National Girls' Programming Contest 2019
Problemset Analysis

Daffodil International University

Prepared and Presented By
Rafid Bin Mostofa
Tahsin Masrur

Hosted by: Toph & Ardent Programmers

November 25, 2019

A. Complex Operations

Author: Bishal Gautam

Number of Teams Solved: 15
Number of Teams Tried: 50

Abridged Problem Statement

- Array of A of length N (10^5 max) given
- Q (10^5 max) operations are given
- In update operation, subtract X (10^{18} max) from all elements in A
- In query operation (L, R, K), print the K-th smallest number among elements of A whose values are within $[L, R]$

Solution Approach

- Instead of updating all elements in the array, add them in a variable P to remember how much subtraction is pending

- Suppose $B = A - P$ i.e. B is the array you'd get if you did update instead of saving in P

Solution Approach

- The answer of query (L, R, K) in array B is actually the same as query $(L + P, R + P, K)$ in array A
- This can be solved by sorting the array A, maintaining P during update and finding $(L + P)$-th element in the sorted array with binary search and adding K to it
- Time Complexity: $O(Qlog_2(N))$

B. Zer VS EEE

Author: Tahsin Masrur

Number of Teams Solved: 0
Number of Teams Tried: 2

Abridged Problem Statement

- Graph G with N nodes and M edges are given
- You have to find if cyclic flow can exist in G abiding by given lower bound L and upper bound H on capacity of each edge

Solution Approach

- Add two new dummy nodes as source and sink
- Capacity of each edge is changed to $H - L$

Solution Approach

- Suppose $f(n) = \Sigma L$ for incoming edges to n - ΣL for outgoing edges from n
- For a node n, if $f(n) \geq 0$, add edge from source to n with capacity $f(n)$

Solution Approach

- If $f(n) < 0$, add edge from n to sink with capacity $-f(n)$
- Run a max flow algorithm like Dinic. If flow in all edges are same as capacity, answer exists
- Time Complexity: $O(N^2M)$

C. Mr. Xifu and His New School

Author: Nishat Tasnim Ahmed Meem

Number of Teams Solved: 5
Number of Teams Tried: 18

Abridged Problem Statement
Given some intervals in $[0, T)$ (that may cross the end pointer and continue from the beginning again). Find the maximum number of overlaps.

Solution Approach

- Keep two arrays *start* and *finish* where $start[t]$ will denote the number of intervals starting from t and $finish[t]$ will hold the intervals ending just before t.

Solution Approach

- Keep two arrays *start* and *finish* where $start[t]$ will denote the number of intervals starting from t and $finish[t]$ will hold the intervals ending just before t.

- For an interval $[s, e)$
  - if $s < e$, increase $start[s]$ and $finish[e]$ by 1.
  - else that means, the intervals continues to the next day. So we can break the interval to two part, $[s, T)$ and $[0, e)$. `++start[s], ++finish[T], ++start[0], ++finish[e]`

Solution Approach

- Keep two arrays *start* and *finish* where $start[t]$ will denote the number of intervals starting from t and $finish[t]$ will hold the intervals ending just before t.

- For an interval $[s, e)$
  - if $s < e$, increase $start[s]$ and $finish[e]$ by 1.
  - else that means, the intervals continues to the next day. So we can break the interval to two part, $[s, T)$ and $[0, e)$. `++start[s], ++finish[T], ++start[0], ++finish[e]`

- Iterate over time t from 0. Keep a *counter*. Decrease it by $finish[t]$ and increase it by $start[t]$. Answer would be the maximum of *counter* in $[0, T)$.

Complexity: $O(n + T)$

Tricky Case!

$[s, s)$ means the interval is 24hrs long.

1
2
12:35 12:35
08:00 09:00

The expected output is 2.

D. Rivalry Friends

Author: Anika Tahsin Chowdhury Simi

Number of Teams Solved: 11
Number of Teams Tried: 22

Abridged Problem Statement

Given an integer array A. There are certain queries.

- Update an element of the array.
- For a given range, find out the summation of $\phi(A_i)$; where ϕ is Euler's Totient Function.

Solution Approach

This can be done with Segment Tree with Point update and Range Query.

For an integer $N = {p_1}^{a_1} {p_2}^{a_2} .. {p_k}^{a_k}$,

$\phi(N) = N(1 - \frac{1}{p_1})(1 - \frac{1}{p_2})..(1 - \frac{1}{p_k})$.

Solution Approach (contd.)

- Build a segment tree with ϕ value of array elements.
- For update of A_i, update $\phi(A_i)$ in position i
- For query, find the sum of $[L, R]$ in segment tree.

Complexity: $O(N + Q \log N)$

Check out for integer overflow!

E. Fortis Fortuna Adiuvat

Author: Bakhtiar Hasan

Number of Teams Solved: 3
Number of Teams Tried: 11

Abridged Problem Statement

Given a directed graph and a certain node X. Given some queries in the form $S - T$. Find $dis(S, X) + dis(X, T)$.

Solution Approach

This problem can be solved by Dijkstra's Shortest Path Algorithm.

- Find the reverse graph of the given directed graph. In both graphs, find all nodes' distance from X.
- For a query $S - T$, $dis(S, X)$ is the distance of node S in the reverse graph while $dis(X, T)$ is the distance of T in the provided graph.

Complexity: $O((V + E) \log E + Q)$

About CPU Limit Exceeded

Many teams received TLE with the same approach.

Which can be avoided via the optimization in Dijkstra. For a node that's currently popped from the queue, check whether the distance its carrying is greater than the actual distance calculated so far for that node.

If its already bigger, no point in relaxing any node via this one anymore.

F. Piece of Cake

Author: Shajia Annoor Joty, Mohammad Abdullah Matin Khan

Number of Teams Solved: 37
Number of Teams Tried: 50

Abridged Problem Statement
For a triangle ABC, given length of side AB, median AD and angle BAD. Calculate $AB^2 + AC^2$.

Solution Approach

If we want to get the sum of the area of the two square shaped cake,$(AB^2 + AC^2)$ we can apply Appollonius's theorem.

According to the theorem, $AB^2 + AC^2 = 2(AD^2 + BD^2)$.

We can get BD from the given length of AB , AD and the given angle BAD. And we already have the length of AD.

The Sine issue

Many teams received "Wrong Answer" verdict due to using the Sine Law to calculate some length.

The reason: `https://en.wikipedia.org/wiki/Law_of_sines#The_ambiguous_case_of_triangle_solution`

G. Is Bita Happy?

Author: Risal Shahriar Shefin

Number of Teams Solved: 0
Number of Teams Tried: 24

Abridged Problem Statement

Given some integers a, k, L, R, you have to find out the parity of $\sum b$ for all b for which `(a & b) & (1 << k)` is non-zero.

Solution Approach

An observation is, for a range $[L, R]$ we have to find the summation of all b which has k^{th} bit on.

Another observation is, if odd number is odd times their sum is odd, otherwise even. So ultimately, our problem is reduced to counting the number of odd numbers in $[L, R]$ which has k^{th} bit on.

Solution Approach (contd.)

One way to do this is Digit-DP.

Formulate a function $f(x)$ which counts the same value for $[0, x-1]$. In binary form, iterate from left to right and keep the prefix same as x. Now for each digit to put (0 or 1), if the number gets lesser than x, count the parity of all possible values to make with same prefix via combinatorics or case-analysis.

Complexity: $O(\log(L+R))$

Solution Approach #2

Another way to this is with another observation.

Notice that a number which has k^{th} bit on comes each 2^k values after and forms a block of 2^k number . We have to simply count how many blocks fit in $[L, R]$, how many odd numbers in a block and we have to take care of partial blocks from where $[L, R]$ range starts or on where $[L, R]$ range ends

H. The Story of Stringland

Author: Sadman Rizwan

Number of Teams Solved: 0
Number of Teams Tried: 2

Abridged Problem Statement

Given a string S and some queries. For each query, find out the k^{th} lexicographically smallest palindrome or report there are not k palindromes altogether.

Solution Approach

It was meant to be a stopper.

Judge solution used Palindromic Tree (can be found in `adilet.org`) to generate all palindromes and Suffix Array to sort the palindromes. Complexity: $O(N \log N + Q)$

Alternate writer used Palindromic Tree to generate all palindromes. But he used Hashing to sort the palindromes.
Complexity: $O(N \log^2 N + Q)$

Both of the solutions precalculates the result and responds to query in constant time.

Another approach exists solely with Hashing which doesn't need Palindrome Tree.

I. Solve This Giveaway Problem First

Author: Hasinur Rahman

Number of Teams Solved: 0
Number of Teams Tried: 17

Abridged Problem Statement

- N, P and a list L of positive integers (max 5 such integers) are given
- Count how many numbers in the range $[1, N^P]$ are divisible by at least one of the numbers from the list L

Solution Approach

- The problem can be reduced by inclusion-exclusion
- Consider all 2^5 subsets of L and find the count of numbers in $[1, N^P]$ divisible by the LCM of the subset

Solution Approach

- To find the count of numbers in $[1, N^P]$ divisible by x, answer is basically $(N^P)/x$, note this is an integer division
- But we need to find $\frac{N^P}{x}\%m$ and normal inverse modulo may not work since this is an integer division

Solution Approach

- Basically, we have to find $\frac{N^P - N^P \% x}{x} \% m$
- So, find $N^P - N^P \% x$ first and then multiply inverse modulo of x to find the answer
- Time Complexity: $O(2^5 log_2 LCM + 2^5 M)$ where $M =$ summation of the length of N over all test cases

Solution Approach #2

- Generate LCM for each combination of numbers from L, there will be 2^S such LCMs
- Now we need to calculate floor$\frac{N^P}{LCM}$ mod $10^9 + 7$
- As inverse modulo will not work due to integer division, divide and conquer Technique will help us.

Solution Approach #2

- Consider $\frac{N \times N}{LCM}$ mod $10^9 + 7$

- Let $N = (a \times LCM + b)$
  Then
  $N \times N = (a \times a \times LCM) \times LCM + 2 \times a \times b \times LCM + b \times b$
  So, new a $= (a \times a \times LCM) + 2 \times a \times b + \frac{b \times b}{LCM}$
  And new b $= (b \times b)\%LCM$

- This way we can get the result by using divide and conquer

- Time Complexity: $O(2^5 log_2 LCM + 2^5 M)$ where $M =$ summation of the length of N over all test cases

J. Let's See

Author: Shahwat Hasnine

Number of Teams Solved: 23
Number of Teams Tried: 57

Abridged Problem Statement

$$f(x) = \sum_{i=1}^{x} (x \pmod i);\ f(0) = 0$$

Find $\sum x$ for all $L \leq x \leq R$ where $f(x) = f(x-1)$.

Solution Approach

It's a pattern!

Such x are $1, 2, 4, 8, 16, 32, 64, 128, 256, \ldots$

Solution Approach

It's a pattern!

Such x are $1, 2, 4, 8, 16, 32, 64, 128, 256, ...$

General form: 2^k.

- For a query $L - R$, just iterate over every power of two and if it fits the range, sum it up!

Complexity: $O(\log R)$.

K. Happy Sub-Sequence

Author: Dhruba Mitra

Number of Teams Solved: 0
Number of Teams Tried: 1

Abridged Problem Statement

- Array p_1, p_2, ... p_m is called to be "Happy" if it is not empty and for every i, p_i is divisible by i.
- Array A of length N (10^5 max) where value of element are less than or equal to 10^6
- Find the number of happy sub-sequences of A

Solution Approach

- Pre-calculate the divisors of each number in $[1,10^6]$
- For i-th array element A_i, iterate over each divisor d and consider this element to be the d-th element in some sub-sequence

Solution Approach

- Let, $f(d)$ be the number of sub-sequences you can make with d-th element being A_i
- If you iterate over d descendingly, $f(d)$ can be updated as $f(d) + f(d-1)$
- Time Complexity: $O(Mlog_2M + Nlog_2(M))$ where M is the max value of number in array i.e. 10^6

L. Information Retrieval

Author: Tahmid Rahman Laskar

Number of Teams Solved: 0
Number of Teams Tried: 1

Abridged Problem Statement

For a given query, some candidate answers are provided. You have to sort these answers relative to their number of match (rules in statement) and output it in the sorted order.

Solution Approach

Its practically a careful implementation with sortings.

First of all, if a word has a similar word, then both words should be considered as same, which can be done via C++ `map`. Then for each distinct word in the query part, find out if it appears in the answer; which can be done via binary search. After finding out the number of match, sort the answers of each distinct query based on the match count. The orders of different queries should be strictly maintained based on how they are given as input.

M. Ada Lovelace

Author: Tarif Ezaz, Tahmid Rahman Laskar

Number of Teams Solved: 139
Number of Teams Tried: 142

Abridged Problem Statement

- Easy giveaway problem requiring only basic programming skills
- Print the sum of ASCII values of 'A', 'd' and 'a'

Solution Approach

- ASCII value of 'A', 'd' and 'a' are 65, 100, 97 respectively
- Sum of these values is the answer 262
- Time Complexity: O(1)

Our Gratitude To Full Judge Panel

- Tarif Ezaz, Software Engineer, Aubichol IT (Judging Director)
- Mahmud Ridwan, Software Engineer, Furqan Software (Judging Coordinator)
- Anika Tahsin Chowdhury, Software Engineer, SRBD
- Araf Al-Jami, Software Engineer, Selise
- Bakhtiar Hasan, Faculty Member, IUT
- Bishal Gautam, Software Engineer, Careem
- Dhruba Mitra, RUET
- Fahim Ferdous Neerjhor, Faculty Member, BRAC University
- Fahim Shahriar Shakkhor, IUT
- Hasibul Haque Himel, IUT
- Hasinur Rahman, RUET

Full Judge Panel (cont'd)

- Meem Ahmed, Faculty Member, Metropolitan University
- Mohammad Abdullah Matin Khan, Software Engineer, Aubichol IT
- Rayhan Chowdhury, Software Engineer, Ridmik Inc.
- Rafid Bin Mostofa, BUET
- Risal Shahriar Shefin, RUET
- Sadman Rizwan, Premier University
- Shajia Annoor, Software Engineer, Aubichol IT
- Shahwat Hasnaine, RUET
- Tahsin Masrur, Software Engineer, SRBD
- Tahmid Rahman, Graduate Student, York University
- Zubayet Zaman Zico, Software Engineer, Enosis Solutions