

AtCoder 過去問とりくみ記録： 2023 GW

2023-05-06 (Sat)

ABC291F - Teleporter and Closed off

DP で解けることは解けるけど間に合わないだろうなあと思ったやつは、案の定間に合わなくて*1、ほいじゃあというので考え直したやつで通った。

間に合わないって分かりきっているアイデアをわざわざ実装しないようにしないとね。

提出例：41205263*2

2023-05-04 (Thu)

ABC294G - Distance Queries on a Tree

オイラーツアー。どうにか自力で通すなど。

提出例：41148690*3

ABC293G - Triple Index

区間 $[l, r]$ に対する答えが求まっている時に、その区間を一つ伸ばしたり縮めたりした時の答えも求まるなあ、とは思ったもののそこから先が思いつかず。

Mo のアルゴリズムというものがあるらしい。 $(l[i], r[i])$ の列を、 l, r どちらも昇順になるようにソートすることはできないのだけど、 l を区間ごとに分割して、その分割の中では r が昇順になるようにソートする。(公式解説に計算量含めた説明あり)

B を正しく求められなくて、RE (多分ゼロじよざん) や TLE (int あふれで B が意図せず小さくなった) になったりした。

提出例：41154823*4

ABC294F - Sugar Water 2

ある濃度 x を決めて、その濃度以上の水溶液を K 個以上作れるかどうかという判定問題にすれば、ニブタンで求まる。

…のだけど、判定問題を上手に解かないと TLE になってしまう*5

解説にあるように、濃度を決めると、その濃度ちょうどになるための砂糖の量がわかるので、各水溶液に対して、目標濃度に対する砂糖の過不足量を求めることができる。そうすることで判定問題の部分にも二分探索が使えて、間に合う。

それはそうと、 K が int に収まらない場合があって、それでバグったりしてしまった。

提出例：41159986*6

*1 <https://atcoder.jp/contests/abc291/submissions/41204876>

*2 <https://atcoder.jp/contests/abc291/submissions/41205263>

*3 <https://atcoder.jp/contests/abc294/submissions/41148690>

*4 <https://atcoder.jp/contests/abc293/submissions/41154823>

*5 <https://atcoder.jp/contests/abc294/submissions/41159152>

*6 <https://atcoder.jp/contests/abc294/submissions/41159986>

ABC293F - Zero or One

素朴にやっていたら間に合いっこないので、 b の候補を絞る必要がある。

私は、 b 進数で表現した時の最下位に着目して、これが 0 か 1 になるということは、 b は N の約数か、 $N - 1$ の約数だな、ってやってみたのだが TLE になってしまった。これでは、まだまだ候補が多すぎるようだ。

解説にあるように、 b 進数表示した時の桁数は 2 進の時が最大。 2 桁になるのは、 $b = N, N - 1$ の時。それ以外の各桁数になる b を求めて候補とすれば良い。

…のだが、意外とバグってなかなか合わなかった。

提出例：41162530^{*7}

2023-05-03 (Wed)

TDP-P うなぎ

$dp[i][j][k]$ を、部分木 i において j 本のパスを選ぶ選び方のうち、頂点 i (部分木の根) の次数が k であるものの個数として DP していくんだけど、結構難しかった。

二回ほどやってみて、やっと腑に落ちた気がするので、今度は自力で解こう。

提出例：41139446^{*8}

ABC296G - Polygon and Points

図形の問題。多角形を $N - 1$ 個の三角形に分けて、三角形の中に入るかどうかの問題に帰着できるわけですが、 N が結構多いので毎回全部試していると間に合いそうにもない。

そこでニブタンですよ、つてのに気づいたらあとは書くだけだった。

提出例：41141099^{*9}

ABC295G - Minimum Reachable City

p_i で指定されるグラフは、 1 を根とし、根に近いノードの方が番号が小さくなるような木のような形をしている (ただし辺は根から葉の向きの方向を持つ)。

1 番目のクエリで作られる強連結成分を Union-Find 木で管理しながら、各強連結成分の最小頂点の番号を持つようにすれば良い。

提出例：41147123^{*10}

2023-05-02 (Tue)

ABC279G - Constrained Nim 2

Grundy 数の問題。 A_i の範囲が大きいので、定義通り求めていては間に合わない。そこで、実験して Grundy 数が周期的になっていることを見抜こう、という。

^{*7} <https://atcoder.jp/contests/abc293/submissions/41162530>

^{*8} <https://atcoder.jp/contests/tdpc/submissions/41139446>

^{*9} <https://atcoder.jp/contests/abc296/submissions/41141099>

^{*10} <https://atcoder.jp/contests/abc295/submissions/41147123>

提出例 : 41122018*¹¹

ABC276F - Simultaneous Swap

まず、多重集合として一致していなければあわない。

A, B 同時に置換が起こるので、A の転倒数と B の転倒数の和の偶奇は変わらない。また、最終的に A, B を一致させられた場合には両者の転倒数は等しくなるので、和は偶数。つまり、初めから転倒数の和は偶数でなければならない、というわけ。

逆に、転倒数の和が偶数（転倒数の偶奇が同じ）であれば必ず一致できるかとか、重複がある場合とか、もう少し考えるべきことがある。あとでもう一度復習しよう。

提出例 : 41126530*¹²

2023-05-01 (Mon)

ABC299F - Square Subsequence

部分列 DP だよなあということ。

ただ、私が部分列 DP について知っていることといえば、同じ部分列を重複してカウントしてしまわないために、同じアルファベットが複数ある場合にはなるべく左から選ぶようにしましょう（そのために LUT (look up table) を使う）というくらい。

それでは解けなくて解説を見た。なるほどーと思った（それが思いつけなかったから解けなかった）のは以下：

- 部分列 TT の後半が始まる位置 x ごとに DP をやる
- dp は、見ている部分列の最後の位置に着目するのだが、 $dp[i][j]$ のように前半の末尾 i と、それに対応する後半の位置 j の二次元で扱う
- 最終的な答えとしてカウントできるのは、 $lut[i+1][s[x]]$ が x になるもののみ

どれも、言われてみれば、なるほど！なんだけど。

提出例 : 41093939*¹³

ABC299G - Minimum Permutation

これも自力で解けなかった…。図を用いた解説¹⁴ がわかりやすかった。

提出例 : 41096155*¹⁵

ABC298F - Rook Score

これは普通に自分で解けた。

提出例 : 41097423*¹⁶

*¹¹ <https://atcoder.jp/contests/abc297/submissions/41122018>

*¹² <https://atcoder.jp/contests/abc296/submissions/41126530>

*¹³ <https://atcoder.jp/contests/abc299/submissions/41093939>

*¹⁴ <https://kanpurin.hatenablog.com/entry/2023/04/30/234125>

*¹⁵ <https://atcoder.jp/contests/abc299/submissions/41096155>

*¹⁶ <https://atcoder.jp/contests/abc298/submissions/41097423>

ABC297F - Minimum Bounding Box 2

$h * w$ の長方形の中に K 個の黒マスを入れる場合の数 $f(h, w)$ は容易に求まるのだが、それを用いて、 K 個のマスを含む最小の長方形が $h*w$ になる場合の数を求めるのに包除原理を用いる必要がある。

あと、 $f(h, w)$ をあらかじめ前計算しておかないと時間内に収まらなかった。

提出例 : 41103973^{*17}

2023-04-30 (Sun)

ABC300F - More Holidays

コンテスト中、20分ほどで書いてみたんだけどサンプルも合わせられないまま時間切れしたやつ。

「開始位置を $[0, N)$ で動かしつつ、おのおのニブタン」

と聞けば、うおお、なるほど！って。

提出例 : 41067339^{*18}

ABC300G - P-smooth number

これも自力で解けず。まず、時間オーバーしてもいいので素朴に解いてみるというのが実装ミスでなかなかあわなかった（ローカルの配列を初期化せずに使うという初歩的ミスにより）りして…。

解説にもある通り、素朴に解くと TLE なんだけど、半分全列挙の要領で半分の問題を二個に分割してやればいけると、と言う。

提出例 : 41076054^{*19}

^{*17} <https://atcoder.jp/contests/abc297/submissions/41103973>

^{*18} <https://atcoder.jp/contests/abc300/submissions/41067339>

^{*19} <https://atcoder.jp/contests/abc300/submissions/41076054>