Problem C. Kickstart Alarm

Confused? Read the quick-start guide.

Small input 13 points	Solve C-small You may try multiple times, with penalties for wrong submissions.
Large input 26 points	You must solve the small input first. You have 8 minutes to solve 1 input file. (Judged after contest.)

Problem

Shil has a very hard time waking up in the morning each day, so he decides to buy a powerful alarm clock to Kickstart his day. This Alarm is called a Kickstart Alarm. It comes pre-configured with **K** powerful wakeup calls. Before going to bed, the user programs the clock with a Parameter Array consisting of the values $\mathbf{A_1}$, $\mathbf{A_2}$, ..., $\mathbf{A_N}$. In the morning, the clock will ring **K** times, with the i-th wakeup call having power POWER_i.

To calculate POWER_i, the alarm generates all the contiguous subarrays of the Parameter Array and calculates the summation of the i-th exponential-power of all contiguous subarrays. The i-th exponential-power of subarray $\mathbf{A_j}$, $\mathbf{A_{j+1}}$, ..., $\mathbf{A_k}$ is defined as $\mathbf{A_j} \times 1^i + \mathbf{A_{j+1}} \times 2^i + \mathbf{A_{j+2}} \times 3^i + \ldots + \mathbf{A_k} \times (k-j+1)^i$. So POWER_i is just the summation of the i-th exponential-power of all the contiguous subarrays of the Parameter Array.

For example, if i = 2, and **A** = [1, 4, 2], then the i-th exponential-power of **A** would be calculated as follows:

- 2-nd exponential-power of [1] = $1 \times 1^2 = 1$
- 2-nd exponential-power of [4] = $4 \times 1^2 = 4$
- 2-nd exponential-power of [2] = $2 \times 1^2 = 2$
- 2-nd exponential-power of [1, 4] = $1 \times 1^2 + 4 \times 2^2 = 17$
- 2-nd exponential-power of [4, 2] = $4 \times 1^2 + 2 \times 2^2 = 12$
- 2-nd exponential-power of [1, 4, 2] = $1 \times 1^2 + 4 \times 2^2 + 2 \times 3^2 = 35$

so the total is 71.

Tonight, Shil is using his Kickstart Alarm for the first time. Therefore, he is quite worried about the sound the alarm might make in the morning. It may wake up the neighbors, or, worse yet, it may wake up the whole planet! However, calculating the power of each wakeup call is quite difficult for him. Given **K** and the Parameter Array $\mathbf{A_1}$, $\mathbf{A_2}$, ..., $\mathbf{A_N}$, can you help him by calculating the summation of power of each wakeup call: $\text{POWER}_1 + \text{POWER}_2 + \ldots + \text{POWER}_K$?

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case consists of one line with nine integers **N**, **K**, $\mathbf{x_1}$, $\mathbf{y_1}$, **C**, **D**, $\mathbf{E_1}$, $\mathbf{E_2}$ and **F**. **N** is the length of array **A**, **K** is the number of wakeup calls. Rest of the values are parameters that you should use to generate the elements of the array **A**, as follows.

Use the recurrences below to generate x_i and y_i for i = 2 to **N**:

- $x_i = (\mathbf{C} \times x_{i-1} + \mathbf{D} \times y_{i-1} + \mathbf{E_1})$ modulo **F**.
- $y_i = (\mathbf{D} \times x_{i-1} + \mathbf{C} \times y_{i-1} + \mathbf{E_2})$ modulo **F**.

We define $\mathbf{A_i} = (x_i + y_i)$ modulo **F**, for all i = 1 to **N**.

Output

For each test case, output one line containing `Case #x: POWER`, where `x` is the test case number (starting from 1) and `POWER` is the summation of POWER_i, for i = 1 to **K**. Since `POWER` could be huge, print it modulo 1000000007 ($10^9 + 7$).

Limits

$1 \le \mathbf{T} \le 100$.
$1 \le \mathbf{x_1} \le 10^5$.
$1 \le \mathbf{y_1} \le 10^5$
$1 \le \mathbf{C} \le 10^5$.
$1 \le \mathbf{D} \le 10^5$.
$1 \le \mathbf{E_1} \le 10^5$.
$1 \le \mathbf{E_2} \le 10^5$.
$1 \le \mathbf{F} \le 10^5$.

Small dataset

$1 \le \mathbf{N} \le 100$.
$1 \le \mathbf{K} \le 20$.

Large dataset

$1 \le \mathbf{N} \le 10^6$.
$1 \le \mathbf{K} \le 10^4$.

Sample

Input

```
2
2 3 1 2 1 2 1 1 9
10 10 10001 10002 10003 10004 10005 10006 89273
```

Output

```
Case #1: 52
Case #2: 739786670
```

In Sample Case #1, the Parameter Array is [3, 2]. All the contiguous subarrays are [3], [2], [3, 2].

For i = 1:

- 1-st Exponential-power of [3] = $3 \times 1^1 = 3$
- 1-st Exponential-power of [2] = $2 \times 1^1 = 2$
- 1-st Exponential-power of [3, 2] = $3 + 2 \times 2^1 = 7$

So $POWER_1$ is 12.

For i = 2:

- 2-nd Exponential-power of [3] = $3 \times 1^2 = 3$
- 2-nd Exponential-power of [2] = $2 \times 1^2 = 2$
- 2-nd Exponential-power of [3, 2] = $3 + 2 \times 2^2 = 11$

So $POWER_2$ is 16.

For i = 3:

- 3-rd Exponential-power of [3] = $3 \times 1^3 = 3$
- 3-rd Exponential-power of [2] = $2 \times 1^3 = 2$
- 3-rd Exponential-power of [3, 2] = $3 + 2 \times 2^3 = 19$

So $POWER_3$ is 24.