Вказівки щодо розв'язання завдання № 29 відбірково-тренувальних зборів команди міста Києва

1. Числа

Розв'язання на 65 балів

Вважаємо, що шукані числа X мають таку саму кількість цифр L, як і число N. Якщо це не так, доповнимо початок десяткового запису числа X нулями. Від такого доповнення не зміниться ні його сума цифр, ні залишок від ділення на 2^a.

Якщо перша цифра числа X менша за першу цифру числа N, то $X < N$, незалежно від решти цифр числа X. Якщо згадані цифри збігаються, то маємо схожу задачу, але вже для чисел з кількістю цифр меншою на 1. Опишемо це рекурентним співвідношенням. Позначимо через $f(k, equal, mod, sum)$ залишок від ділення на $10^9 + 7$ кількості чисел X, які задовольняють усі такі умови:

- сума перших k цифр числа X дає остачу sum від ділення на 2^b;
- число X дає остачу mod від ділення на 2^a;
- при $equal = 1$ число X має з N однакові перші k цифр;
- при $equal = 0$ серед перших k цифр числа X існує така цифра, яка менша від відповідної цифри числа N, а всі цифри, розташовані перед нею, дорівнюють відповідним цифрам числа N;
- число X має L цифр у десятковому записі (можливо, з нулями на початку) та останні $L - k$ цифр числа X дорівнюють нулю.

Запровадимо такі позначення:

% — взяття залишку від ділення;

N_k — k-та цифра числа N ($k = 1, 2, ..., L$).

Справджуються такі рекурентні співвідношення:

$$f(k, 0, \mathrm{mod}, sum) = \sum_{i=0}^{N_k - 1} f\left(k-1, 1, (\mathrm{mod} - 10^{L-k} \cdot i)\%2^a, (sum - i)\%2^b\right)$$

$$+\sum_{i=0}^{9} f\left(k-1, 0, (\mathrm{mod} - 10^{L-k} \cdot i)\%2^a, (sum - i)\%2^b\right),$$

$$f(k, 1, \mathrm{mod}, sum) = f\left(k-1, 1, (\mathrm{mod} - 10^{L-k} \cdot N_k)\%2^a, (sum - N_k)\%2^b\right),$$

$$f(0, equal, \mathrm{mod}, sum) = \begin{cases} 1, \text{ якщо } equal = 1, \mathrm{mod} = 0, sum = 0 \\ 0, \text{ інакше} \end{cases}.$$

Відповіддю є сума: $f(L, 0, 0, 0) + f(L, 1, 0, 0)$. Усі обчислення роблять за модулем 10^9+7. Подану функцію можна обчислити за $O(L \cdot 2^a \cdot 2^b)$ операцій.

Розв'язання на 100 балів

Число 10^a кратне 2^a. Тому довільне число кратне 2^a тоді й лише тоді, коли число, записане за допомогою останніх a цифр числа у тому самому порядку, кратне 2^a. Якщо $k \le L - a$, то зі значень $f(k, equal, mod, sum)$ для всіх можливих варіантів mod ($0 \le mod < 2^a$) нам достатньо обчислити лише значення для $mod = 0$; якщо ж $mod > 0$, то заздалегідь відомо, що $f(k, equal, mod, sum) = 0$. З урахуванням сказаного вище, складність алгоритму складає $O(L \cdot 2^b + 2^b \cdot 2^a \cdot a)$.

2. Гра

Доведемо, що Сашко може перемогти, задавши *в середньому* (в сенсі математичного сподівання) лише $O(\ln N)$ питань.

Нехай a, b, c, d — різні натуральні числа від 1 до N включно. Будемо казати, що підмножина Q множини $\{1, 2, ..., N\}$ розділяє пари $\{a, b\}$ і $\{c, d\}$, якщо обидва елементи однієї з пар належать до Q, а обидва елементи іншої пари не належать до Q. Будемо казати, що сукупність множин $Q_1, Q_2, ..., Q_t$, що є підмножинами множини $\{1, 2, ..., N\}$, розділяє пари $\{a, b\}$ і $\{c, d\}$, якщо ці пари розділяє хоча б одна із множин $Q_1, Q_2, ..., Q_t$.

Знайдемо ймовірність того, що вибрана випадковим чином *одна* множина Q розділяє певні дві пари $\{a, b\}$ і $\{c, d\}$. Без втрати загальності можемо вважати, що a належить до Q (інакше розглянемо доповнення Q). Елементи b, c, d належать або не належать Q з ймовірністю 1/2 незалежно один від одного. Ймовірність того, що Q розділяє пари $\{a, b\}$ і $\{c, d\}$, дорівнює $(1/2)^3 = 1/8$. Тому ймовірність того, що множина *не* розділяє дві пари, дорівнює $1 – 1/8 = 7/8$.

Виберемо незалежно випадковим чином t підмножин множини $\{1, 2, ... , N\}$. Обмежимо зверху ймовірність того, що система не розділяє якісь дві невпорядковані пари. Ця ймовірність не перевищує такий добуток:

$$\frac{1}{2}\ C_N^2 \cdot C_{N-2}^2 \cdot \mathrm{P}\{\text{система не розділяє певні дві пари}\} < \frac{N^4}{8} \cdot \left(\frac{7}{8}\right)^t. \qquad (1)$$

Тут

$$C_n^k = \frac{n!}{k!\cdot(n-k)!} = \frac{n\cdot(n-1)\cdots(n-k+1)}{1\cdot 2\cdots k}$$

— кількість k-елементних підмножин n-елементної множини, для якої використовують і таке позначення: $\binom{n}{k}$.

Праву частину нерівності (1) можна зробити меншою, ніж довільне додатне ε таким вибором:

$$t > \frac{4\ln N - \ln\varepsilon - \ln 8}{\ln(8/7)} = O(4\ln N - \ln\varepsilon) = O(\ln N). \qquad (2)$$

Покажемо, що отримавши відповіді на запитання щодо множин Q_j, що розділяють довільну пару двоелементних множин чисел, можна вказати:

- або одне з двох чисел, що були загадані;
- або три числа, два з яких було загадано.

Позначимо через S сукупність пар, що могли дати ту саму послідовність відповідей, що була отримана після запитів щодо $Q_1, ... , Q_t$. Будь-які дві множини з S мають непорожній перетин, бо інакше якась із множин Q_j розділила би дві пари, що не перетинаються, і послідовність відповідей була б різною. Маємо:

а) якщо існує елемент a, що належить до всіх пар з S, то відповідь — $\{a\}$;

б) інакше розглянемо довільні дві пари з S. Вони мають спільний елемент, тому без втрати загальності можемо позначити ці пари як $\{a, b\}$ і $\{a, c\}$. Згідно з припущенням, не всі пари з S містять елемент a. Тому знайдеться така пара з S, що не містить a, проте — як і всі пари з S — має спільні елементи з парами $\{a, b\}$ і $\{a, c\}$. Такою може бути лише пара $\{b, c\}$. Тепер, якби в S була ще яка-небудь четверта пара, то вона не мала би спільних елементів принаймні з однією з пар $\{a, b\}$, $\{a, c\}$ або $\{b, c\}$, чого бути не може. Отже, загадана Олесею пара чисел є підмножиною трійки чисел $\{a, b, c\}$, яку й має назвати Сашко.

Алгоритм може бути таким:

1. Випадковим чином вибрати систему множин $Q_1, ... , Q_t$, що розділяє всі пари з ймовірністю, не меншою за $1 – \varepsilon$ за $O(tN)$ операцій.

2. Зробити t запитів щодо належності до множин $Q_1, \ldots, Q_t$.
3. Встановити множину S пар, що могли б дати таку саму послідовність відповідей безпосереднім перебором за $O(tN^2)$ — *найбільший внесок.*
4. Якщо S містить дві пари, що не перетинаються, то це означає, що система $Q_1, \ldots, Q_t$ не розділяє всі пари. Тоді треба повторити процедуру породження, перейшовши до кроку 1. Визначити, чи S містить дві пари, що не перетинаються, можна за час порядку $O(N^2)$. Якщо система $Q_1, \ldots, Q_t$ розділяє всі пари, то обов'язково справджуються умови одного з двох розглянутих випадків: а) або б). У випадку а) потужність S не перевищує $N - 1$ (це кількість пар, що містять деякий фіксований елемент), а у випадку б) потужність S дорівнює 3. Тому якщо потужність S одночасно більша за $N - 1$ і не дорівнює 3, то система $Q_1, \ldots, Q_t$ ніяк не може розділяти всі пари. Інакше (якщо потужність S не перевищує $N - 1$ або дорівнює 3) факт наявності «поганої» пари пар можна встановити безпосереднім перебором.
5. Визначити, який тип відповіді може вказати Сашко — а) чи б) — безпосереднім перебором за $O(N)$ у випадку а) чи за $O(1)$ у випадку б).

Математичне сподівання загальної кількості запитів Сашка дорівнює

$$(1-\varepsilon)\cdot t+\varepsilon(1-\varepsilon)\cdot 2t+\varepsilon^2(1-\varepsilon)\cdot 3t+\ldots=t(1-\varepsilon)\sum_{k=1}^{\infty}k\varepsilon^{k-1}=t(1-\varepsilon)\frac{1}{(1-\varepsilon)^2}=\frac{t}{1-\varepsilon},$$

що має порядок $O(\ln N)$ — див. співвідношення (2). З аналізу кроку 3 маємо: математичне сподівання складності роботи алгоритму становить $O(tN^2) = O(N^2 \ln N)$.

Математичне сподівання кількості запитів залежить як від N, так і від ε. Доцільно вибрати таку величину ε, що мінімізує математичне сподівання при заданому N. Відповідна функція $\varepsilon_{\min}(N)$ не має аналітичного виразу, але її можна наблизити за результатами обчислювального експерименту, тобто шляхом великої кількості повторних розігрувань гри зі сталими параметрами.

Описаний алгоритм не найефективніший при повторному породженні системи множин $Q_1, \ldots, Q_t$, бо інформацію про попередні запити ніяк не використано. Краще використовувати всю наявну інформацію. Наприклад, на кроці 4 лише збільшувати t на деяку сталу величину й робити додаткові запити доти, доки отримана система не розділить всі пари із S. При цьому вибір нових запитів може залежати від вибору попередніх запитів (наприклад, не варто повторювати деякий запит більше одного разу) або навіть від відповідей на попередні запити. Інакше кажучи, множини $Q_1, \ldots, Q_t$ не вибирають незалежно. Аналіз такого алгоритму складніший, і тут його не подано. Питання про оптимальний вибір параметрів такого алгоритму в залежності від N залишено відкритим.