Day 5, SPbSU LOUD Enough Contest 2

February 4, 2023

Good and Lucky Matrices

- There is always the same number of good and lucky $n \times n$ matrices.
- Construct a good matrix row-by-row.
- Each new row should *not* be in some linear space.
- The total number of good matrices is $(2^n - 2^0) \cdot (2^n - 2^1) \cdot \ldots \cdot (2^n - 2^{n-1})$.
- A lucky matrix: at least one 1 in the first row, the corresponding column can be arbitrary. Repeat on the $(n-1) \times (n-1)$ matrix.
- The total number of lucky matrices is $(2^n - 1) \cdot 2^{n-1} \cdot (2^{n-1} - 1) \cdot 2^{n-2} \cdot \ldots \cdot (2-1) \cdot 2^0$.
- These numbers are the same.

Good and Lucky Matrices

- OK, but how to solve the problem?
- An “uglier” way : good matrices $\leftrightarrow$ sequences of binary blocks $\leftrightarrow$ lucky matrices.
- Requires implementing four conversions, but works.
- The intended solution converts good matrices to lucky matrices (and vice versa) directly, so you only need to implement two conversion procedures.

Good and Lucky Matrices

- Good $\to$ lucky: run Gauss, but don't swap the rows.
- Instead, for each row i, find the first yet-unused column j with $A_{i,j} = 1$.
- Now, "freeze" the following items into the answer: $A_{i,k}$ for each unused column k and $A_{r,j}$ for each $r \geqslant i$.
- Now, proceed with the usual step of the Gauss algorithm: ensure that the new values of $A_{r,j}$ are all zero when $r > i$. Notice that we already "froze" these matrix entries into the answer.
- The result is a lucky matrix, with $i \to j$ being exactly the greedy matching.

Good and Lucky Matrices

- Good → *lucky*: run Gauss, but don't swap the rows.
- Instead, for each row i, find the first yet-unused column j with $A_{i,j} = 1$.
- Now, "freeze" the following items into the answer: $A_{i,k}$ for each unused column k and $A_{r,j}$ for each $r \geqslant i$.
- Now, proceed with the usual step of the Gauss algorithm: ensure that the new values of $A_{r,j}$ are all zero when $r > i$. Notice that we already "froze" these matrix entries into the answer.
- The result is a good matrix, with $i \to j$ being exactly the greedy matching.

Good and Lucky Matrices

- Lucky $\to$ good: recover the greedy matching and then reverse the above process, starting from the lower rows and then going to the upper ones.
- The process was specifically made to be reversible: we remember $A_{r,j}$ before the row-xor operations, therefore we know whether or not we have actually done them during the Gauss algorithm.

Two Missing Numbers/Nimbers

- There was an unintentional hint: the problem was named "Two Missing Nimbers" in the testing system and in the contest standings.
- Interpret the elements of the input as elements of the field of size 2^{64}.
- This field has several interpretations, they are all isomorphic and have characteristic 2 (meaning that $x + x = 0$ for any x).
- Suppose that the target numbers are x and y.
- The sum of all input numbers is $x + y$ (all other numbers cancel out).
- We can also compute the sum of squares, but it is not useful: $(x + y)^2 = x^2 + 2xy + y^2 = x^2 + y^2$, because $2 = 0$.
- The sum of cubes is better: $(x + y)^3 - (x^3 + y^3) = 3(x^2 y + xy^2) = xy(x + y)$.
- Recover xy through division.

Two Missing Numbers/Nimbers

- Alternatively: compute $1/x + 1/y = (x + y)/(xy)$ (zeroes are improbable).
- Or compute $x^2y + xy^2$ directly and not through the sum of cubes: when we add z to the list, this value increases by $z^2 \cdot s + z \cdot s^2$ (again, here we use that $(x_1 + \ldots + x_k)^2 = x_1^2 + \ldots + x_k^2$).
- In the end, we know $x + y$ and xy.
- Need to solve a quadratic equation. There are multiple methods.
- The following should work: to solve $p(x) = 0$, where $p(x) = x^2 + ax + b$, compute the GCD of $(x + r)^{(2^{64}-1)/3} - 1$ and $p(x)$ (r is random here).
- This way, we filter out exactly one third of all non-zero elements of the field.
- A common heuristic suggests that we will recover all roots in $O(1)$ iterations.