_2, ... , $a_{k_i}(1 \leq a_j \leq n,\ 1 \leq j \leq k_i)$, the indexes of these k_i cities.

It's also guaranteed that for each test case $\sum k_i \leq 100000$.

Output

For every year print the answer.

Sample Input

```
1
7 2
4 2 6
5 6 1
2 5 3
1 2 5
3 7 4
1 3 2
1 3
4 7 6 2 3
```

Sample Output

```
0
22
```

Hint

```
int main()
{
    // avoid stack overflow
    ///////////////////////////////////////
    int size = 256 << 20; // 256MB
    char *pp = (char*)malloc(size) + size;
    __asm__("movl %0, %%esp\n" :: "r"(pp));
    ///////////////////////////////////////
    /*

        your code here

    */
    return 0;
}
```

J Repeated String

Description

qwb has talent for solving string problems.He obtains a lowercase string A of length $n \times m$, and he wants to construct another lowercase string B of the same length to make the distance from A to B as small as possible.Because of his laziness, he decides to construct B in an easy way. Firstly, he constructs a string of length n, then he repeats it for m times.

For two strings A and B with the same length L, their distance is defined as the following:

$$dist(A, B) = \sum_{i=1}^{L} |A[i] - B[i]|$$

Tell qwb what's the minimum distance he can get.

Input

The first line contains an integer number T, the number of test cases.

For each test case :

The first line contains two integers n, $m(1 \leq n \times m \leq 10^5)$.

The next line contains a lowercase string A of length $n \times m$.

Output

For each test case print the minimum distance.

Sample Input

2
1 6
abcdef
2 3
aabbcc

Sample Output

9
4

K Brackets

Description

Let us define a regular brackets sequence in the following way:

1. Empty sequence is a regular sequence.
2. If "S" is a regular sequence, then "(S)" is a regular sequence.
3. If "A" and "B" are regular sequences, then "AB" is a regular sequence.

For example, all of the following sequences are regular brackets sequences: "()", "(())", "()()", "()(())". And all of the following sequences are not: "(", ")", ")(", "(((()".

Doc has n pairs of brackets and he wants to know how many regular brackets sequences of length $2n$ can he construct that satisfy the x-th left bracket must not be matched before the y-th left bracket is matched($x < y$), tell him the answer modulo $10^9 + 7$.

Input

The first line contains an integer number T($T \leq 200$), the number of test cases.

i-th of each next T lines contains three numbers n, x, y($2 \leq n \leq 500$, $1 \leq x < y \leq n$).

Output

For each test case print the answer modulo $10^9 + 7$.

Sample Input

```
3
2 1 2
6 2 4
15 9 10
```

Sample Output

```
1
57
4566642
```

L QAQ Number

Description

QAQ, the greatest mathematician of the 21st century, found a new number called QAQ Number. The QAQ Numbers are positive integers without leading zeros which satisfy the following conditions:

1. It has exactly $3k$ digits(k is a positive integer), and can be divided into three sections $a_1 \ldots a_k$, $a_{k+1} \ldots a_{2k}$ and $a_{2k+1} \ldots a_{3k}$.
2. Digits of the same section are the same. Explicitly, $a_1 = a_2 = \ldots = a_{k-1} = a_k$, $a_{k+1} = a_{k+2} = \ldots = a_{2k-1} = a_{2k}$, $a_{2k+1} = a_{2k+2} = \ldots = a_{3k-1} = a_{3k}$.
3. The first and third sections are the same, which means $a_1 = a_{2k+1}$, $a_2 = a_{2k+2}$, … , $a_k = a_{3k}$.

For instance, 111222111, 919 and 666666 are QAQ Numbers, but 1111, 010 , 444455554443 are not.

Now QAQ wants to know how many QAQ Numbers are there in range [L, R](inclusive).

Input

The first line contains an integer number T, the number of test cases.

i-th of each next T lines contains two numbers L and R ($1 \le L \le R \le 10^{18}$).

Output

For each test case print one number.

Sample Input

3

1 100

101 999

1000 1000000000

3 6 9

- - - - - -

Sample Output

$C_9^1 \times C_{10}^1$

0

90

180

111 222 111

$(C_9^1 \times C_{10}^1) \times (9/3 - 3/3)$.

$90\left[(R位数-1)/3 - (L位数-1)/3\right]$