
Expected Waiting Time

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 512 megabytes

DreamGrid is a famous coach of competitive programming. He is so kind that many competitors are willing to ask him for advice.

DreamGrid meets exactly n competitors every day in his office, and thus n “arriving” events and n “selecting” events will happen. An arriving event at time t indicates that a competitor arrives at the waiting room of the office at time t , and a selecting event at time t indicates that DreamGrid randomly selects (with equal probability) a competitor from the waiting room to talk with. Of course, if the selecting event happens, the waiting room must not be empty. After the talk, the competitor leaves the office and never comes back.

After several days, DreamGrid starts to be curious about the average of the total expected waiting time of every competitor in all valid cases. The waiting time of a competitor is the time he is selected by DreamGrid minus the time he arrives at the waiting room. A case is a sequence of length $2n$ consisting of n arriving events and n selecting events, where the i -th event will happen at time a_i . For a valid case, it must be satisfied that when a selecting event happens, the waiting room must not be empty.

For example, let's denote an arriving event as ‘A’, and a selecting event as ‘S’. If $n = 2$, $a_1 = 1$, $a_2 = 2$, $a_3 = 3$ and $a_4 = 4$, then the sequence “AASS” is valid, but the sequence “ASSA” is not valid, as the “selecting” event happening at time $a_3 = 3$ is not valid.

As the answer may not be an integer, you are supposed to calculate $ab^{-1} \bmod p$, where $\frac{a}{b}$ (a and b are coprime) is the answer, $p > 2n$ and p is prime, and b^{-1} is the modular multiplicative inverse of b with respect to the modulus p . It's easy to prove that the prime factors of b will never be larger than $2n$.

Input

There are multiple test cases. The first line of input contains an integer T ($1 \leq T \leq 10^4$), indicating the number of test cases. For each test case:

The first line contains five integers n, p, b_0, A, B ($1 \leq n \leq 10^6$, $2n < p \leq 2 \times 10^9$, $0 \leq b_0, A, B < p$), where p is a prime number. The meanings of n and p are described above. The rest of them is a generator of a , where $a_0 = 0$, $a_i = a_{i-1} + b_i + 1$ and $b_i = (A \cdot b_{i-1} + B) \bmod p$ for all $1 \leq i \leq 2n$.

It is guaranteed that the sum of n in all test cases does not exceed 10^7 .

Output

For each test case, output one integer denoting the answer in a single line.

Example

standard input	standard output
5	1
1 1000000007 0 1 0	12
2 1000000007 0 1 1	1
2 7 5 2 3	21
3 31 15 6 24	879705565
20 1000000007 0 1 0	

Note

Let's denote the arriving event as ‘A’, and the selecting event as ‘S’.

In the first test case, we have $a_1 = 1$ and $a_2 = 2$, and there is only one valid sequence “AS”. The waiting

time of the only competitor is $2 - 1 = 1$, so the answer is 1, and we need to output $1 \bmod 1000000007 = 1$.

In the second test case, we have $a_1 = 2$, $a_2 = 5$, $a_3 = 9$ and $a_4 = 14$. There are two valid sequences “ASAS” and “AASS”.

For the first sequence, the expected waiting time of the first arriving competitor is $5 - 2 = 3$, and the expected waiting time of the second arriving competitor is $14 - 9 = 5$.

For the second sequence, the expected waiting time of the first arriving competitor is $((9 - 2) + (14 - 2))/2 = 9.5$, and the expected waiting time of the second arriving competitor is $((9 - 5) + (14 - 5))/2 = 6.5$. So the answer is $((3 + 5) + (9.5 + 6.5))/2 = 12$, and we need to output $12 \bmod 1000000007 = 12$.

In the third test case, we have $a_1 = 7$, $a_2 = 9$, $a_3 = 15$ and $a_4 = 22$. Just like the analysis for the 2nd sample test case, the total expected waiting time for the sequence “ASAS” is $(9 - 7) + (22 - 15) = 9$, and the total expected waiting time for the sequence “AASS” is $((15 - 7) + (22 - 7))/2 + ((15 - 9) + (22 - 9))/2 = 21$. So the answer is $(9 + 21)/2 = 15$, and we need to output $15 \bmod 7 = 1$.