2023“钉耙编程”中国大学生算法设计超级联赛（2）

2023 年 7 月 20 日 12:00 ∼ 17:00

目录

1 Alice Game

时间限制：1000ms 空间限制：64MB

1.1 题目描述

Alice and Bob are playing a game.

There are n monsters in the game, and they stand in a line. Alice and Bob take turns. Each turn, the player can choose one of two actions:

1. Destroy a consecutive monster sequence of size less than or equal to K.

2. Select K consecutive monsters to destroy, and after destroying these K monsters, the sequential monster sequence in which they were originally located must be divided into two non-empty sequences. The two remaining sequences will not be considered continuous.

Here is an example of operation 2, if $K = 2$ and there are four monsters $ABCD$ in the field. Now we can destroy monsters BC because they are continuous, and after destroying them we can be left with monsters $AeeD$ (e means the area is empty). But we cannot destroy the monster AB or CD, because the remaining two sequences must be non-empty (in fact, if we do this, only one continuous sequence remains). Similarly, we can't destroy monsters AC or BD because monsters A and C are not continuous.

When a player cannot operate, he loses. Now, Alice will play the game first. She wants to know, can she win at this game?

1.2 输入格式

An integer T indicates that there are T groups of data.

Next T rows. Enter two integers K and n on each line.

Guarantee $1 \leq T \leq 10000, 2 \leq K \leq 10^7, 0 \leq n \leq 10^9$.

1.3 输出格式

Output total T lines.

If Alice can win, output "Alice", otherwise output "Bob ".

1.4 输入输出样例

输入样例：

2

2 2

2 3

输出样例:

Alice

Bob

2 Binary Number

时间限制：1000ms 空间限制：64MB

2.1 题目描述

Markyyz is learning binary numbers. There is an easy problem in his homework.

You are given a binary number $s_{1\sim n}$ (s_1 is the highest bit. s_n is the lowest bit.). You need to do an operation **exactly** k times: select an interval $[l, r]$ ($1 \le l \le r \le n$) arbitrarily and flip $s_l, s_{l+1}, ..., s_r$, in other word, for all $i \in [l, r]$, s_i becomes 1 if s_i is 0, s_i becomes 0 if s_i is 1. What is the biggest result binary number after the k operations.

Markyyz found useless algorithms useless on the problem, so he asked SPY to help. SPY looked down on the problem but finally got WA (wrong answer). Can you help them to find the correct solution?

2.2 输入格式

The first line of the input contains a single integer T ($1 \le T \le 6 \times 10^4$), indicating the number of test cases.

In each test case:

The first line contains two integers n, k. ($1 \le n \le 10^5, 0 \le k \le 10^{18}$)

The second line contains a binary number $s_{1\sim n}$. ($s_1 = 1$, $\forall i \in [2, n] : s_i \in \{0, 1\}$)

It's guarenteed that in all test cases, $\sum n \le 2.5 \times 10^6$

2.3 输出格式

You need to print a string of length n in one line, representing the biggest binary number after the k operations.

2.4 输入输出样例

输入样例：

```
2
8 2
10100101
5 233333333333333333
11101
```

输出样例：

```
11111101
11111
```

3 Conuter Strike

时间限制：5000ms 空间限制：256MB

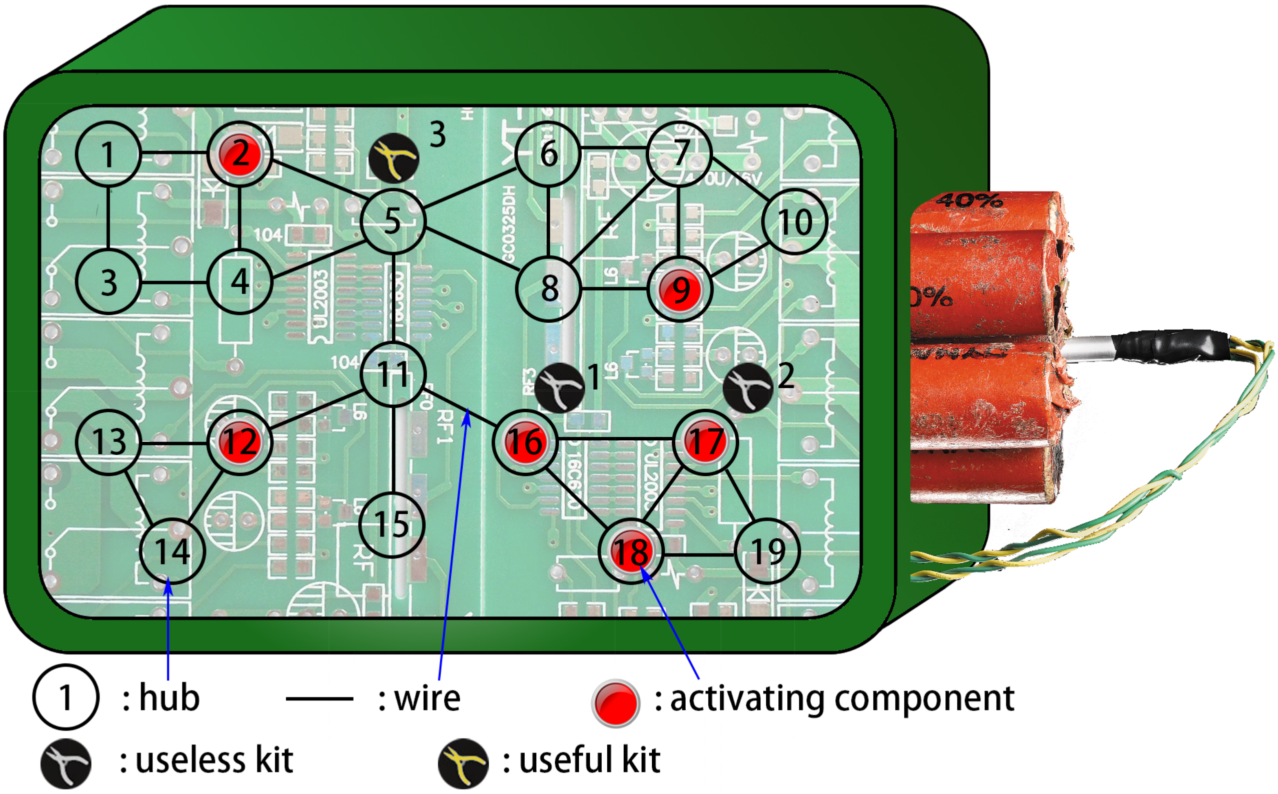
3.1 题目描述

SPY likes useful algorithms, while Markyyz always learns useless algorithms. Although they learn different type of algorithms, they both like a computer game called Counter Strike (”CS” for short, different from the game with the same name in reality). It is a two-player competition. One player acts as a terrorist (”T” for short), responsible for planting bombs. The other one acts as a counter terrorist (”CT” for short), responsible for defusing bombs.

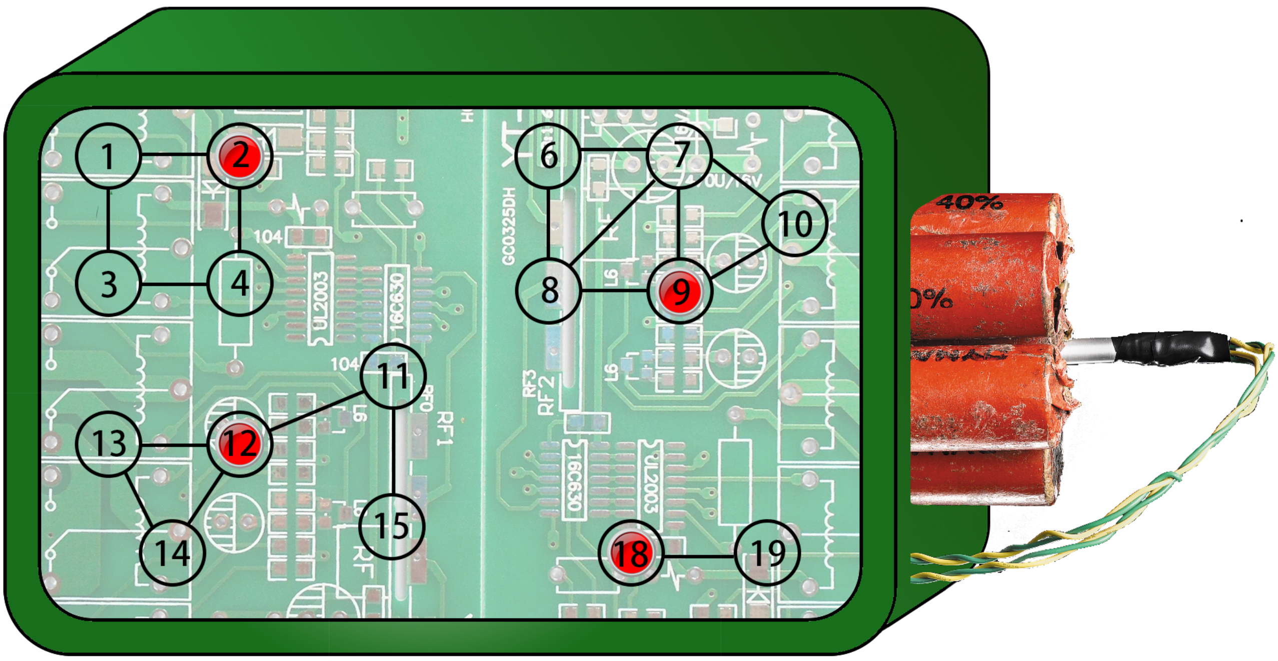
At the start of each game, the system generates a bomb with a circuit in it. The circuit consists of n hubs (numbered from 1 to n) and m wires. Each wire connects two different hubs, any two hubs could be directly or indirectly connected by wires, and no two hubs are directly connected by more than one wire. In other word, the circuit can be regarded as an undirected graph with n vertices and m edges, **without multiple-edges or self-loops**.

The game is divided into three stages:

1. Bomb planting stage: T will receive k **activating components** numbered from 1 to k. T needs to select k different hubs $h_1, h_2, ..., h_k$, and set the i-th **activating component** on the h_i-th hub. 2. Bomb defusing stage: CT will receive a useful kit for free, and buy q useless kits numbered from 1 to q. A useful kit can remove an arbitrary hub, while a useless kit can only remove a hub with an **activating component** in order, which means the i-th useless kit can only remove the h_i-th hub. Once a hub is removed, all wires directly connected to it will be cut off. 3. Bomb activating stage: If any two **activating components** are connected directly or indirectly by wires, the bomb will be activated and T will win. Otherwise, the bomb will be defused and CT will win.



For example, the picture above shows a bomb with a circuit consisting of 19 hubs and 27 wires. T obtained 6 **activating components** and set them on hub $16, 17, 2, 18, 12, 9$. CT can buy 2 useless kits to remove the 16-th and the 17-th hub, then remove the 5-th hub with a useful kit.



Now Markyyz acts as T, and SPY acts as CT. Markyyz has placed k **activating components**. To save money for the future games, SPY wants to buy the minimum number of useless kits to defuse the bomb. Although SPY makes himself master of useful algorithms, he can't find the answer within the time complexity of $O((n+m)\log n)$. Can you help him to calculate the answer (in other word, the minimum value of q for CT to win the game) ?

3.2 输入格式

The first line of the input contains an integer t ($1 \le t \le 5000$), indicating the number of test cases.

In each test case:

The first line contains three integers n, m, k ($2 \le n, m \le 2 \times 10^5, 2 \le k \le n$), indicating the number of hubs, the number of wires, and the number of **activating components**.

The next m lines, each line contains two integers u, v ($1 \le u, v \le n, u \ne v$), indicating a wire connected to the u-th hub and the v-th hub.

The next line contains k different integers $h_1, h_2, ..., h_k$ ($1 \le h_i \le n$). The i-th **activating component** is set in the h_i-th hub.

It is guaranteed that in all test cases $\sum n, \sum m \le 10^6$.

3.3 输出格式

You need to output a single integer in one line, indicating the minimum value of q for CT to win the game.

3.4 输入输出样例

输入样例：

1
19 27 6
1 2
1 3

2 4
3 4
2 5
4 5
5 6
5 8
6 7
6 8
8 7
8 9
7 9
7 10
9 10
5 11
11 12
12 13
13 14
14 12
11 15
11 16
16 17
16 18
17 18
17 19
18 19
16 17 2 18 12 9
输出样例：
2

4 Card Game

时间限制：1000ms 空间限制：64MB

4.1 题目描述

Recently, playing card games has become popular. SPY and Markyyz are also playing a game. In this game, the cards must be placed in piles. Any number of cards can be stacked in the same pile. Any card can be placed at the bottom of an empty pile. The stacked cards go from the bottom to the top, with decreasing and consecutive values. For example, piles $(5, 4, 3, 2, 1)$, $(8, 7, 6, 5, 4, 3)$, (9), and $()$ (an empty pile) are all valid, while piles $(4, 2, 1)$ (not consecutive), $(1, 2, 3)$ (not decreasing), and $(9, 8, 7, 5, 6, 4, 3, 2)$ (neither consecutive nor decreasing) are not valid. (The description of the piles mentioned above is in the order from bottom to top).

In one move, a player can choose a card from the top of a non-empty pile and move it to the top of another pile. Throughout the player's moves, the stacking rules of the cards must be followed, otherwise it is considered a foul.

SPY is now playing the card game on a table with n piles, where one pile contains k cards $(k, k-1, k-2, ..., 2, 1)$, called pile 1, and the rest of the piles are empty piles. All the free slots are empty. SPY wants to move all the cards from pile 1 to another pile (pile 2). At this point, clever Markyyz comes up with a question:

Given the number of piles n, under the condition of not fouling, what is the maximum value of k that allows the movement of k cards as described above?

Since the answer could be large, take the modulus of 998244353.

4.2 输入格式

There are multiple test cases in this problem.

The first line of input contains a positive integer t $(1 \leq t \leq 10^5)$, indicating the number of test cases.

Afterwards, there are t test cases. Each test case consists of a single line containing an integer n $(2 \leq n \leq 10^9)$, representing the number of piles.

4.3 输出格式

For each test case, output a single line containing an integer, representing the maximum value of k for the number of cards. Take the modulus of 998244353.

4.4 输入输出样例

输入样例：

```
3
2
3
114514
```

输出样例：

```
1
3
766171354
```

5 Or

时间限制：6000ms 空间限制：512MB

5.1 题目描述

DDOSvoid is learning about bitwise operations and has come across an interesting problem.

You are given two sequences, a_i and b_i, both of length n. Additionally, there are m queries. In each query, you are given an interval $[l, r]$. Your task is to calculate the bitwise OR operation on the following integers:$a_l, a_l + b_{l+1}, a_l + b_{l+1} + b_{l+2}, \cdots, a_{l+1} + b_{l+2}, a_{l+1} + b_{l+2} + b_{l+3}, \cdots, a_r$. In other words, you need to evaluate $\bigoplus_{i=l}^{r} \bigoplus_{j=i}^{r} (a_i + \sum_{k=i+1}^{j} b_k)$. The symbol $\oplus$ represents the bitwise OR operation.

5.2 输入格式

The first line of the input contains a single integer T, indicating the number of test cases.

In each test case:

The first line contains to integers $n, m (1 \le n \le 10^5, 1 \le m \le 10^6)$.

The second line contains n integers $a_i (0 \le a_i \le 5 \times 10^8)$.

The third line contains n integers $b_i (0 \le b_i \le 5000)$.

The next m lines, each line contains two integers $l, r (1 \le l \le r \le n)$.

It is guaranteed that in all test cases, $\sum n \le 10^5, \sum m \le 10^6$.

5.3 输出格式

To simply the output, we use ans_i to represent the answer to the i-th query and $base = 233, P = 998244353$.

In each test case you just need to output an integer $(\sum_{i=1}^{m} ans_i \times base^i) \bmod P$.

5.4 输入输出样例

输入样例：

1

5 1

1 2 3 4 5

1 1 1 1 1

2 4

输出样例：

1631

Hint：

For query $[2, 4]$, you need to calculate the bitwise OR operation on the following integers, $a_2, a_2 + b_3, a_2 + b_3 + b_4, a_3, a_3 + b_4, a_4$

6 Fencing the cows

时间限制：10000ms 空间限制：256MB

6.1 题目描述

Little ColdHand wants to build a fence to enclose his cows' grazing area. However, in order for the fence to be effective, it must include all m grass locations. Otherwise, the cows might rebel against him.

To address this issue, Little ColdHand sought assistance from the Interstellar Cow Company. However, the company provided him with only n fence points, and he can only build the fence from a point to another point. The final cost will be **the number of points** used.

Little ColdHand is aware that the most cost-effective fence would be a convex hull, but he doesn't know the exact number of points required for it. Therefore, he has approached you to help solve this problem:

Determine the **minimum** number of points needed to construct a fence that completely encloses all m grass-eating locations.

P.S. If the fence intersects any of the grass locations, we do not consider those locations as fully enclosed.

6.2 输入格式

The first line of input contains the integer T $(1 \le T \le 10)$, the number of test cases. The description of test cases follows.

The first line of each test case contains two integers, n and m $(1 \le n \le 500, 1 \le m \le 500)$ —the number of fence points and the number of grass locations.

Each of the next n lines contains the description of fence points. Each line contains two integers x_i and y_i $(-10^9 \le x_i, y_i \le 10^9)$, describes the fence point a_i at (x_i, y_i).

Each of the next m lines contains the description of grass location. Each line contains two integers x_i and y_i $(-10^9 \le x_i, y_i \le 10^9)$, describes the grass location b_i at (x_i, y_i).

it is guaranteed that the sum of n and m over all test cases both do not exceed 4000.

6.3 输出格式

For each test case, if any solution exists, output an integer in a line, indicating the **minimum** cost of fence. otherwise, output -1

6.4 输入输出样例

输入样例：

```
2
4 1
1 1
1 -1
-1 1
-1 -1
0 0
4 1
1 1
1 -1
```

-1 1
-1 -1
1 0
输出样例：
4
-1

7 foreverlasting and fried-chicken

时间限制：1000ms 空间限制：256MB

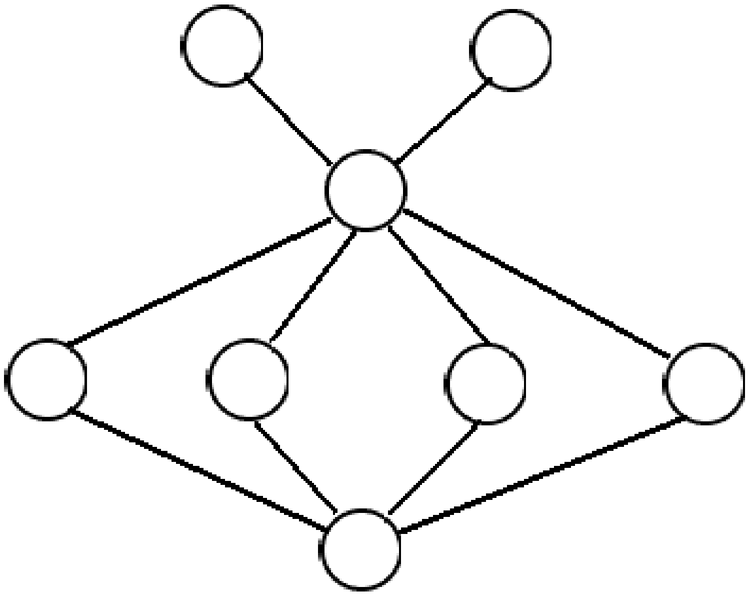
7.1 题目描述

As we all know, There are two ACM heros known as foreverlasting and fried-chicken in BIT. They are immersed in perfect love respectively. The following link tells the love story of fried-chicken.

https://www.zhihu.com/question/62332494/answer/3076483871

Pedestrian1 likes graph and mathematics. He needs your help to solve an easy problem. You are given a simple undirected graph named "fried-chicken" with n nodes and m edges. Please note that the graph is not necessarily connected. The nodes are labeled from 1 to n.

Pedestrian1 wants to know how many "foreverlasting" graphs in the "fried-chicken" graph.



The above image defines a "foreverlasting" graph.

Please note that two "foreverlasting" graphs are considered different when there is at least one different edge between the two edge sets that make up the two "foreverlasting" graphs.

In other word, the given graph is $G(V, E)$. You need to calculate the number of subgraphs $G'(V', E')(V' \subseteq V, E' \subseteq E)$ which satisfy $V' = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$,
$E' = \{(v_1, v_3), (v_2, v_3), (v_3, v_4), (v_3, v_5), (v_3, v_6), (v_3, v_7), (v_4, v_8), (v_5, v_8), (v_6, v_8), (v_7, v_8)\}$

Since the answer may be very large, Pedestrian1 wants to know the answer modulo 1000000007.

7.2 输入格式

The first line of input contains the integer T $(1 \leq T \leq 10)$, the number of test cases. The description of test cases follows.

The first line of each test case contains two integers, n and m $(1 \leq n \leq 1000, m \leq \frac{n(n-1)}{2})$ —the number of nodes and the number of edges, respectively.

Each of the next m lines contains the description of an edge. Each line contains two integers u_i and v_i $(\leq\ u_i, v_i \leq n, u_i \neq v_i)$ —an edge connects node u_i to node v_i.

It is guaranteed that no two edges connect the same unordered pair of nodes.

Furthermore, it is guaranteed that the sum of n over all test cases both do not exceed 3000.

7.3 输出格式

For each testcase, output an integer representing the answer modulo 1000000007.

7.4 输入输出样例

输入样例：

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

输出样例：

1

8 Hello World 3 Pro Max

时间限制：3000ms 空间限制：256MB

8.1 题目描述

Once upon a time, Markyyz invented a problem named ”Hello World”.

Later, Markyyz invented a problem named ”Hello World 2”, which is a harder version of ”Hello World”.

Two thousand years later, SPY invented a problem named ”Hello World 3”, which is an even harder version of ”Hello World”.

Now, SPY is inventing a problem named ”Hello World 3 Pro Max”, which is ...

SPY has a string S of length n consisting of lowercase letters: h, e, l, o, w, r, d. The string is generated randomly in the following way: for each character in S, it is **independently** generated from the set $\{h, e, l, o, w, r, d\}$ with possibilities $p_1, p_2, ..., p_7$. In other words, there is a probability of p_1 for the letter h, p_2 for the letter e, and so on. It is guaranteed that sum of p_i's is equal to 1.

Initially, each character of string S is unknown. Then, SPY will perform q operations of two types:

• Type 1: 1 x c, which means SPY determines that the character S_x is c. In this problem, the characters in string S are indexed starting from 1, so S can be expressed as $S_1S_2S_3...S_n$. It is guaranteed that no two operations will conflict with each other.

• Type 2: 2 l r, which means SPY wants to know the expected number of **subsequences** equals to $helloworld$ in the substring $S(l, r)$, modulo $10^9 + 7$. Here, $S(l, r)$ means the substring of S starting at index l and ending at index r (formally $S_lS_{l+1}...S_r$).

After each operation of Type 2, you should answer the query by outputting the expected number on a separate line, modulo $10^9 + 7$.

8.2 输入格式

There are multiple tests.

The first line of input consists a single integer $t(1 \le t \le 10)$, representing the number of test cases.

In each test case, the following lines provide the details:

The first line consists a single integer $n(1 \le n \le 5 \times 10^4)$, representing the length of string S.

The second line contains 7 integers $P_1, P_2, ..., P_7(1 \le P_i \le 10^8)$. Let $P_t = P_1 + P_2 + ... + P_7$ be the sum of these values. The possibilities of the letters are defined as $p_i = \frac{P_i}{P_t}$.

The third line contains a single integer $q(1 \le q \le 5 \times 10^4)$, representing the number of operations.

The next q lines describe the operations, each line specifying the type and parameters of the operation.

It is guaranteed that sum of n in all test cases will not exceed 5×10^4, sum of q in all test cases will not exceed 5×10^4.

8.3 输出格式

After every operation of Type 2, output the expected number on a single line, modulo $10^9 + 7$.

8.4 输入输出样例

输入样例：

1

11

1 1 1 1 1 1 1

16
1 1 h
2 1 11
2 2 11
1 2 e
1 3 l
1 4 l
1 5 l
2 1 11
1 6 o
1 7 w
2 2 11
1 8 o
1 9 r
1 10 l
1 11 d
2 1 11

输出样例：

667718262
953066461
937670535
0
3

9 String Problem

时间限制：1000ms 空间限制：64MB

9.1 题目描述

Little L raised a question:

Given a string S of length n containing only lowercase letters.

You need to select several non-empty substrings of S so that they are disjoint pairwise, and each substring is a palindrome.

Assuming you have selected K substrings($s_1, s_2 ... s_k$) that satisfy the above conditions, your score is the sum of the lengths of all substrings minus K. It is $\sum_{i=1}^{K} len(s_i) - K$

But Little L is a dedicated person, and to increase difficulty, Little L requires that each palindrome string contain at most one kind of letter

Little L wants you to find the maximum score.

9.2 输入格式

A positive integer T in the first line represents the number of test groups, for each group of test data:

The only line contains a string of length n which containing only lowercase letters.

$T \leq 20, \sum n \leq 10^6$

9.3 输出格式

For each test data, print a number representing the maximum score.

9.4 输入输出样例

输入样例：

2

etxabaxtezwkdwokdbbb

aaaaa

输出样例：

2

4

10 Klee likes making friends

时间限制：1000ms 空间限制：64MB

10.1 题目描述

Klee likes to make friends,now there are n people standing in a line and Klee can make friend with the i^{th} people with a cost of a_i, Klee has a principle that at least 2 out of m consecutive people must be her friends.Please help her calculate the minimum amount of costs she needs.

10.2 输入格式

The first line of the input contains a single integer $T(1 \leq T \leq 20)$indicating the number of test cases.
In each test case:
The first line contains two integers$n, m.(2 \leq n \leq 20000, 2 \leq m \leq 2000, m \leq n)$
The second line contains n intergers $a_1, a_2, ..., a_n$ $(0 \leq a_i \leq 20000)$
It's guarenteed that in all test cases, $\sum n \leq 50000$

10.3 输出格式

For each test case:
You need to print a integers represents the smallest cost.

10.4 输入输出样例

输入样例：
1
7 3
1 5 7 2 1 4 8
输出样例：
13

11 SPY finding NPY

时间限制：1000ms 空间限制：64MB

11.1 题目描述

Recently, SPY has retired from XCPC. He cherishes the memory of learning algorithms from scratch and winning the ICPC gold medal. So he is finding an NPY (non-programming youth) to be his successor. SPY is so popular that n NPYs want to be his apprentice. As SPY only needs one NPY, he sets a test for the n NPYs. The rules are as follows:

n NPYs are indexed from 1 to n. SPY will interview the n NPYs in order. The i-th NPY will be tested in the i-th interview. After an NPY is interviewed, SPY will get her IQ (intelligence quotient) number (an integer in $[0, 2023^{2023^{2023}}]$) . SPY can decide whether to accept her or not. Once he accepts an NPY, the test would be finished and he won't interview the following NPYs. Once he refuses an NPY, he won't give her another chance.

Notice that there are no two NPYs with the same IQ. SPY has a specital strategy to find an NPY with high IQ. He sets an integer k ($0 \le k < n$) before the test.

1、No matter how intelligent the first k NPYs are, they will be refused. SPY will record the highest IQ number x within the first k NPYs. If $k = 0$ then $x = -1$.

2、Then he will interview the $(k+1)$-th to the $(n-1)$-th NPY. Once SPY interviews an NPY with IQ higher than x, he will accept her and finish the test.

3、If no NPY is accepted, SPY will accept the n-th NPY.

The IQ rank of the n NPYs is random, which means their rank is a permutation of $1 \sim n$, and the $n!$ possible situations occur with equal probability. Althouth SPY is a master of useful algorithms, it is difficult for him to set the number k. Can you help him to calculate the **minimum** k to maximize the probability to accept the NPY with the highest IQ?

11.2 输入格式

The first line contains a single integer T ($1 \le T \le 10^4$), indicating the number of test cases.

The next T lines, each line contains a single integer n ($1 \le n \le 10^4$), indicating the number of NPYs.

11.3 输出格式

For each test, you should output one integer in a line, indicating the integer k.

11.4 输入输出样例

输入样例：

```
8
1
2
3
4
9000
9001
9002
9003
```

输出样例：

0

0

1

1

3311

3311

3311

3312

Hint:

In the third test, there are 3 NPYs. Let the array p represent to the IQ rank. The IQ rank of i-th NPY is p_i. The u-th NPY with $p_u = 1$ has the lowest IQ, and the v-th NPY with $p_v = 3$ has the highest IQ.

There are $3! = 6$ situations occur with equal probability. The following list shows the IQ rank of the accepted NPY in all situations, and the probability to accept the NPY with the highest IQ.

	$k = 0$	$k = 1$	$k = 2$
$p = [1, 2, 3]$	1	2	3
$p = [1, 3, 2]$	1	3	2
$p = [2, 1, 3]$	2	3	3
$p = [2, 3, 1]$	2	3	1
$p = [3, 1, 2]$	3	2	2
$p = [3, 2, 1]$	3	1	1
$probability$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{3}$

12 Coin

时间限制：1000ms 空间限制：64MB

12.1 题目描述

Little F went to a party this day, and now little F wants to organize everyone to play a game together!

A total of N people participated in the game (Little F watched them play). Little F prepared N identical coins. At the beginning, each person had exactly one coin, and each person had a number a_i indicating the upper limit of coins he could held.

There are M rounds in the game. In each round, Little F will choose two people A and B (of course, these two people are different), and then these two people have three choices:

1. A gives B a coin 2. B gives A a coin 3. They do nothing

Note that at any time, the number of coins held by each person cannot exceed his holding limit a_i.

Among the N people, there are K people who are friends of Little F, and now little F wants to know, after M rounds of games, in all possible situations, what is the maximum total number of coins held by his K friends.

12.2 输入格式

A positive integer T in the first line represents the number of test groups, for each group of test data:

The first line contains three integers N, M, K. Respectively represent the total number of people, the number of game rounds and the number of friends of Little F.

The next line contains N integers, and the i-th number a_i represents the maximum number of coins held by the i-th person.

In the next M lines, two integers (A, B) are given in each line to indicate that the people selected in this round of the game are A and B

The next line contains K numbers $b_1, b_2...b_k$ represents the K friends of Little F

$1 \leq T \leq 10, 1 \leq N, M, a_i \leq 3000, 1 \leq K \leq N$

12.3 输出格式

For each set of test data, the output line contains an integer, indicating the maximum total number of coins held by the K friends of Little F.

12.4 输入输出样例

输入样例：

2
4 4 1
4 4 4 4
1 4
4 3
3 2
2 1
1
5 4 2
1 3 1 2 2
1 3

3 4
1 5
2 5
2 4
输出样例：
3
4

13 Turret

时间限制：15000ms 空间限制：64MB

13.1 题目描述

Monsters come out of the building. Please set up turrets to resist the monsters.

On a two-dimensional plane, there are four walls forming a square area, with the lower left corner located at $(0, 0)$ and the upper right corner located at $(10000, 10000)$.

There are n buildings in the area, and each building is formed by m points in counterclockwise order. For each building, the first given point is the monster spawn point. Every building is convex and any two buildings do not overlap or contact.

You can set up turrets at any position outside buildings, **including** the edges of the buildings and the walls. Note that the coordinates of the turret **do not** need to be integers. A spawn point can be attacked by a turret when the link of them does not pass inside any building.

Please calculate the minimum number of turrets needed to ensure that each monster spawn point can be attacked by at least one turret.

13.2 输入格式

The first line of the input contains a single integer T $(1 \leq T \leq 10)$, indicating the number of test cases.

In each test case:

The first line contains a single integer n $(1 \leq n \leq 15)$, representing the number of buildings.

For each building:

The first line contains one integer m $(3 \leq m \leq 20)$

Each of the next m lines contains two integer number x_i, y_i $(0 < x_i, y_i < 10000)$, representing the ith point of the building.

It's guarenteed that the for each test case the sum of m do not exceed 200.

13.3 输出格式

Output an integer representing the minimum number of turrets needed.

13.4 输入输出样例

输入样例：

1
4
4
1 1
2 1
2 2
1 2
4
3 3
4 3
4 4
3 4

4
3 1
4 1
4 2
3 2
4
1 3
2 3
2 4
1 4
输出样例：
1
Hint:
You can set up a turret at $(0, 10)$