9_費馬最後定理 (Fermat_Last_Theorem)

(1分 /2分 /4分 /13分)

時間限制: 5 seconds
記憶體限制: 512 MB

問題敘述

古希臘的阿基米德得知現代德國的高斯能夠快速求解 $1 \sim N$ 的總和，深感不服。

於是，阿基米德將問題升級，改為求解 $\sum_{i=1}^{N} \lfloor \frac{i}{N} \rfloor$ 的和。為此，他需要先證明對於任何有理數，都存在一個比它小的整數。透過這個定理，他證明了著名的阿基米德定理。

然而，不久之後，阿基米德發現答案永遠是一個定值。為增加難度，他改變了分數的上下位置，要求求解 $\sum_{i=1}^{N} \lfloor \frac{N}{i} \rfloor$。

當他剛把這個式子寫下時，牛頓路過，並利用 $\lfloor \frac{N}{i} \rfloor$ 的取值以及富比尼定理秒殺了這個問題。之後，他利用這個結果證明了 $\int_{i=1}^{\infty} \frac{1}{x} dx$ 發散，最終發明了微積分。

不願落後的阿基米德再次提高難度，變成了給定 Q 筆詢問，每次給定 L 和 R，要求 $\sum_{i=L}^{R} \lfloor \frac{N}{i} \rfloor$。但英國的凱萊認為這仍然不夠美感，大筆一揮，將問題改為給定 N 個矩陣 M_i，每次詢問要求計算 $\prod_{i=L}^{R} M_i$ 的結果。

明天要去決鬥伽羅瓦聽聞了這個問題，但他覺得在整數下的運算太無聊了，因此將加法和乘法改為在 $GF(2)$ 下的運算，也就是 $a + b$ 變成 $(a + b) \bmod 2$，$a \times b$ 變成 $(a \times b) \bmod 2$。

題目漸漸變得完整：給定 N 個矩陣 $M_0, M_1, \ldots, M_{N-1}$，接下來會有 Q 筆詢問，每次詢問 L 和 R，要求計算 $\prod_{i=L}^{R} M_i$ 的結果，並且其中的運算是在模 2 下進行的。

在大家忙著計算的時候，身為業餘數學家的費馬負責將算出來的答案記下來。但因為詢問數量實在太多，幾乎用光他筆記本中所有空白處。為此他要大家將答案全部 XOR 起來後再告訴他。

可是，最終筆記本剩餘的空間還是太小，因此儘管費馬已經對費馬最後定理找到了一個精妙的證明，卻依然沒有足夠的空位寫下。

而三百多年後，英國數學家懷爾斯才發現費馬需要的空頁超過一百頁，就是後話了。

輸入格式

輸入的第一行包含兩個正整數 N 和 k，代表矩陣的數量和每個矩陣的大小。

接下來的 N 個區塊中，每個區塊包含 k 行，描述了一個 $k \times k$ 的矩陣 M_i。

接下來一行包含一個正整數 Q，表示詢問的數量。

接下來兩行，分別包含三個正整數 A_1、B_1、x_0 和 A_2、B_2、y_0，表示生成詢問區間的公式。

對於 $i = 1, 2, \ldots, Q$，詢問的區間 $[L_i, R_i]$ 可以由以下方式生成：

$$x_i = (A_1 \times x_{i-1} + B_1) \bmod (10^9 + 7)$$
$$y_i = (A_2 \times y_{i-1} + B_2) \bmod (10^9 + 7)$$
$$L_i = \min(x_i \bmod N, y_i \bmod N)$$
$$R_i = \max(x_i \bmod N, y_i \bmod N)$$

輸出格式

令 $P_i = \prod_{j=L_i}^{R_i} M_j$，其中運算是在模 2 下進行的。

請輸出 $\bigoplus_{i=1}^{Q} P_i$，其中 $\oplus$ 是 XOR 運算。

資料範圍

- $1 \leq N \leq 3 \times 10^5$。
- $1 \leq k \leq 3$。
- $0 \leq M_i$ 裡的元素 ≤ 1（$0 \leq i \leq N-1$）。
- $1 \leq Q \leq 3 \times 10^7$。
- $1 \leq A_1, B_1, x_0, A_2, B_2, y_0 < 10^9 + 7$。

子任務

- 子任務 1 (1 point) $N, Q \leq 3000$。
- 子任務 2 (2 points) $N, Q \leq 3 \times 10^5$。
- 子任務 3 (4 points) $Q \leq 3 \times 10^6$。
- 子任務 4 (13 points) 無額外限制。

範例

輸入範例 1

```
3 3
000
000
000
111
111
111
100
000
000
3
1 1 2
1 1 2
```

輸出範例 1

```
011
111
111
```

輸入範例 2

```
3 2
10
00
01
01
10
00
3
604078599 524693017 85841332
671146413 219874579 658380675
```

輸出範例 2

```
10
00
```

範例說明

輸入範例 1 的矩陣如下：

$$M_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, M_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{1}$$

輸入範例 2 中的矩陣如下：

$$M_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, M_3 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \tag{2}$$

請注意，$\oplus$ 運算符號表示位元 XOR 運算。

對於兩個 $k \times k$ 的矩陣 U、V，若 $W = U \oplus V$，則對於 $1 \le i \le k$、$1 \le j \le k$：

$W[i][j] = (U[i][j] \oplus V[i][j])$