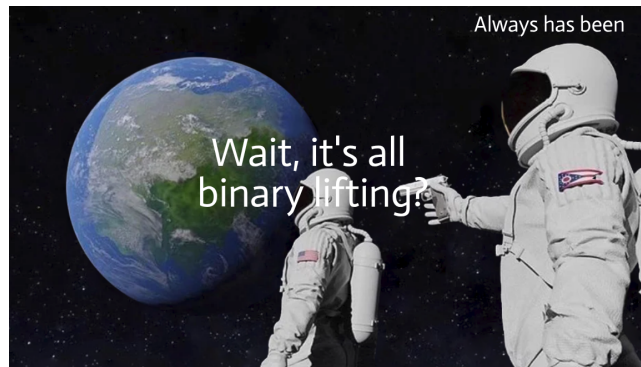


Анализ на задача guess

Решения, условие: Димитър Шапатов; Идея, анализ: Кирил Зашев



Задачата на пръв поглед е много лесна – стандартно двоично търсене. Това, което прави задачата интересна, са забранените оператори – целта им е да ни попречат да направим стандартното двоично търсене за тази задача. Освен това, ефективността на нашето решение трябва да е същата като на двоичното търсене. Как е възможно това?

Предполагаме, че доста от участниците са чували за binary lifting и LCA. Всъщност тук решението за 100 точки много прилича на намирането на LCA на два върха, след като сме ги изравнили на една и съща дълбочина. Идеята е следната – пробваме да "скочим" с най-голямата степен на двойката. Ако заявката ни върне false, това означава че сме прескочили търсеното ни число, затова пробваме със следващата по-малка степен на двойката. Ако заявката върне true, прибавяме дължината на този скок към отговора ни и продължаваме по същия начин търсенето от новата позиция. Всъщност, това което доказва решението на тази задача е, че всяко двоично търсене може да бъде замесетено с това двоично подскачане, но в общия случай няма смисъл. Удобството тук е, че с това двоично подскачане задачата може да се напише единствено с оператор + чрез смятане на степени на двойката със събиране.

Интересен факт – понеже в задачата е забранен операторът /, не можем да използваме `#include <bits/stdc++.h>`.