Game Theory

Victor Cao, Jason Zhang

Introduction

Game theory is one of the most important categories of competitive programming.

It involves solving problems where players make optimal moves in a finite, perfect information game to determine a winner.

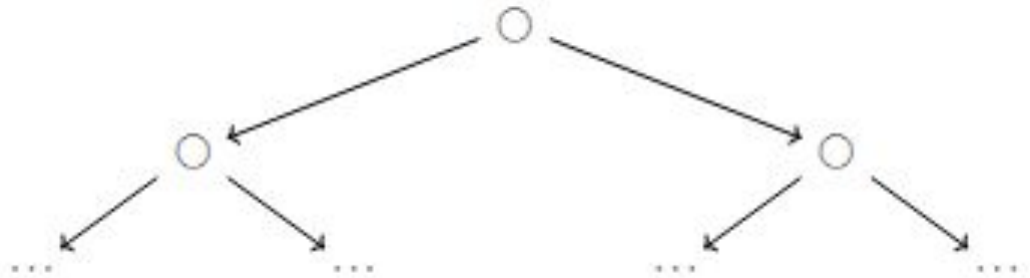
Disclaimer

There is no generalization to solve these sorts of problems (besides a few cases).

We hope to give some examples of ranging difficulty to build intuition.

Tree Semantics

It is useful to think of games as trees. Sometimes, games are even played on trees!



Greedy Algorithms

These algorithms choose the most optimal move in the short term without considering future consequences. Here is a simple codeforces example (800)

There are n players at a circular table, the *i*-th player has *a_i* dishes to eat. They take turns eating, and any player can go first.

During their turn, if the player has a dish remaining, they must eat exactly one dish. Then the next player starts their turn and continues until all dishes are finished.

The player hwo eats the last dish is the winner. Determine how many players can be winners

Greedy Algorithms

Here is a more involved codeforces example (1500)

Alice and Bob are playing game on array *a* initially containing *n* positive integers, Alice goes first

If *a* is non-decreasing, the game immediately ends. Otherwise, the player can choose any element *x* from the array and positive integers $1 < y,z < x$ such that $x = yz$, and replace *x* with *y* and *z*, in any order. If no such move is possible, the game ends.

When the game ends, if *a* is non-decreasing, Bob wins. Otherwise, Alice wins. Determine who wins optimally.

Hints

Consider when the array will have no possible moves.

Is there any way for Alice to ensure victory?

Solution

If a is non-decreasing, Bob instantly wins

If there is a value within A that is divisible by 2 primes *p* and *q*, Alice can instantly ensure victory if it is her move. This is because she can choose the larger prime to be second, and it can never go before the other one

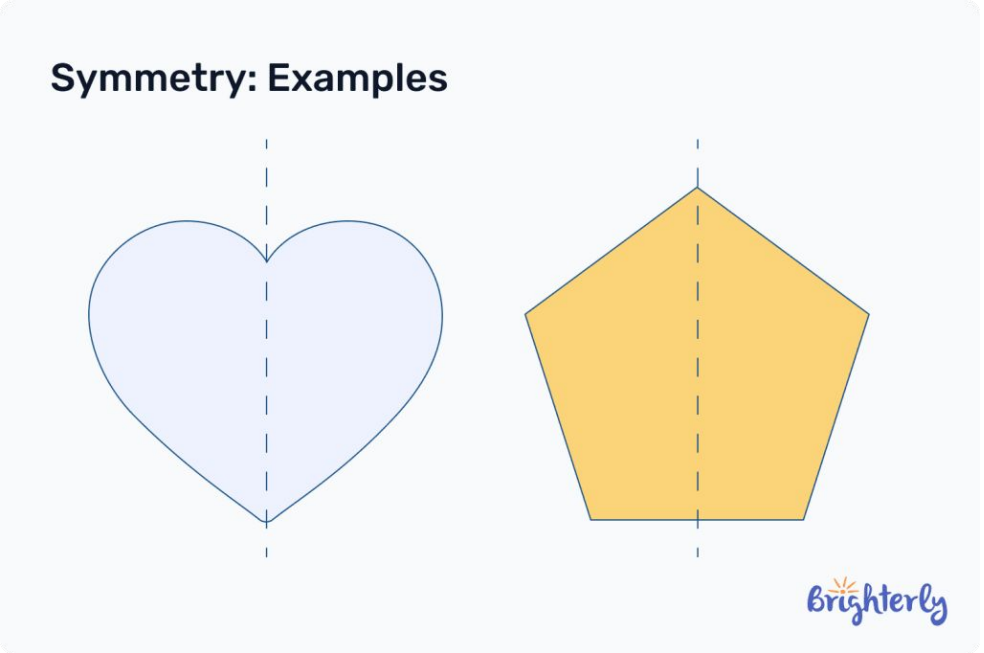
Therefore, the only case left is if all values in a are of the form p^n, or 1. The primes/1 must be sorted in a way such that the primes are non-decreasing

Invariants

An invariant is some property (about either a strategy or position) that does NOT change after some move, transformation, etc.

Symmetry

Tl;dr: this is helpful



In game theory, a **symmetric game** is a game where the payoffs for playing a particular strategy depend only on the other strategies employed, not on who is playing them. If one can change the identities of the players without changing the payoff to the strategies, then a game is symmetric. Symmetry can come in different varieties. **Ordinally symmetric games** are games that are symmetric with respect to the ordinal structure of the payoffs. A game is **quantitatively symmetric** if and only if it is symmetric with respect to the exact payoffs. A **partnership game** is a symmetric game where both players receive identical payoffs for any strategy set. That is, the payoff for playing strategy *a* against strategy *b* receives the same payoff as playing strategy *b* against strategy *a*.

B. Tree Tag

time limit per test: 1 second
memory limit per test: 256 megabytes

Alice and Bob are playing a fun game of tree tag.

The game is played on a tree of n vertices numbered from 1 to n. Recall that a tree on n vertices is an undirected, connected graph with $n-1$ edges.

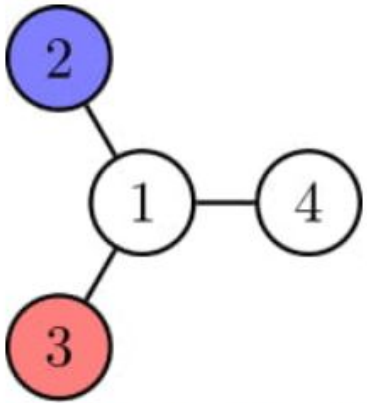
Initially, Alice is located at vertex a, and Bob at vertex b. They take turns alternately, and Alice makes the first move. In a move, Alice can jump to a vertex with distance **at most** da from the current vertex. And in a move, Bob can jump to a vertex with distance **at most** db from the current vertex. The distance between two vertices is defined as the number of edges on the unique simple path between them. In particular, either player is allowed to stay at the same vertex in a move. Note that when performing a move, a player only occupies the starting and ending vertices of their move, not the vertices between them.

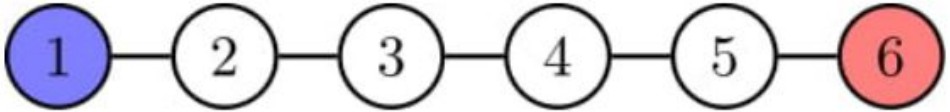
If after at most 10^{100} moves, Alice and Bob occupy the same vertex, then Alice is declared the winner. Otherwise, Bob wins.

Determine the winner if both players play optimally.

https://codeforces.com/contest/1404/problem/B

$(2 \leq n \leq 10^5, 1 \leq a, b \leq n, a \neq b, 1 \leq da, db \leq n-1)$


2
1
4
3


1
2
3
4
5
6

Exercise for the audience

There is a tree with vertices 1-N. The edges are denoted as (x_i, y_i).

Alice and Bob play a game against each other. Starting with Alice, they alternate the following move:

Select an extant edge and remove it, creating two separate connected components. Remove the component that does not contain vertex 1.

A player loses when no moves are available

Exercise for the audience

Bob is tired of losing to Alice and, to ensure he doesn't lose again, decided to choose a game in which he is guaranteed to win. Bob has thought of a number from 1 to n, where it is known that $n = 2^d$ for some non-negative integer d. Initially, Alice **knows** whether the chosen number is even or not.

In one move, Alice can either halve the number or subtract 1. Alice can only halve the number if the current number is even. Only Alice takes turns.

After her move, Alice receives a response from Bob: either -1, which means the number has become 0 and Alice has won, or a non-negative integer x. If we denote the current number as a, then for x the following conditions hold simultaneously:

1. a is divisible by 2^x.
2. a is not divisible by 2^{x+1}.

For example, if $a = 5$, then $x = 0$, since 5 is divisible by $2^0 = 1$ and not divisible by $2^1 = 2$, and if $a = 12$, then $x = 2$, since 12 is divisible by $2^2 = 4$ and not divisible by $2^3 = 8$.

It can be shown that for any integer $a > 0$, there exists a unique such x.

Bob is still afraid that Alice will win, so the game will have no more than k moves. Additionally, Bob wants to maximize his chances of winning, so he wants to play as many games as possible. Given n and k, calculate the number of initial numbers from 1 to n such that Alice, playing optimally, cannot win in no more than k moves.

More Problems

https://codeforces.com/problemset/problem/2203/D ~1700

https://codeforces.com/contest/1404/problem/D ~2800

https://codeforces.com/contest/1375/problem/F ~2600

https://codeforces.com/problemset/problem/2200/A ~800

https://codeforces.com/problemset/problem/2200/E ~1500

https://codeforces.com/problemset/problem/2199/A ~1000

https://codeforces.com/problemset/problem/2196/A ~1200

https://codeforces.com/problemset/problem/2184/D ~1600