Problem A. Group Assignments

Problem Description

There are n students in the class, if the i^{th} and j^{th} students are in the same group, there will be a *badness* of $c_{i,j}$.

KCW asks TAs to divide the students into several groups such that each group has at least 2 students, and he wants the sum of badness among each group be the smallest possible.

Please tell TAs the minimum possible sum of badness, and also provide any group assignment.

Input Format

- line 1: n
- line $1+i$ ($1 \leq i \leq n$): $c_{i,1}\ \ c_{i,2}\ \ \ldots\ \ c_{i,n}$

Output Format

- line 1: ans : minimum possible sum of badness.
- line 2: $g_1\ \ g_2\ \ \ldots\ \ g_n$: the group which i^{th} student belongs to.
  - Your output should satisfy $1 \leq g_i \leq n$ for $i = 1, 2, \ldots, n$.

Constraints

- $2 \leq n \leq 21$.
- $0 \leq c_{i,j} \leq 99$ for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$.
- $c_{i,i} = 0$ for $i = 1, 2, \ldots, n$.
- $c_{i,j} = c_{j,i}$ for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, n$.
- All input values are integers.

Subtasks

1. (20 points) $n \leq 9$.
2. (20 points) $n \leq 12$.
3. (20 points) $n \leq 15$.
4. (20 points) $n \leq 18$.
5. (20 points) No additional constraints.

- If you output the minimum sum of badness correctly, you can get 50% of a subtask's score;
- furthermore, if you construct a group assignment that yields minimum sum of badness, you can get the other 50% score.
- If you can not construct the grouping, please still print any valid construction.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1-4	1500	262144
1	1-28	1500	262144
2	1-43	1500	262144
3	1-58	1500	262144
4	1-73	1500	262144
5	1-99	1500	262144

Samples

Sample Input 1

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

This sample input satisfies the constraints of all the subtasks.

Sample Output 1

```
6
1 1 1
```

Since there must be at least 2 students in each group, there is only one way to assign the students to groups. The total badness is $c_{1,2} + c_{1,3} + c_{2,3} = 1 + 2 + 3 = 6$.

Sample Input 2

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

This sample input satisfies the constraints of all the subtasks.

Sample Output 2

```
7
2 4 4 2
```

The total badness is $(c_{1,4}) + (c_{2,3}) = 4 + 3 = 7$.

Note that both $\{(1,3),(2,4)\}$ and $\{(1,2),(3,4)\}$ has total badness 7, these two group assignments are also considered correct.

Sample Input 3

```
5
0 2 7 1 8
2 0 2 8 1
7 2 0 8 2
1 8 8 0 8
8 1 2 8 0
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 3

```
6
1 2 2 1 2
```

Sample Input 4

```
9
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 4

```
0
2 1 4 7 4 7 1 2 1
```

You are not required to minimize the number of groups.

Problem B. Knapsack Problem (Real)

- 2023.12.08 03:30 Update: Strengthen tests, rejudged solutions.
- 2023.12.06 18:30 Update: Memory Limit 4 MiB $\to$ 16 MiB, rejudged solutions.

Problem Description

There are n items, each item i has a cost c_i and a value v_i, and can be picked infinitely many times. Let S be the multiset of items you picked, please find the maximum value of $\sum_{i \in S} v_i$ under the constraint of $\sum_{i \in S} c_i \le V$.

There are t test cases.

Input Format

- line 1: t

t blocks:

- line 1: $n \;\; V$
- line 2: $c_1 \;\; c_2 \;\; \ldots \;\; c_n$
- line 3: $v_1 \;\; v_2 \;\; \ldots \;\; v_n$

Output Format

t blocks:

- line 1: ans

Constraints

- $1 \le t \le 10$.
- $1 \le n \le 100\,000$.
  - Sum of n across all test cases $\le 100\,000$.
- $1 \le V \le 10^9$.
- $1 \le c_i \le 1200$ for $i = 1, 2, \ldots, n$.
- $1 \le v_i \le 10^9$ for $i = 1, 2, \ldots, n$.
- All input values are integers.

Subtasks

1. (25 points) $c_i \le 20$ for $i = 1, 2, \dots, n$; $V \le 20$.
2. (35 points) $c_i \le 100$ for $i = 1, 2, \dots, n$; $V \le 500$.
3. (15 points) $c_i \le 100$ for $i = 1, 2, \dots, n$; $V \le 60\,000$.
4. (10 points) $c_i \le 100$ for $i = 1, 2, \dots, n$.
5. (10 points) $v_i = c_i$ for $i = 1, 2, \dots, n$.
6. (5 points) No additional constraints.

- You can get 60% of a subtask's score if you use no more than 256 MiB of memory;
- furthermore, you can get another 40% if you use no more than 16 MiB.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1	2000	262144
1	2	2000	262144
2	2-4	2000	262144
3	2-7	2000	262144
4	2-9	2000	262144
5	10-11	2000	262144
6	1-15	2000	262144

Samples

Sample Input 1

```
4
5 23
5 9 8 8 3
1 11 9 6 3
3 8
3 4 5
7 5 6
1 3
1
1000000000
1 1199
1200
1000000000
```

This sample input satisfies the constraints of Subtask 6.

Sample Output 1

```
26
14
3000000000
0
```

- In the first test case, one possible solution is to pick the multiset $\{2, 3, 5, 5\}$, which has a total cost of $23 = c_2 + c_3 + c_5 + c_5$ and a total value of $26 = v_2 + v_3 + v_5 + v_5$.
- In the second test case, the best solution is to pick the multiset $\{1, 1\}$ with cost 6 and value 14.
- In the third test case, the best solution is to pick the multiset $\{1, 1, 1\}$ with cost 3 and value $3 \cdot 10^9$.
- In the fourth test case, you can not afford any item, thus the maximum value is 0.

Problem C. Longest Common Subsequence

Problem Description

Given two **permutations** a and b of $1, 2, \ldots, n$, please find **any** subsequence $a' \subseteq a$ and $b' \subseteq b$ such that

1. $|a'| = |b'| = m$,
2. $a'_i \neq b'_i$ for $i = 1, 2, \ldots, m$,
3. m is maximized, i.e., there does not exist any answer with length $m + 1$.

There are t test cases.

Input Format

- line 1: t

t blocks:

- line 1: n
- line 2: $a_1 \ a_2 \ \ldots \ a_n$
- line 3: $b_1 \ b_2 \ \ldots \ b_n$

Output Format

t blocks:

- line 1: m
- line 2: $a'_1 \ a'_1 \ \ldots \ a'_m$
- line 3: $b'_1 \ b'_2 \ \ldots \ b'_m$

Constraints

- $1 \leq t \leq 10$.
- $2 \leq n \leq 1\,000\,000$.
  - Sum of n across all test cases $\leq 1\,000\,000$.
- $a_1, a_2, \ldots, a_n$ is a permutation of $1, 2, \ldots, n$.
- $b_1, b_2, \ldots, b_n$ is a permutation of $1, 2, \ldots, n$.
- All inputs values are integers.

Subtasks

1. (20 points) $n \leq 16$.
2. (60 points) $n \leq 1000$.
3. (10 points) $n \leq 40\,000$.
4. (10 points) No additional constraints.

- If you answered the optimal length m correctly, you can get 50% of a subtask's score;
- furthermore, if your construction is correct, you can get the other 50% of the score.
- Note: The second and third line should not be left blank. You can simply print m 1s in both line if you are not able to construct the answer.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1	1500	262144
1	1-7	1500	262144
2	1-12	1500	262144
3	1-18	1500	262144
4	1-32	1500	262144

Samples

Sample Input 1

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

This sample input satisfies the constraints of all the subtasks.

Sample Output 1

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

In the first test case, $a' = [3, 4, 1]$ is a subsequence of $a = [3, 5, 4, 1, 2]$, and $b' = [5, 1, 2]$ is a subsequence of $b = [3, 4, 5, 1, 2]$. The length of a' and b' is 3, which is the maximum possible length.

Choosing $a' = [1, 2, 3, 4]$ and $b' = [2, 3, 4, 5]$ is not a valid answer, since a' and b' are not subsequences of a and b, respectively.

Choosing $a' = [3, 5, 1]$ and $b' = [4, 5, 2]$ is also not a valid answer, since $a'_2 = b'_2$.

Problem D. Longest Increasing Subsequence

- 2023.12.06 19:30 Update: Strengthen test cases.

Problem Description

> We assume all the set notations $S \subseteq \{0, 1, \ldots, n-1\}$ in the following context.
>
> We assume $S = \{s_0, s_1, \ldots, s_{k-1}\}$ and $s_0 < s_1 < \cdots < s_{k-1}$ in the following context.

Given an array $a = [a[0], a[1], \ldots, a[n-1]]$, please answer the following question:

- A subset S is ***increasing*** if $a[s_0] \le a[s_1] \le \cdots \le a[s_{k-1}]$.
- A subset S is ***valid*** if S is *increasing* and there does not exist $S' \supsetneq S$ and S' is *increasing*.
- The score of a subset S is $a[s_0] + a[s_1] + \cdots + a[s_{k-1}]$.

Find the numbers of all valid subsets and sum of scores of all valid subsets, both modulo $1\,000\,000\,007$.

There are t test cases.

Input Format

- line 1: t

t blocks:

- line 1: n
- line 2: $a[0]\ \ a[1]\ \ \ldots\ \ a[n-1]$

Output Format

t blocks:

- line 1: $count\ \ sum$

Constraints

- $1 \le t \le 10$.
- $2 \le n \le 100\,000$.
  - Sum of n across all test cases $\le 200\,000$.
- $1 \le a[i] \le 10^9$ for $i = 0, 1, \ldots, n-1$.

Subtasks

1. (10 points) $n \leq 15$.
2. (20 points) $n \leq 200$.
3. (35 points) $n \leq 2000$.
4. (25 points) $a[i] \leq 15$ for $i = 0, 1, \ldots, n-1$.
5. (10 points) No additional constraints.

- If you find the numbers of all valid subsets correctly, you can get 80% of a subtask's score;
- furthermore, if you output the sum of scores of all valid subsets correctly, you can get the remaining 20%.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1	1500	262144
1	1-10	1500	262144
2	1-23	1500	262144
3	1-35	1500	262144
4	1, 36-41	1500	262144
5	1-58	1500	262144

Samples

Sample Input 1

```
6
5
2 3 6 4 6
5
1 15 8 2 4
7
5 3 4 1 7 2 6
2
2 1
3
15 15 15
15
1 5 3 2 8 6 4 13 11 9 14 12 10 7 15
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 1

```
2 32
3 32
6 67
2 3
1 45
53 2478
```

- In the first test case, $\{0, 1, 2, 4\}$ and $\{0, 1, 3, 4\}$ are the only valid subsets. The score for these two subsets are $17 = 2 + 3 + 6 + 6$ and $15 = 2 + 3 + 4 + 6$, respectively.
- In the second test case, $\{0, 1\}$, $\{0, 2\}$, and $\{0, 3, 4\}$ are the only valid subsets.
- In the third test case, $\{0, 4\}$, $\{0, 6\}$, $\{1, 2, 4\}$, $\{1, 2, 6\}$, $\{3, 4\}$, and $\{3, 5, 6\}$ are the only valid subsets.
- In the fourth test case, $\{0\}$ and $\{1\}$ are the only valid subsets.
- In the fifth test case, $\{0, 1, 2\}$ is the only valid subset.

Problem E. Maximum Sum Path

Problem Description

Given a graph G with n vertices numbered 1 to n and m directed edges (u_j, v_j). Each directed edge (u_j, v_j) is colored black or white, denoted as c_j ($c_j \in \{\texttt{B}, \texttt{W}\}$). Each room has a requirement ℓ_i, meaning you can leave room i only if you entered room i with star count $\geq \ell_i$. Each edge has a certain number of stars w_j that can be obtained the first time it is traversed.

You can choose any room i with $\ell_i = 0$ as the starting room and a color as the starting color. After that, you must alternate between black and white edges.

What is the maximum number of stars you can collect? For each room i, please output the maximum number of stars $star_i$ that can be collected if you end up in room i.

It is guaranteed that even with an infinite number of stars, it is not possible to traverse this graph infinitely.

Input Format

- line 1: $n \ m$
- line 2: $\ell_1 \ \ell_2 \ \ldots \ \ell_n$
- line $2 + j$ ($1 \leq j \leq m$): $u_j \ v_j \ c_j \ w_j$

Output Format

- line 1: $star_1 \ star_2 \ \ldots \ star_n$

Constraints

- $1 \leq n \leq 100\,000$.
- $1 \leq m \leq 300\,000$.
- $0 \leq \ell_i \leq 10^9$ for $i = 1, 2, \ldots, n$.
- $1 \leq u_j, v_j \leq n$ for $j = 1, 2, \ldots, m$.
- $0 \leq w_j \leq 10\,000$ for $j = 1, 2, \ldots, m$.
- $c_j \in \{\texttt{B}, \texttt{W}\}$ for $j = 1, 2, \ldots, m$.
- It is guaranteed that even with an infinite number of stars, it is not possible to traverse this graph infinitely.
- All input values except c_js are integers.

Subtasks

1. (15 points) $n \leq 5$; $m \leq 10$.
2. (40 points) $n \leq 500$; $m \leq 2000$.
3. (45 points) No additional constraints.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1-7	1000	262144
1	1-22	1000	262144
2	1-37	1000	262144
3	1-57	1000	262144

Samples

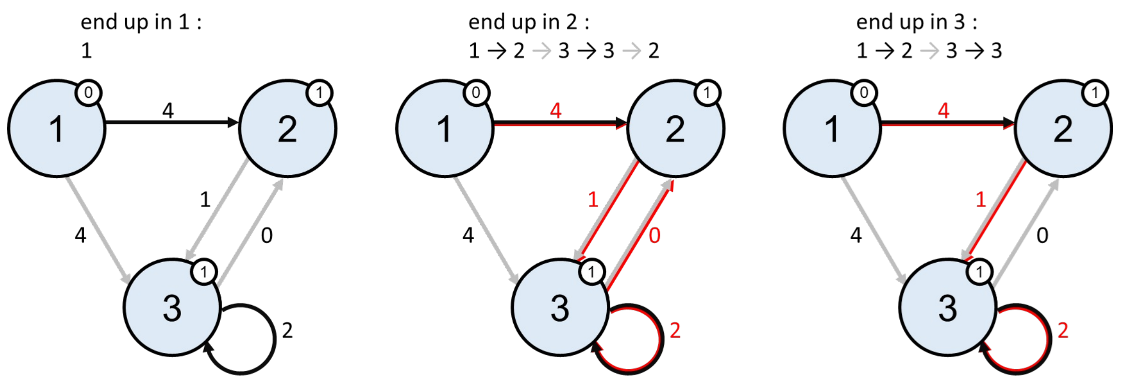
Sample Input 1

```
3 5
0 1 1
1 2 B 4
2 3 W 1
3 2 W 0
3 3 B 2
1 3 W 4
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 1

```
0 7 7
```



Sample Input 1

You can only start from room 1.

- There is no way to go back to room 1 again, thus the maximum star you can collect if you end in room 1 is 0.
- You can walk through black edge 1 to earn 4 stars, white edge 2 to earn 1 star, black edge 4 to earn 2 stars, and white edge 3 to earn 0 stars. The maximum star you can collect if you end in room 2 is 7.
- You can walk through edges 1, 2, 4 to earn 7 stars and end in room 3.

Sample Input 2

```
3 5
0 5 1
1 2 B 4
2 3 W 1
3 2 W 0
3 3 B 2
1 3 W 4
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 2

```
0 6 6
```

- The route in the Sample 1 is not valid. After the first edge you will be in room 2 with 4 star, since $4 < \ell_2 = 5$ stars, you can not leave room 2.
- You can walk through edges 5, 4, 3 to earn 7 stars and end in room 2.
- You can walk through edges 5, 4 to earn 7 stars and end in room 3.

Sample Input 3

```
3 5
0 5 5
1 2 B 4
2 3 W 1
3 2 W 0
3 3 B 2
1 3 W 4
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 3

```
0 4 4
```

Sample Input 4

```
4 4
4 0 9 0
3 2 W 5
1 2 W 7
4 1 B 4
4 3 B 7
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 4

```
4 11 7 0
```

Sample Input 5

```
5 9
0 0 0 0 0
1 2 B 7
1 3 W 9
1 4 B 9
2 3 B 4
2 3 W 1
2 4 W 3
3 4 W 5
3 5 B 5
4 5 W 6
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 5

```
0 7 9 10 15
```

Sample Input 6

```
2 2
0 0
1 2 B 1234
2 1 B 9876
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 6

```
9876 1234
```

Sample Input 7

```
1 2
1000000000
1 1 B 10000
1 1 B 9999
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 7

```
0
```

There might be the case that you can not enter any room from the start.

Problem F. Counting Paths on a Graph

Problem Description

Given integer n and a set of m edges $\{(u_i, v_i)\}_{i=1}^{m}$, construct the undirected simple graph G as below:

- Nodes are numbered $1, 2, \ldots, n$.
- Two node a and b has an edge if
  1. there exist $i \in [1, m]$ such that $(a, b) = (u_i, v_i)$, or
  2. $|a - b| = \gcd(a, b)$.

A route passing through node $(r_1, r_2, \ldots, r_k)$ is called *excited* if the difference between adjacent node is strictly increasing, that is,

- $|r_{i+1} - r_i| > |r_i - r_{i-1}|$ for $i = 2, 3, \ldots, k - 1$.

Please find the number of excited routes in G with length > 1, modulo p.

Input Format

The first line contains three integers n, m, and p.

Each of the next m lines contains two integers u_i and v_i.

Output Format

Print a single non-negative integer, the number of excited routes in G with length > 1, modulo p.

Constraints

- $1 \leq n \leq 3\,000\,000$.
- $0 \leq m \leq 500\,000$.
- $10^8 \leq p \leq 10^9$.
- $1 \leq u_i < v_i \leq n$ for $i = 1, 2, \ldots, m$.
- $(u_i, v_i) \neq (u_j, v_j)$ for $1 \leq i < j \leq m$.
- $|u_i - v_i| \neq \gcd(u_i, v_i)$ for $i = 1, 2, \ldots, m$.
- All inputs are integers.

Subtasks

1. (10 points) $n \le 10$; $m = 0$.
2. (10 points) $n \le 10$.
3. (20 points) $n \le 80$.
4. (20 points) $n \le 400$.
5. (20 points) $n \le 3000$.
6. (10 points) $n \le 100\,000$.
7. (10 points) No additional constraints.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1-5	4000	262144
1	1-14	4000	262144
2	1-22	4000	262144
3	1-28	4000	262144
4	1-34	4000	262144
5	1-41	4000	262144
6	1-47	4000	262144
7	1-53	4000	262144

Samples

Sample Input 1

```
5 0 1000000000
```

This sample input satisfies the constraints of all the subtasks.

Sample Output 1

```
14
```

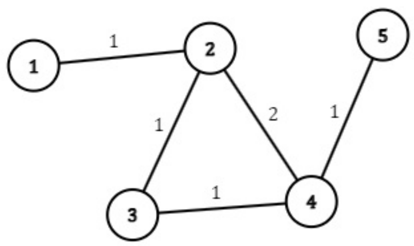


Figure 1: The graph in Sample Input 1.
Number on the edges show the difference between the two nodes they connect.

There are 10 excited routes with length 2:

- $(1,2),(2,1),(2,3),(2,4),(3,2),(3,4),(4,2),(4,3),(4,5),(5,4)$

There are 4 excited routes with length 3:

- $(1,2,4),(3,2,4),(3,4,2),(5,4,2)$

Sample Input 2

```
5 3 1000000000
1 4
2 5
1 5
```

This sample input satisfies the constraints of Subtasks 2, 3, 4, 5, 6, 7.

Sample Output 2

```
46
```

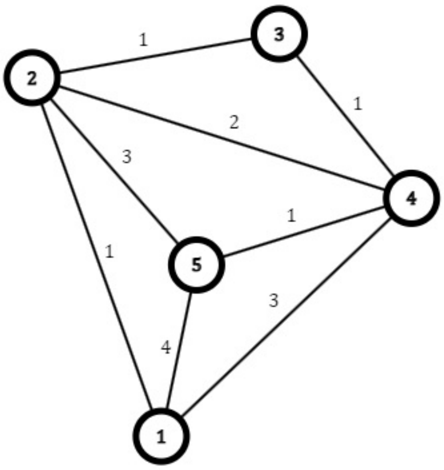


Figure 1: The graph in Sample Input 2.

For example, $(1, 2, 5)$ is an excited route of length 3, and $(1, 2, 4, 1, 5)$ is an excited route of length 5.

Sample Input 3

```
6 7 676767676
1 3
1 4
1 5
1 6
2 5
2 6
3 5
```

This sample input satisfies the constraints of Subtasks 2, 3, 4, 5, 6, 7.

Sample Output 3

```
170
```

The graph in Sample Input 3 is K_6, a complete graph.

Sample Input 4

```
2779 0 360565573
```

This sample input satisfies the constraints of Subtasks 5, 6, 7.

Sample Output 4

```
0
```

Make sure you output the answer modulo p. 🙂

Sample Input 5

```
100000 10 998244353
14608 28155
50059 70059
78848 96576
14982 72820
72663 82680
24809 50767
5371 69650
71737 86095
35624 82984
16759 20037
```

This sample input satisfies the constraints of Subtasks 6, 7.

Sample Output 5

```
964178665
```

Problem G. Rod Cutting

Problem Description

There are a rod of length $3n$ meters, you want to cut the rod and pack it into n bags of rods which lengths sum up to 3 meters each.

A cut at i^{th} meter mark costs c_i dollars, and a cut on a rod of ℓ meters costs $C \cdot \ell$ dollars.

Please find out the minimum cost to pack the rod into n bags of rods.

Input Format

- line 1: $n \;\; C$
- line 2: $c_1 \;\; c_2 \;\; \ldots \;\; c_{3n-1}$

Output Format

- line 1: ans

Constraints

- $1 \leq n \leq 800$.
- $0 \leq C \leq 10^6$.
- $0 \leq c_i \leq 10^6$ for $i = 1, 2, \ldots, 3n-1$.
- All input values are integers.

Subtasks

1. (20 points) $n \leq 6$.
2. (20 points) $n \leq 15$.
3. (15 points) $n \leq 30$.
4. (15 points) $n \leq 50$.
5. (10 points) $c_i = 0$ for $i = 1, 2, \ldots, 3n-1$.
6. (10 points) $C = 0$.
7. (5 points) $n \leq 300$.
8. (5 points) No additional constraints.

No.	Testdata Range	Time Limit (ms)	Memory Limit (KiB)
Samples	1-6	2000	262144
1	1-42	2000	262144
2	1-48	2000	262144
3	1-54	2000	262144
4	1-60	2000	262144
5	61-66	2000	262144
6	67-74	2000	262144
7	1-60,75-80	2000	262144
8	1-92	2000	262144

Samples

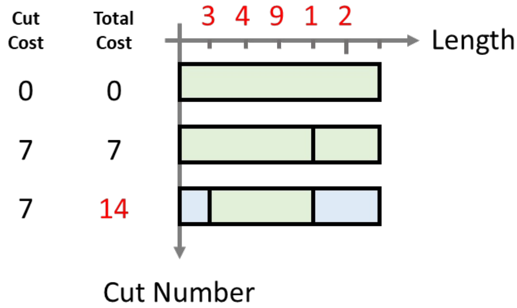
Sample Input 1

```
2 1
3 4 9 1 2
```

This sample input satisfies the constraints of Subtasks 1, 2, 3, 4, 7, 8.

Sample Output 1

```
14
```



Sample Input 1

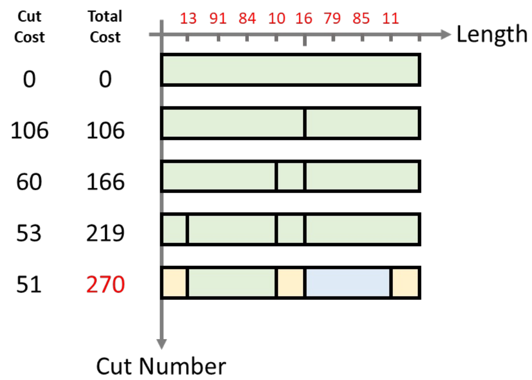
Sample Input 2

```
3 10
13 91 84 10 16 79 85 11
```

This sample input satisfies the constraints of Subtasks 1, 2, 3, 4, 7, 8.

Sample Output 2

```
270
```



Sample Input 2

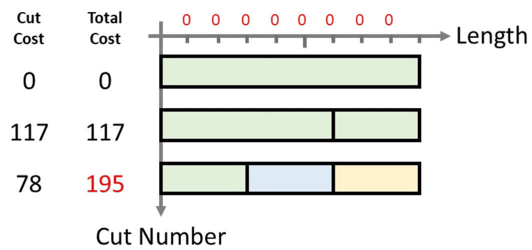
Sample Input 3

```
3 13
0 0 0 0 0 0 0 0
```

This sample input satisfies the constraints of Subtasks 1, 2, 3, 4, 5, 7, 8.

Sample Output 3

```
195
```



Sample Input 3

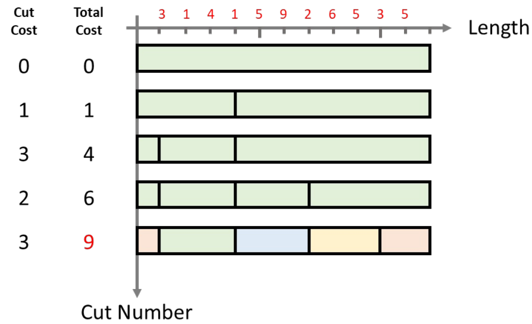
Sample Input 4

```
4 0
3 1 4 1 5 9 2 6 5 3 5
```

This sample input satisfies the constraints of Subtasks 1, 2, 3, 4, 6, 7, 8.

Sample Output 4

```
9
```



Sample Input 4

Sample Input 5

```
6 12
17 79 30 85 87 10 15 62 81 99 13 71 18 23 85 11 35
```

This sample input satisfies the constraints of Subtasks 1, 2, 3, 4, 7, 8.

Sample Output 5

```
785
```

Sample Input 6

```
2 1000000
1000000 1000000 1000000 1000000 1000000
```

This sample input satisfies the constraints of Subtasks 1, 2, 3, 4, 7, 8.

Sample Output 6

```
7000000
```