このゲームは Nim そのものであるため、次の問題に言い換えられます。

> $0 \leq A_i \leq K (1 \leq i \leq N)$、$A_1 \oplus A_2 \oplus \ldots \oplus A_N = 0$ を満たす数列 $A_1, A_2, \ldots, A_N$ は何通りあるでしょうか。答えは非常に大きくなる可能性があるため、10^9+7 で割った余りを答えてください。（$1 \leq N \leq 2000$、$0 \leq K \leq 10^{18}$）

ここで非負整数 x の i ビット目を $f(x,i)$ とすると、$A_1 \oplus A_2 \oplus \ldots \oplus A_N = 0$ という条件は、次のように言い換えられます。

$$\text{すべての正整数 } j \text{ について、} \sum_{i=1}^{N} f(A_i, j) \text{ は偶数}$$

つまり、A_i の 2 進数表記での各桁に注目し、1 が偶数個になるような数列 A が何通りあるかを求めればよいです。そこで次の DP を考えます。

$dp[i][j] :=$ (上位から i ビット目までを見たときに、K 未満と確定した要素の数が j 個のときの A の通り数)

$dp[i][j]$ が既に求まっているものとして遷移を考えます。ここで、K の 2 進数表記での桁数を D とおくことにします。

(1) $f(K, D-i) = 0$ のとき

$N-j$ 個の要素については、K を超えてはいけないので 0 を選ぶ必要があり、1 通りです。残りの j 個の要素については、0 と 1 のどちらを選択することも可能ですが、1 の個数は偶数でなければならないため 2^{j-1} 通りです（ただし、$j=0$ のときは 1 通りです）。ということで遷移は以下です。

$$\begin{cases} dp[i+1][j] \mathrel{+}= dp[i][j] \times 2^{j-1} & (j \geq 1) \\ dp[i+1][j] \mathrel{+}= dp[i][j] & (j = 0) \end{cases}$$

(2) $f(K, D-i) = 1$ のとき

j 個の要素については、$N-j$ が奇数の場合は奇数個の 1 を、偶数の場合は偶数個の 1 を選ぶことになり、どちらも 2^{j-1} 通りです。

$N-j$ 個の要素については、0 と 1 のどちらを選択することも可能です。0 を選択する個数を k としたとき、K 未満に確定する要素が k 個増えるため、遷移先は $dp[i+1][j+k]$ になります。$N-j$ 個から k 個の 0 を選ぶ方法は ${}_{N-j}C_k$ 通りあるため、最終的には次のような遷移になります。

$$dp[i+1][j+k] \mathrel{+}= dp[i][j] \times 2^{j-1} \times_{N-j} C_k$$

ここで一つ注意すべき点があり、$j=0$ かつ $N-k$ が奇数のときは、$\sum_{l=1}^{N} f(A_l, D-i)$ が必ず奇数となるため 0 通りです。

$$\begin{cases} dp[i+1][j+k] \mathrel{+}= 0 & (j = 0 \text{ かつ } N-k \text{ が奇数}) \\ dp[i+1][j+k] \mathrel{+}= dp[i][j] \times_{N-j} C_k & (j = 0 \text{ かつ } N-k \text{ が偶数}) \\ dp[i+1][j+k] \mathrel{+}= dp[i][j] \times_{N-j} C_k \times 2^{j-1} & (otherwise) \end{cases}$$

$dp[0][0] = 1$ を初期値として DP を行ったあと、$\sum_{i=0}^{N} dp[D][i]$ が答えになります。事前に $0 \leq n, r \leq N$ の範囲の 2^n と ${}_nC_r$ を計算しておくことで、各遷移の計算量が $O(1)$ となり、全体で $O(N^2 \log K)$ なので間に合います。