Another Update-Query Problem

Limits: 1s, 512 MB

You will be given an array **A** of length **N**. On that array you will have to do **Q**operations. Operations are of two types.

Operation-1: U(l, r, x): Add x to A_i, where i is in range [l, r]

Operation-2: Q(l, r, d): Print the output of this series modulo 1000000007:

$1A_l + (1+d)A_{l+1} + (1+2d)A_{l+2} + (1+3d)A_{l+3} + \ldots + (1+(r-l)d)A_r$

Input

Input starts with an integer **T (1≤T≤10)**, denoting the number of test cases.
First line of each case has two integers, **N (1≤N≤100000)** and **Q (1≤Q≤100000)**.
Second line has N integers **A_i (1≤A_i≤1000000000)**, indicating the values of the array.
Following **Q** lines each has four integers. First of those four integers are **C (1≤C≤2)**.
If **C=1**, then it requests an operation of type-1. The other three integers will be **L (1≤L≤N)**, **R (1≤R≤N, L≤R)**, **X (1≤X≤1000000000)**. And you will have to do operation **U(L, R, X)**.
If **C=2**, then it requests an operation of type-2. The other three integers will be **L (1≤L≤N)**, **R (1≤R≤N, L≤R)**, **D (1≤D≤1000000000)**. And you will have to print the value of **Q(L, R, D)** modulo **1000000007**.

Easy Subtask:
1≤N≤2000
1≤Q≤2000
C=2 for all cases. That means there is no request for operation-1.

Medium Subtask:
1≤N≤100000
1≤Q≤100000
C=2 for all cases. That means there is no request for operation-1.

Hard Subtask:
Full specification

Output

For each case, first print the case number, starting from 1, in a separate line. For each request of operation-2, print the result in a separate line.

Samples

Input	Output
2 5 4 2 8 19 9 1 2 2 5 3 2 2 4 6 2 2 5 3 2 2 5 8 3 3 6 7 1 2 2 2 7 1 1 3 8 2 1 2 6	Case 1: 157 258 157 357 Case 2: 7 119

Notes