2017

NCTU Programming Summer Camp

Final Contest

ID	Title	Time Limit (s)	Memory Limit (MB)
A	Buildings	5	512
B	String	1	512
C	Deck	1	512
D	Liar	1	512
E	Victory	5	512
F	Frank	1	512
G	Permutation	1	512

A. Buildings

time limit: 5 seconds
memory limit: 512 MB
input: standard input
output: standard output

Old houses of height 1 were built along the main road. From time to time, new buildings of the shape of exponential function with base m are created. People don't destroy old houses when building new ones; only the taller parts are added. The mayor is interested in the skyline of the city. Please write a program to help him find out how tall a point is.

Input

The first line of the input is an integer T $(1 \leq T \leq 20)$ denoting the number of test cases. Each test case starts with a line containing two integers m $(2 \leq m \leq 10^5)$, the base of exponential functions and n $(1 \leq n \leq 10^5)$, the number of queries. Each of the following n lines contains one of the queries.

- Build a skyscraper on $[l, r]$: $1\ l\ r$ $(1 \leq l < r \leq 10^5, l \equiv r\ mod\ 2)$
- Tell how tall the buildings is: $2\ x$ $(1 \leq x \leq 10^5)$

Output

For each query of second type, output the height modulo $10^9 + 7$ of the building at x.

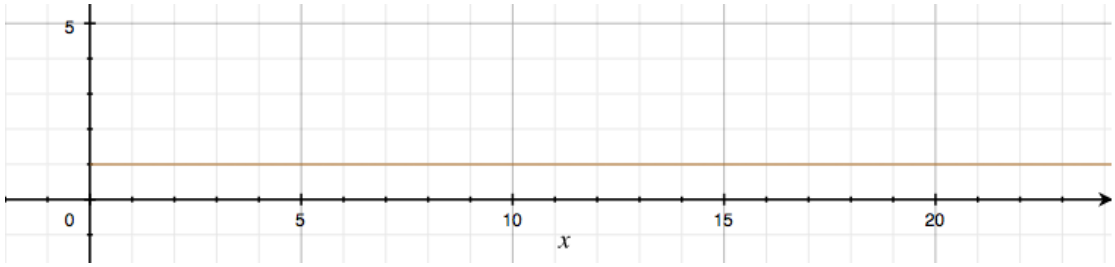
Sample Input

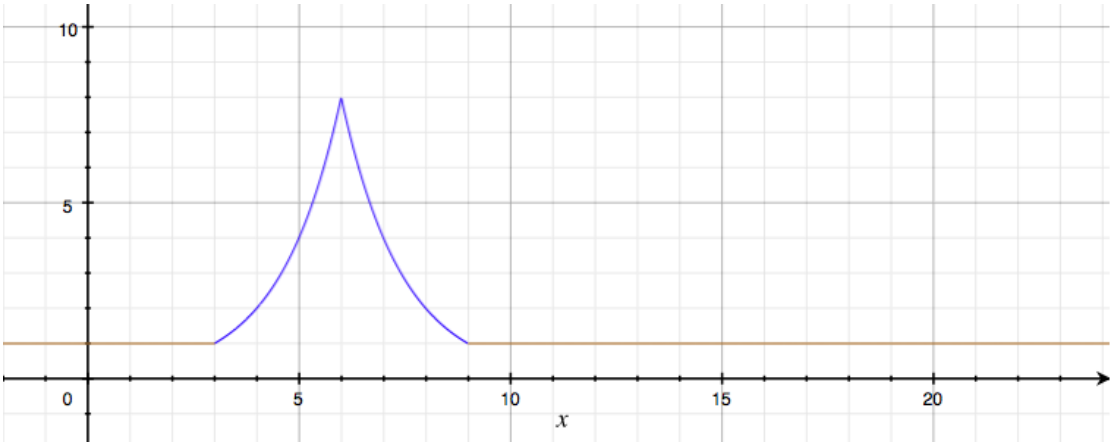
```
1
2 9
2 7
1 3 9
2 7
1 9 13
2 9
2 10
1 4 12
2 7
2 11
```

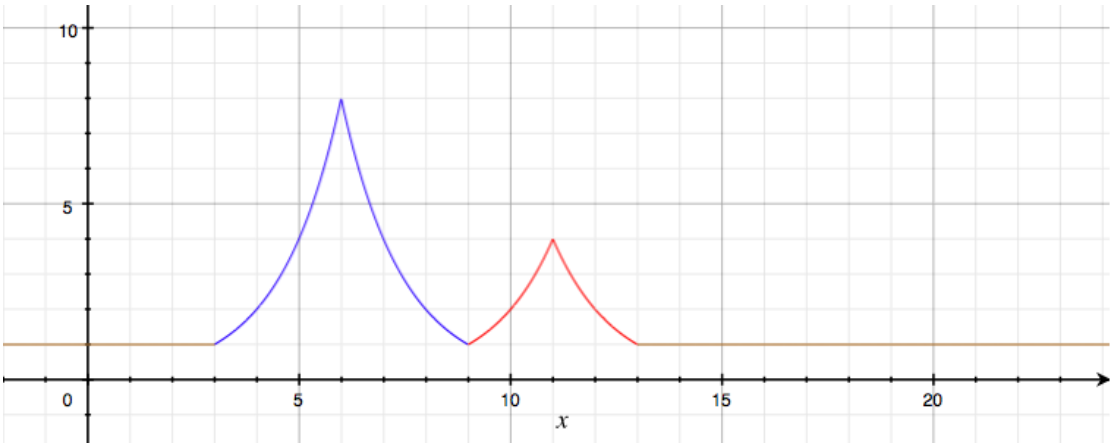
Sample Output

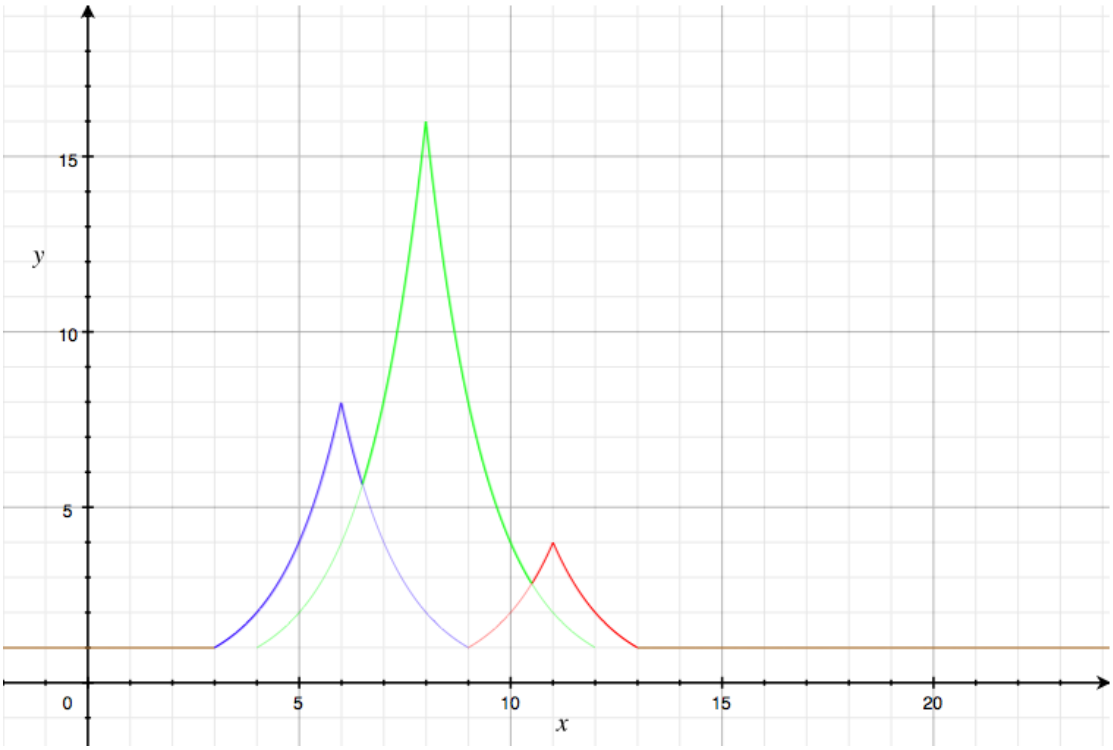
1
4
1
2
8
4

Note








15
y
10
5
0
5
10
15
20
x

B. String

time limit: 1 seconds
memory limit: 512 MB
input: standard input
output: standard output

A string s of length n has a total of $\frac{n(n+1)}{2}$ substrings. Please find a substring t that appears in s most often (overlap of different occurrences are allowed), and output the number of times it appears in s.

Input

The first line of the input is an integer T $(1 \leq T \leq 20)$ denoting the number of test cases. Each of the cases contains two lines. The first line contains a single integer n $(1 \leq n \leq 10^6)$ indicating the length of s. The second line contains s.

Output

For each test case, output in a separate line the number of occurrences of t in s.

Sample Input

```
2
2
ab
4
abab
```

Sample Output

```
1
2
```

C. Deck

time limit: 1 seconds
memory limit: 512 MB
input: standard input
output: standard output

Ian has n cards in his collection. Each of these cards is characterized by its power p_i, magic number c_i and level l_i. Ian wants to build a deck with total power not less than k, but magic numbers may not allow him to do so — Ian can't place two cards in a deck if the sum of their magic numbers is a prime number. Also Ian cannot use a card if its level is greater than the level of Ian's character. At the moment Ian's character's level is 1. Help Ian to determine the minimum level he needs to reach in order to build a deck with the required total power.

Input

The first line of the input is an integer T $(1 \leq T \leq 20)$ denoting the number of test cases. Each of the test case starts with a line containing two integers n and k $(1 \leq n \leq 100, 1 \leq k \leq 100000)$, followed by n lines, each of these lines contains three numbers that represent the corresponding card: p_i, c_i and l_i $(1 \leq p_i \leq 1000, 1 \leq c_i \leq 100000, 1 \leq l_i \leq n)$.

Output

For each test case, output an integer in a separate line. If Ian won't be able to build a deck with required power, print -1. Otherwise print the minimum level Ian has to reach in order to build a deck.

Sample Input

```
1
2 7
4 4 1
5 8 2
```

Sample Output

```
2
```

D. Liar

time limit: 1 seconds
memory limit: 512 MB
input: standard input
output: standard output

MZ drew a shape on a plane. It is either a circle or a square. You wanted to know what shape he had drawn but he wouldn't tell. Instead, he told you the maximum distance between all pair of points on his shape and the area of the shape. Now you want to verify if he's telling the truth by computing the perimeter of the shape.

Input

The first line of the input is an integer T $(1 \leq T \leq 50)$ denoting the number of test cases. Each of the next T lines contains two positive real number d and a, the distance and the area.

Output

For each test case, output a line.

- If the shape is a square, output "Square: p" where p is the perimeter
- If the shape is a circle, output "Circle; p" where p is the perimeter
- Otherwise, output "You liar!"

Sample Input

```
3
2 3.1415926
2 2
1 1
```

Sample Output

```
Circle; 6.2831853
Square: 5.6568542
You liar!
```

Note

If MZ is telling the truth, the error between the real value and the number he told does not exceed 10^{-5}. Your output will be considered correct if its absolute or relative error does not exceed 10^{-5}.

E. Victory

time limit: 5 seconds
memory limit: 512 MB
input: standard input
output: standard output

V stands for victory! On a grid of size $n * n$, there is a number in each cell. You need to choose a V-shape region of the grid and calculate the sum of the numbers in that region. The V-shape region need not to be symmetric. For example, $\{(x-1, y+1), (x, y), (x+1, y+1)\}$ and $\{(x-1, y+1), (x, y), (x+1, y+1), (x+2, y+2)\}$ are both valid regions, but neither of $\{(x, y)\}$ and $\{(x-1, y+1), (x, y)\}$ are valid. What's the maximum sum you can get?

Input

The first line of the input is an integer T $(1 \leq T \leq 20)$ denoting the number of test cases. For each test case, the first line contains an integer n $(3 \leq \mathrm{n} \leq 1000)$. Then n lines follow, each consisting of n space separated integers in the range $[-1000, 1000]$.

Output

For each test case, output in a separate line the maximum sum you can get.

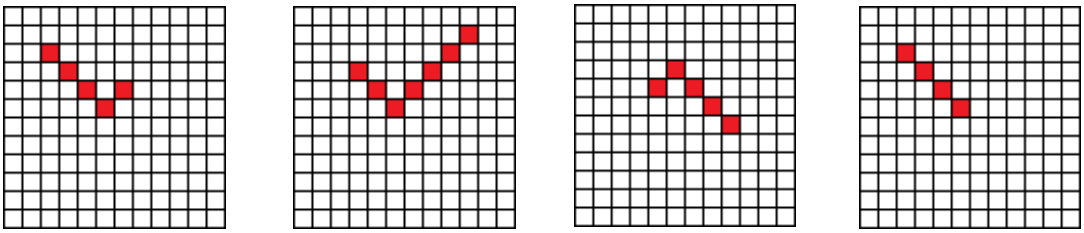
Sample Input

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

Sample Output

```
7
```

Note



The left two images are examples of valid V-shape, while the right two images are examples of invalid V-shape.

F. Frank

time limit: 1 seconds
memory limit: 512 MB
input: standard input
output: standard output

Frank is a Codeforces fanatic. Every night he randomly selects some problems into his circular list. After that, Frank will pick one problem from the circular list as a starting point and starts to solve problems in clockwise order until he feels bored. Specifically, each problem has its own interesting value for Frank. Frank feels bored if and only if the sum of interesting values of problems he read is not positive. In order to solve as many problems as possible, Frank need to know which problem he should start with. Since this problem is too easy for Frank, he wants you to solve it for him.

Input

The first line of the input is an integer T $(1 \leq T \leq 20)$ denoting the number of test cases. For each case, the first line contains a single integer n $(1 \leq T \leq 10^5)$, the number of problems. The next line contains n space separated integers v_i $(|v_i| \leq 10^5, 1 \leq i \leq n)$ – the interesting values of the problems in the circular list in clockwise order.

Output

For each test case, output one line containing two integer s and t separated by a space, denoting the start problem and the last problem he should read to maximize the number of problems solved. If he won't solve any problem, output "0 0" without quotes. If there are multiple solutions, output the one with minimum s.

Sample Input

```
3
4
0 1 0 -2
5
```

```
-1 -2 -3 -4 -5
11
-1 -2 -3 -4 -5 -6 56 -7 -8 -9 -10
```

Sample Output

```
2 3
0 0
7 6
```

G. Permutation

time limit: 1 seconds
memory limit: 512 MB
input: standard input
output: standard output

Given an array of length 10 describing the number of each digit $(0\sim9)$ available, you can pick some of them to obtain a non-negative integer (leading zeroes not allowed, except for 0). How many different non-negative integer x can you obtain such that $i|x\ \forall i \in \{2,3,4,5,6,8,9,10,11\}$.

- Example 1

  Suppose the array is $[1\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0]$, then you have one 0 and one 6 to pick from. The valid non-negative integers you can obtain are $0, 6, 60$.

- Example 2

  Suppose the array is $[0\ 0\ 0\ 2\ 0\ 0\ 0\ 0\ 0\ 0]$, then you have two 3 to pick from. The valid non-negative integers you can obtain are $3, 33$.

- Example 3

  Suppose the array is $[2\ 0\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 0]$, then you have two 0 and one 3 to pick from. The valid non-negative integers you can obtain are $0, 3, 30, 300$.

- Example 4

  Suppose the array is $[0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1]$, then you have only one 9 to pick from. The only valid non-negative integer you can obtain is 9.

Input

The first line of the input is an integer $T\ (1 \le T \le 50)$ denoting the number of test cases. For each case, there is a line containing ten space separated integers $k_i (0 \le k_i \le 9)$, meaning that you can use digit i for at most $k_i$times to obtain the non-negative integer x. It's guaranteed that $1 \le \sum k_i \le 9$.

Output

For each test case, output the number of different x you can obtain such that the above condition is satisfied.

Sample Input

```
2
2 0 0 0 5 2 0 0 0 0
1 2 1 0 0 1 0 1 1 0
```

Sample Output

```
5
3
```

Note

- Divisibility by 2: A number is divisible by 2 if and only if its last digit is divisible by 2 or in other words, is even.
- Divisibility by 3: A number is divisible by 3 if and only if the sum of its digits is divisible by 3.
- Divisibility by 4: A number is divisible by 4 if and only if its last two digits form a number that is divisible by 4.
- Divisibility by 5: A number is divisible by 5 if and only if its last digit equals 5 or 0.
- Divisibility by 6: A number is divisible by 6 if and only if it is divisible by 2 and 3 simultaneously (that is, if the last digit is even and the sum of all digits is divisible by 3).
- Divisibility by 8: A number is divisible by 8 if and only if its last three digits form a number that is divisible by 8.
- Divisibility by 9: A number is divisible by 9 if and only if the sum of its digits is divisible by 9.
- Divisibility by 10: A number is divisible by 10 if and only if its last digit is a zero.
- Divisibility by 11: A number is divisible by 11 if and only if the sum of digits on its odd positions either equals to the sum of digits on the even positions, or they differ in a number that is divisible by 11.