A - Octave

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 100 points

Problem Statement

The frequency of a sound doubles for every increase of 1 octave in pitch.

If the pitch of a sound with frequency X hertz is raised by Y octaves, what will its frequency be in hertz?

Constraints

- $1 \leq X \leq 444$
- $1 \leq Y \leq 3$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$X \quad Y$$

Output

Output the answer as an integer in one line. Omit the unit (hertz).

Sample Input 1

```
110 2
```

Sample Output 1

```
440
```

For a sound 2 octaves above a sound with frequency 110 hertz, its frequency is $110 \times 2 \times 2 = 440$ hertz.

Sample Input 2

```
233 3
```

Sample Output 2

```
1864
```

For a sound 3 octaves above a sound with frequency 233 hertz, its frequency is $233 \times 2 \times 2 \times 2 = 1864$ hertz.

Sample Input 3

```
432 1
```

Sample Output 3

```
864
```

For a sound 1 octave above a sound with frequency 432 hertz, its frequency is $432 \times 2 = 864$ hertz.

B - Trifecta

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 200 points

Problem Statement

N horses numbered 1 to N had a race.

All horses started simultaneously, and horse i took T_i seconds from the start to the goal.

Find the numbers of the horses that finished in 1st, 2nd, and 3rd places. It is guaranteed that all T_i are distinct.

Constraints

- $3 \leq N \leq 32$
- $1 \leq T_i \leq 200$
- All T_i are distinct.
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
T_1 \quad \ldots \quad T_N
\end{array}
$$

Output

Output the numbers of the horses that finished in 1st, 2nd, and 3rd places, in this order, separated by spaces.

Sample Input 1

```
4
100 110 105 95
```

Sample Output 1

```
4 1 3
```

The horses finished in the order $4, 1, 3, 2$. Output the numbers for 1st, 2nd, and 3rd places, which are $4, 1, 3$, in this order, separated by spaces.

Sample Input 2

```
8
72 74 69 70 73 75 71 77
```

Sample Output 2

```
3 4 7
```

C - Striped Horse

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 300 points

Problem Statement

> Ringo-san is trying to paint a horse with stripes to make it a zebra.

There are N squares numbered 1 to N arranged in a line.
Initially, all squares are white, and the cost of painting square i black is C_i.

Consider performing the following procedure once to paint some squares black:

- Choose a positive integer x freely.
- Paint square i black for all integers i satisfying $1 \leq i \leq N$ such that the remainder when $(i + x)$ is divided by $2W$ is less than W.

Find the minimum total cost to perform this procedure.

You are given T test cases; solve each of them.

Constraints

- $1 \leq T \leq 2 \times 10^5$
- $1 \leq N \leq 2 \times 10^5$
- $1 \leq W \leq 2 \times 10^5$
- $1 \leq C_i \leq 10^9$
- The sum of N over the T test cases is at most 2×10^5.
- The sum of W over the T test cases is at most 2×10^5.
- All input values are integers.

Input

The input is given from Standard Input in the following format, where case_i denotes the i-th test case.

$$T$$
$$\mathrm{case}_1$$
$$\vdots$$
$$\mathrm{case}_T$$

Each test case is given in the following format:

$$N \quad W$$
$$C_1 \quad \ldots \quad C_N$$

Output

Output T lines. The i-th line should contain the answer for the i-th test case.

Sample Input 1

```
4
8 2
1 10 10 1 1 10 10 1
8 3
1 10 10 1 1 10 10 1
8 4
1 10 10 1 1 10 10 1
4 100
100000 100000 100000 100000
```

Sample Output 1

```
4
12
22
0
```

In the first test case, if the procedure is executed with $x = 4$, squares $1, 4, 5, 8$ are painted black, and the total cost is 4. The total cost cannot be less than 4, so this is the minimum.

D - Forbidden List 2

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 400 points

Problem Statement

There is a list consisting of N distinct integers. The i-th ($1 \leq i \leq N$) integer in the list is A_i.

You are given Q questions; answer each of them. The j-th question ($1 \leq j \leq Q$) is as follows:

- Find the Y_j-th smallest value among the integers greater than or equal to X_j that are not in the list.

Constraints

- $1 \leq N \leq 3 \times 10^5$
- $1 \leq Q \leq 3 \times 10^5$
- $1 \leq A_i \leq 10^9$ ($1 \leq i \leq N$)
- $A_1, \ldots, A_N$ are distinct.
- $1 \leq X_j \leq 10^9$ ($1 \leq j \leq Q$)
- $1 \leq Y_j \leq 10^9$ ($1 \leq j \leq Q$)
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{aligned}
&N \quad Q \\
&A_1 \quad \cdots \quad A_N \\
&X_1 \quad Y_1 \\
&\vdots \\
&X_Q \quad Y_Q
\end{aligned}
$$

Output

Output Q lines. The j-th line ($1 \leq j \leq Q$) should contain the answer to the j-th question.

Sample Input 1

```
5 4
16 9 2 3 1
6 10
12 4
1 1
1000000000 1000000000
```

Sample Output 1

```
17
15
4
1999999999
```

For the first question, the smallest 10 integers greater than or equal to 6 that are not in the list are $6,\ 7,\ 8,\ 10,\ 11,\ 12,\ 13,\ 14,\ 15,\ 17$. Thus, the answer to the question is 17.

For the second question, the smallest 4 integers greater than or equal to 12 that are not in the list are $12,\ 13,\ 14,\ 15$. Thus, the answer to the question is 15.

For the third question, the 1st smallest value among the integers greater than or equal to 1 that are not in the list is 4.

For the fourth question, the $1\,000\,000\,000$-th smallest value among the integers greater than or equal to $1\,000\,000\,000$ that are not in the list is $1\,999\,999\,999$.

Sample Input 2

```
10 10
284008711 658403910 982178205 50598815 694147827 230009803 763277509 509451676 821970166 284008710
740250292 159734720
255870361 8400028
23659634 718117163
697334729 301140741
698853172 270344164
713418715 285312509
50065000 52368934
46642556 591869945
607623561 273664826
482426028 265015448
```

Sample Output 2

```
899985013
264270388
741776803
998475472
969197337
998731226
102433934
638512505
881288390
747441478
```

E - Cookies

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 450 points

Problem Statement

There are 10^{100} cookies of each of N types. The deliciousness per cookie of the i-th type is A_i.

You will choose a total of K cookies from these. Two ways of choosing cookies are considered the same if and only if the multisets of types of chosen cookies match.

For each of the $\binom{N+K-1}{K}$ ways of choosing, consider the sum of deliciousness of the chosen cookies. Let $S_1, S_2, \ldots$ be these sums in descending order, with duplicates included. Find $S_1, \ldots, S_X$.

Constraints

- $1 \leq N \leq 50$
- $1 \leq K \leq 10^5$
- $1 \leq X \leq \min\left(10^5, \binom{N+K-1}{K}\right)$
- $-10^9 \leq A_i \leq 10^9$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{lll}
N & K & X \\
A_1 & \ldots & A_N
\end{array}
$$

Output

Let $S_1, S_2, \ldots$ be the possible sums of deliciousness of the chosen K cookies in descending order, with duplicates included. Output $S_1, \ldots, S_X$ in this order, separated by newlines.

Sample Input 1

```
2 4 3
20 10
```

Sample Output 1

```
80
70
60
```

There are 5 ways to choose 4 cookies: "k cookies of type 1 and $4-k$ cookies of type 2" $(0 \leq k \leq 4)$, where the sums of deliciousness of the chosen cookies are $80, 70, 60, 50, 40$.

Sample Input 2

```
3 100000 5
-1 -1 -1
```

Sample Output 2

```
-100000
-100000
-100000
-100000
-100000
```

Different ways of choosing cookies may result in the same sum of deliciousness.

Sample Input 3

```
9 14142 13
31 41 59 26 53 58 97 93 23
```

Sample Output 3

```
1371774
1371770
1371766
1371762
1371758
1371754
1371750
1371746
1371742
1371738
1371736
1371735
1371734
```

F - Egoism

Time Limit: 2 sec / Memory Limit: 1024 MiB

Score : 550 points

Problem Statement

> At AtCoder Ranch, horses bathe themselves. Some horses clean up afterward, while others leave a mess.

There are N horses, numbered 1 to N. Horse i ($1 \leq i \leq N$) has a **mood** A_i and a **tidiness** B_i (where B_i is 1 or 2).

At night, the N horses bathe once each. Let p_j be the number of the horse that bathes in the j-th turn ($1 \leq j \leq N$). The **satisfaction** of horse p_j is given as follows:

- If $j \geq 2$, the product of the mood of horse p_j and the tidiness of horse p_{j-1}.
- If $j = 1$, the mood of horse p_j.

You will be given Q queries, one for each day over Q days. Process them in order. The query for the k-th day ($1 \leq k \leq Q$) is as follows:

- Change the mood of horse W_k to X_k and its tidiness to Y_k (where Y_k is 1 or 2). Then, find the maximum possible total satisfaction of the N horses' baths that night if you can decide the order in which the N horses bathe arbitrarily.

Constraints

- $1 \leq N \leq 2 \times 10^5$
- $1 \leq Q \leq 2 \times 10^5$
- $1 \leq A_i \leq 10^6$ ($1 \leq i \leq N$)
- B_i is 1 or 2. ($1 \leq i \leq N$)
- $1 \leq W_k \leq N$ ($1 \leq k \leq Q$)
- $1 \leq X_k \leq 10^6$ ($1 \leq k \leq Q$)
- Y_k is 1 or 2. ($1 \leq k \leq Q$)
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \quad Q \\
A_1 \quad B_1 \\
\vdots \\
A_N \quad B_N \\
W_1 \quad X_1 \quad Y_1 \\
\vdots \\
W_Q \quad X_Q \quad Y_Q
\end{array}
$$

Output

Output Q lines. The k-th line ($1 \leq k \leq Q$) should contain the answer to the query for the k-th day.

Sample Input 1

```
4 4
1 2
10 1
3 1
7 2
2 7 1
1 3 1
2 2 1
3 1000000 2
```

Sample Output 1

```
32
27
18
2000015
```

For the query on the first day, if the horses bathe in the order $3, 1, 4, 2$, the satisfaction of each horse is as follows:

- Horse 3's satisfaction is 3
- Horse 1's satisfaction is $1 \times 1 = 1$
- Horse 4's satisfaction is $7 \times 2 = 14$
- Horse 2's satisfaction is $7 \times 2 = 14$

The total satisfaction in this case is 32.

For the query on the second day, if the horses bathe in the order $4, 2, 1, 3$, the satisfaction of each horse is as follows:

- Horse 4's satisfaction is 7
- Horse 2's satisfaction is $7 \times 2 = 14$
- Horse 1's satisfaction is $3 \times 1 = 3$
- Horse 3's satisfaction is $3 \times 1 = 3$

The total satisfaction in this case is 27.

For the query on the third day, if the horses bathe in the order $4, 3, 2, 1$, the total satisfaction is 18.

For the query on the fourth day, if the horses bathe in the order $4, 3, 1, 2$, the total satisfaction is 2000015.

Sample Input 2

```
10 10
340019 2
598908 1
177330 2
575439 2
916653 2
451275 2
762769 2
36625 2
273231 2
367619 2
8 334230 2
8 835378 1
4 777076 2
6 61501 1
4 516395 1
4 35678 2
5 751493 1
3 815798 1
2 369777 2
5 941470 2
```

Sample Output 2

```
9417616
10146681
10549955
9708906
8847525
8463663
7860112
8606740
8516097
9236070
```

G - Haunted House

Time Limit: 3 sec / Memory Limit: 1024 MiB

Score : 600 points

Problem Statement

Halloween season has arrived. As a test of courage, you are about to challenge a haunted house.

There is a haunted house with F floors, and each floor is represented by an H-row by W-column grid. Let cell (k, i, j) be the cell at the i-th row from the top and j-th column from the left of the grid representing floor k.

The content of cell (k, i, j) is represented by the character $S_{k,i,j}$ as follows:

- If $S_{k,i,j}$ is a digit (`0` - `9`), cell (k, i, j) is an empty cell with a number of coins equal to the single-digit integer represented by the digit.
- If $S_{k,i,j}$ is `#`, cell (k, i, j) is a wall cell.

You can move directly from an empty cell (k, i, j) to another cell (k', i', j') if cell (k', i', j') is an empty cell and one of the following is satisfied:

- $k' = k$ and $|i' - i| + |j' - j| = 1$ are satisfied.
- $(k', i', j') = (k + 1, i, j)$ is satisfied.
- $(k', i', j') = (k - 1, i, j)$ is satisfied, and a **ladder** is placed at cell (k, i, j).

You cannot move outside the grid.

You are given Q questions; answer each of them. The i-th question ($1 \leq i \leq Q$) is as follows:

- Find the maximum total number of coins in the cells you visit if you choose exactly one empty cell and place a ladder there, start at cell (G_i, A_i, B_i), and repeatedly move.

Each question is independent. That is, a ladder placed for one question does not affect other questions.

Constraints

- F, H, W are integers.
- $1 \leq F \leq 10$
- $1 \leq H, W \leq 500$
- $S_{k,i,j}$ is either a digit (`0`, `1`, `2`, `3`, `4`, `5`, `6`, `7`, `8`, `9`) or `#`. ($1 \leq k \leq F, 1 \leq i \leq H, 1 \leq j \leq W$)
- Q is an integer.
- $1 \leq Q \leq 10^5$
- G_i, A_i, B_i are integers. ($1 \leq i \leq Q$)
- $1 \leq G_i \leq F$ ($1 \leq i \leq Q$)
- $1 \leq A_i \leq H$ ($1 \leq i \leq Q$)
- $1 \leq B_i \leq W$ ($1 \leq i \leq Q$)
- S_{G_i,A_i,B_i} is not `#`. ($1 \leq i \leq Q$)

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
F \quad H \quad W \\
S_{1,1,1}S_{1,1,2}\cdots S_{1,1,W} \\
S_{1,2,1}S_{1,2,2}\cdots S_{1,2,W} \\
\vdots \\
S_{1,H,1}S_{1,H,2}\cdots S_{1,H,W} \\
S_{2,1,1}S_{2,1,2}\cdots S_{2,1,W} \\
S_{2,2,1}S_{2,2,2}\cdots S_{2,2,W} \\
\vdots \\
S_{2,H,1}S_{2,H,2}\cdots S_{2,H,W} \\
\vdots \\
S_{F,1,1}S_{F,1,2}\cdots S_{F,1,W} \\
S_{F,2,1}S_{F,2,2}\cdots S_{F,2,W} \\
\vdots \\
S_{F,H,1}S_{F,H,2}\cdots S_{F,H,W} \\
Q \\
G_1 \quad A_1 \quad B_1 \\
G_2 \quad A_2 \quad B_2 \\
\vdots \\
G_Q \quad A_Q \quad B_Q
\end{array}
$$

Output

Output Q lines. The i-th line ($1 \leq i \leq Q$) should contain the answer to the i-th question.

Sample Input 1

```
3 5 6
######
##5###
#1#11#
#1#11#
######
######
##313#
#4##1#
#242##
######
######
#0####
##4###
###99#
######
3
1 2 3
2 3 2
3 2 2
```

Sample Output 1

```
47
34
0
```

For the first question, by placing a ladder at cell $(2,3,5)$ and moving as follows, you can make the total number of coins in the visited cells 47:

1. Start at cell $(1,2,3)$ on the first floor.
2. Go up to the second floor and move in the order $(2,2,3) \to (2,2,4) \to (2,2,5) \to (2,3,5)$.
3. Go down to the first floor and move in the order $(1,3,5) \to (1,4,5) \to (1,4,4) \to (1,3,4) \to (1,4,4)$.
4. Go up to the second floor and move in the order $(2,4,4) \to (2,4,3) \to (2,4,2) \to (2,3,2) \to (2,4,2) \to (2,4,3) \to (2,4,4)$.
5. Go up to the third floor and move in the order $(3,4,4) \to (3,4,5)$.

For the second question, by placing a ladder at cell $(2,4,4)$ and moving as follows, you can make the total number of coins in the visited cells 34:

1. Move in the order $(2,3,2) \to (2,4,2) \to (2,4,3) \to (2,4,4)$ on the second floor.
2. Go down to the first floor and move in the order $(1,4,4) \to (1,4,5) \to (1,3,5) \to (1,3,4) \to (1,4,4)$.
3. Go up to the second floor and be at cell $(2,4,4)$.
4. Go up to the third floor and move in the order $(3,4,4) \to (3,4,5)$.

For the third question, no matter how you place the ladder, you cannot move from the starting cell $(3,2,2)$.

Sample Input 2

```
3 6 30
31#990#7###71298#6#9####3##04#
###66###6######7#4#3#1079239#6
891###8#73#7#5838#589#225#1282
#####5#306#9#38#50##6#71##8658
6#68#4#5###30###3#5716987#####
06#8####3#####134#5##2##28#169
#5#11####0#75#8######28#6#904#
85#71####0##9#7#0##8#2##00####
#1#09#3772#838#67#6##261###0#4
#4##78###7#########9#9#488##08
43#1##4632##46714436##849#77#6
8#83#61##4#14476#9#1##72#3####
5#99####52##6#9####9#4#4132#3#
##910#27529#69##491###0#06#9##
0#60845#02##28##610#2####25#6#
#229156374##47#30907#60#####28
#0#810#54###21##1#3###8#5917##
###6#56#60##04#73###6255###2##
10
3 4 23
3 2 4
1 4 14
2 3 12
1 1 16
1 2 9
3 5 6
1 3 19
2 2 4
3 4 9
```

Sample Output 2

```
136
217
474
255
474
285
217
451
251
217
```