
**SHOPEE CODE LEAGUE 2021**

## Shopee Programming Contest

<u>Competition Details</u>

The competition starts **from 1:00pm (GMT+7) / 2:00pm (GMT+8) to 4:00pm (GMT+7) / 5:00pm (GMT+8).** The **countdown will begin at 1pm (GMT+7) / 2PM (GMT+8) and you will have 3 hours or up to 4:00pm (GMT+7) / 5:00pm (GMT+8) (whichever earlier) to submit your codes.**

**Duration**: 3 Hours

# Shopee Tanam

## Problem Statement

In Shopee, there is a game called Shopee Tanam. It's a game developed by Shopee game developers in Indonesia. Shopee Tanam means that you can plant some plants on the game platform. One of the game developers has an idea to create a new sub game in Shopee Tanam. In the sub game, you are given a park and the park has 1 dimensional shape and it consists of **M** different cells. In each cell you must plant exactly 1 tree. Each day, the tree can yield a beneficial fruit / poisonous fruit / neither beneficial nor poisonous fruit. The beneficial fruit means that you will get a positive number of health.The poisonous fruit gives you a negative number of health. Otherwise, you will get 0 health.

Then you are given a character that will start at the left side of the one dimensional park. You are given **N** days to play the game. Each day, the number of health in each fruit produced by a tree might change. In one day the character can do one of these actions:

1. Not crossing the park at all (stay at the current spot).
2. Walk through the park going through up to **M** cells, gathering all the beneficial and poisonous fruits along the way, then the character turns around and goes back to the initial position before he walks through the park.
3. Crossing the park completely, and going to the opposite side of the park, then the character rests there.

Note: On one day, the fruit in a cell can be gotten **at most once.**

Help the character to get the maximum amount of health.

## Input

There will be **T (1 <= T <= 10)** number of test cases. On each test case, there will be a line with 2 integers, **N** and **M** (1 <= **N** <= 1000, 1 <= **M** <= 1000). Then **N** lines follow up, with each line containing **M** integers, $\mathbf{A_{i,j}}$ ($-10^9$ <= $\mathbf{A_{i,j}}$ <= $10^9$), which means the health of the fruit on day **i** at cell **j**.

## Output

The output must consist of **T** number of integers, indicating the maximum amount of health that the character can get.

**Sample Input**

3
1 5
-9 -8 1 2 3
2 3
1 4 -5
-1 -9 100
2 3
1 4 -5
-1 -1 100

**Sample Output**

0
100
103

**Explanation**

1. For the first example, the character can decide to not cross the park at all, hence the total health that the character gets is 0.
2. For the second example,
    a. On the first day, the character goes from the left, then completely crossing the park, then he rests at the right side of the park, and the total number of health is: 1 + 4 + (-5) = 0.
    b. On the second day, the character goes from the right side of the bridge, only to the first cell from the right, then goes back to the right side of the bridge, and the total number of health the character gets from the second day is 100.
    c. Hence, the total health that the character gets from 2 days is 0 + 100 = 100.
3. For the third example,
    a. On the first day, the character goes from the left, but only getting the health from the first 2 cells, then the character goes back to the left side of the park, and the total number of health is: 1 + 4 = 5.
    b. On the second day, the character goes from the left side of the park, and goes completely crossing the park, gaining total health of: (-1) + (-1) + 100 = 98 for the second day.
    c. Hence, the total health that the character gets from 2 days is 5 + 98 = 103.

# Shoffee

**Problem Statement**

It has been a year since Noel joined Shopee. Like every ordinary day, before starting daily work, Noel will go to the pantry and make a cup of coffee for himself.

A box of length $N$ is placed next to the coffee machine in the pantry, and coffee beans of different flavors are placed in a row. Noel has his own taste preference value $V_i$ for each type of coffee bean. Noel has a habit of his own, that is, every time he will select coffee beans in **consecutive** boxes (assuming that each flavor of coffee beans is unlimited supply) and put them into the coffee machine to get a cup of mixed coffee whose taste preference value $\overline{V}$will be the average value of the chosen flavors.

A cup of mixed coffee will be called Shoffee if its taste preference value $\overline{V}$is not less than $K$, Shoffee can quickly wake Noel up.

Noel hopes that every day he can drink a cup of Shoffee and keep himself in a good working status. Please help him calculate how many types of Shoffee can be in total.

**Input Format**

Each test case will consist of exactly $2$ lines.

The first line are two positive integers $N$ $(1 \leq N \leq 10^5)$ and $K$ $(1 \leq N \leq 10^4)$ splitted by space, representing the number of coffee bean flavors, and Noel's expectation for the coffee.

The second line contains $N$ positive integers $V_i$ $(1 \leq V_i \leq 10^4)$ splitted by space, representing Noel's preference value for each type of coffee bean.

**Output Format**

For each test case, please output an answer in one line representing the number of Shoffee can be in total.

**Sample Input**

Case 1:

```
3 3
1 3 4
```

Case 2:

```
6 3
1 1 4 5 1 4
```

**Sample Output**

Case 1:

```
3
```

Case 2:

```
10
```

**Explanation**

For the first test case, there are totally $N = 6$ different consecutive sequences: $(1)$, $(3)$, $(4)$, $(1, 3)$, $(3, 4)$, $(1, 3, 4)$, and their average values are $\overline{V} = 1,\ 3,\ 4,\ 2,\ 3.5,\ 8/3$.

Among these, there will be $3$ numbers greater than or equal to $K = 3$, so the answer will be $3$.

# Order Delivery

**Problem Statement**

In the parallel universe, where there are 13 months, Shopee has a 13.13 campaign. During this 13.13 campaign, Shopee gives free shipping delivery vouchers to all users who buy item X. Shopee has N warehouses to store the item X, and each warehouse has Wi number of item X. Each warehouse is located in a city and all cities have at most one warehouse. To serve the customers, each warehouse has their own courier delivery. The cost of the delivery in warehouse i is Ci dollar per kilometer. Interestingly, in this parallel universe, the distance between neighboring cities is exactly one kilometer. The cities can be represented as a graph, where a node represents the city and the edge represents the road between cities and all the cities are connected. Warehouse i is located at city Pi.

During the 13.13 campaign, people are very excited to buy this item X because of the free shipping discounts. As a result, there are M orders created, each order contains Ki number of item X, and it needs to be delivered to city Gi. To serve all the customers, multiple warehouses can be used to serve a single order. So, one order can be served by multiple warehouses.

Because of the free shipping discounts, Shopee needs to pay the delivery fee of all the orders. Your task is to help Shopee to minimize the delivery fee in this 13.13 campaign.

**Input Format**

The first line contains three integers N, D, and E (1 <= N <= 20, 1 <= D <= N, N-1 <= E <= 200) representing the number of cities, warehouses, and roads in this parallel universe. The next E lines contain 2 integers Xi and Yi (1 <= Xi,Yi <= N, Xi != Yi) which indicates that there is a road between city Xi and Yi. The next D line contains 3 integers Wi, Ci, and Pi (1 <= Wi <= 10^9, 1 <= Ci <= 10^6, 1 <= Pi <= N) which represents the number of item X in warehouse i and the delivery fee of warehouse i per kilometer and the location of warehouse i. The next line contains an integer M (1 <= M <= 100000) which represents the number of orders. Each of the next M lines contain two integers Ki and Gi (1 <= Ki <= 10^9, 1 <= Gi <= N, sum of all Ki <= 10^9) which represent the number of item X ordered in order-i and the city of order i.

**Output Format**

Output a single integer contains the total delivery cost of all orders. It is guaranteed that Shopee can serve all the orders.

**Sample Input 1**

```
8 3 11
1 2
1 3
2 3
3 4
4 5
5 6
5 7
5 8
4 6
3 7
7 8
12 5 1
11 10 6
1 6 7
3
3 4
4 4
7 5
```

**Sample Output 1**

```
136
```

# Divider

**Problem Statement**

Shopee has N software engineers. Shopee accommodates them by arranging N tables on a 1D plane. However, many people raise concerns that the work environment is too noisy. To mitigate this issue, Shopee decides to group the engineers into K groups. A group is a non-overlapping segment of contiguous engineers. Shopee will then put dividers between the groups.

The noise value of a group is defined by the following function:
noise(l, r) = sum(A[l], A[l + 1], ..., A[r]) * (r - l + 1)
Where:
- noise(l, r) represents the noise value of a group consisting of the l-th engineer up to the r-th engineer.
- A[i] represents the noise factor of the i-th engineer.

Shopee wants to minimize the total noise value. Please help Shopee to find the minimum total noise value possible.

**Input**

The first line of input contains 2 integers: N K (1 <= N <= 10,000; 1 <= K <= min(N, 100)) representing the number of engineers and the number of groups respectively. The next line contains N integers: A[i] (1 <= A[i] <= 10,000) representing the noise factor of the i-th engineer.

**Output**

Output in a line an integer representing the minimum total noise value possible.

**Sample input #1**

4 2
1 3 2 4

**Sample output #1**

20

**Explanation**
1 | 3 2 4
(1 * 1) + (9 * 3) = 28
1 3 | 2 4
(4 * 2) + (6 * 2) = 20
1 3 2 | 4
(6 * 3) + (4 * 1) = 22

We can see that from all the possibilities, 20 is the minimum total noise value.

# Shopee Farm

**Problem Statement**

Shopee Farm is a popular game in the Shopee app. In this game, you are living on a planet which looks like a giant soccer ball. Similar to a soccer ball, there are 12 pentagon faces and 20 hexagon faces which form a "Truncated Icosahedron" as follows:

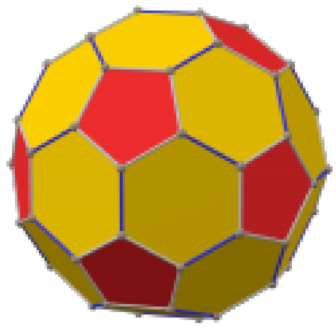

Figure 1: The giant soccer ball

On each pentagon face, there is a tree. Each tree will produce some Shopee coins after each day. After seven days, if the coins on a face are not collected, the tree on that face will stop producing coins (until the coins are collected). Due to the climate difference between pentagon faces and hexagon faces, trees are not only available on hexagon faces.

Let's "flatten" the soccer ball and number the pentagon faces as follows:

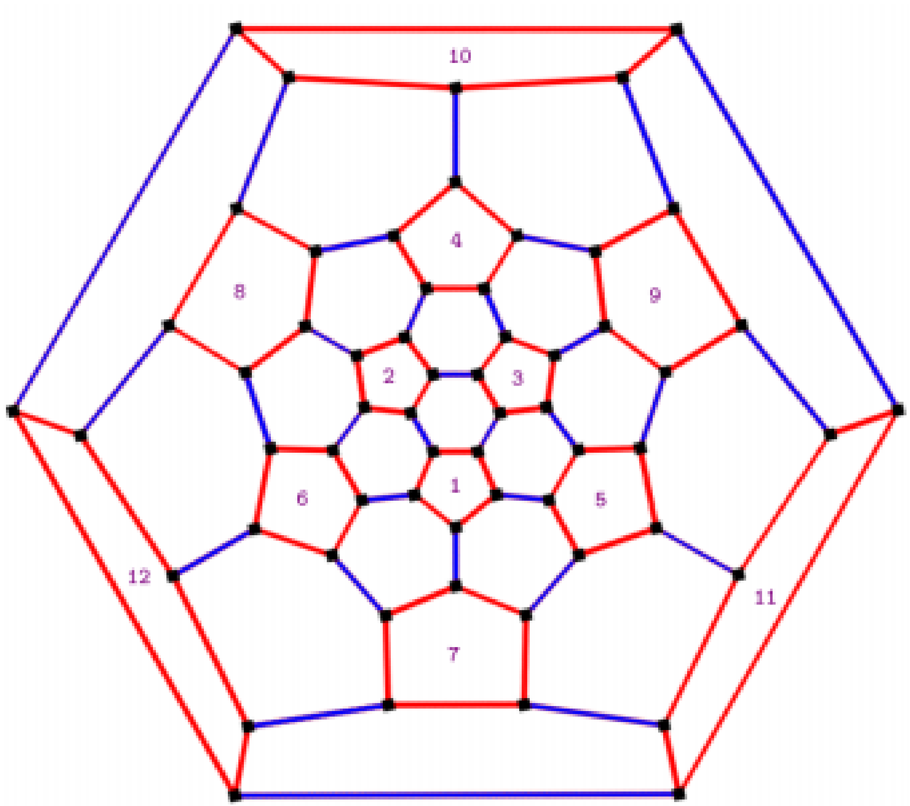


Figure 2: The giant soccer ball flattened

There are $D$ days in total. On the first day (day 0), you are standing on the hexagon face surrounded by face 1, 2, and 3. Each day, you have to move to an *adjacent* face, collect all the Shopee coins on that face (if available). The number of coins you can collect is determined by this rule:

- If you are standing on a hexagon face, there are no Shopee coins to collect.
- Otherwise, let $K$ be the id of the pentagon face that you are standing on,

  – If this is the first time you visit this face, the number of coins is $C[K, 7]$

  – Otherwise, let $P$ be the index of the most recent day that you visited this face, let $Q$ be the index of the current day, the number of coins is $C[K, min(Q - P, 7)]$.

$C$ is a 12-by-7 matrix of integers. It is guaranteed that $C[K, i] \le C[K, j]$ for all $i < j$. Find the maximum number of Shopee coins that you can collect at the end of day $D - 1$.

## Input

The first line consists of an integer $D$, the number of days ($1 \le D \le 10^3$).

In the next 12 lines, line $K$ ($1 \le K \le 12$) consists of 7 integers $C[K, i]$ ($1 \le i \le 7$, $0 \le C[K, i] \le 10^5$).

## Output

Output the maximum number of Shopee coins that you can collect.

| Sample Input | Sample Output |
|---|---|
| 1<br>100 200 300 400 500 600 700<br>101 201 301 401 501 601 701<br>102 202 302 402 502 602 702<br>103 203 303 403 503 603 703<br>104 204 304 404 504 604 704<br>105 205 305 405 505 605 705<br>106 206 306 406 506 606 706<br>107 207 307 407 507 607 707<br>108 208 308 408 508 608 708<br>109 209 309 409 509 609 709<br>110 210 310 410 510 610 710<br>111 211 311 411 511 611 711 | 702 |

## Explanation

On day 0, move to face 3, collect 702 durians.

## Note

To make sure that you have read the problem statement correctly, here are some facts that you can use to verify your understanding:

- The distance between face 7 and face 10 is 3.
- The distance between face 1 to face 10 is 5.
- There are 12 pentagon faces and 20 hexagon faces.