

Kickstart Round C 2018

A. Planet Distance

B. Fairies and Witches

C. Kickstart Alarm

Ask a question

View my submissions

Submissions

Planet Distance

10nt	Not attempted
TOPL	Not attempted

239/386 users correct (62%)

15pt Not attempted 235 users attempted

Fairies and Witches

15pt	Not attempted		
	10/16 users correct		
	(63%)		

21pt Not attempted 8 users attempted

Kickstart Alarm

13pt	Not attempted	
	23/29 users correct	
	(79%)	

26pt Not attempted 10 users attempted

- Top Scores

nuip	100
alex20030190	74
rkm0959	64
rapel	64
thundercracker	64
teomrn	64
phirasit	64
Nyan101	64
OnionPringles	64
nhho	61

Problem C. Kickstart Alarm

Confused? Read the quick-start guide.

Small input
13 points
Solve C-small
You may try mu

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 A_1 , A_2 , ..., 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 A_j , A_{j+1} , ..., A_k is defined as $A_j \times 1^j + A_{j+1} \times 2^j + A_{j+1}$

 $\mathbf{A_{j+1}} \times 2^{j} + \mathbf{A_{j+2}} \times 3^{j} + ... + \mathbf{A_{k}} \times (k-j+1)^{j}$. 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 $\mathbf{A} = [1, 4, 2]$, then the i-th exponential-power of \mathbf{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 A_1 , A_2 , ..., A_N , can you help him by calculating the summation of power of each wakeup call: POWER₁ + POWER₂ + ... + 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, x_1 , y_1 , C, D, E_1 , 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:

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

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

 $1 \le N \le 100$.

1 ≤ **K** ≤ 20.

Large dataset

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

Sample

Input

```
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 POWER₁ 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 POWER₂ 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 POWER₃ is 24.

All problem statements, input data and contest analyses are licensed under the <u>Creative Commons Attribution License</u>.

© 2008-2018 Google Google Home - Terms and Conditions - Privacy Policies and Principles

Powered by



Google Cloud Platform