ARC062 / ABC046 解説

sigma425

A: AtCoDeer くんとペンキ

$a = b = c$ の場合は 1 を、$a = b \neq c$ または $a = c \neq b$ または $b = c \neq a$ の場合は 2 を、それ以外の場合は 3 を出力すれば良いです。

B: AtCoDeer くんとボール色塗り

まず左端のボールに塗る色を決めます。これは K 通りあります。さらに左から順番に塗る色を決めていきます。この時、これまでどのように塗っていようが、"ひとつ左のボールに塗られている色" 以外の $K - 1$ 通りの色が塗れるので、全体では $K \cdot (K - 1)^{N-1}$ 通りになります。答えが大きいので、塗り方をすべて試す方法では間に合いません。

C: AtCoDeer くんと選挙速報

各途中状態で、できるだけ投票数が少ない方がよいです (投票数を増やすのはいつでも出来るので)。今高橋君に A 票、青木君に B 票入っていて、次に満たすべき比率が $x : y$ だとすると、$A \leq nx \ \wedge \ B \leq ny$ なるような最小の自然数 n を取れば、次にあり得る最小の得票数は nx, ny であることがわかります。このような n は $max(\lceil A/x \rceil, \lceil B/y \rceil)$ で計算できます。はじめに $A = 1, B = 1$ として、これを N 回繰り返すことで $O(N)$ で答えが得られます。

D: AtCoDeer くんと変なじゃんけん

相手が P 回パーを出したとします。まず自分が全てグーを出したとすると、この時の得点は $-P$ 点です。自分のグーのどれかひとつをパーに変えることを考えると、

- そのターンの相手の手がグーのとき　・・・ あいこ (0 点) から勝ち (1 点) に変わるので、 得点は 1 増えます。
- そのターンの相手の手がパーのとき　・・・ 負け (−1 点) からあいこ (0 点) に変わるので、 得点は 1 増えます。

従って、自分や相手がどのターンでパーを出したかによらず、得点は (自分がパーを出した回数)-(相手がパーを出した回数)　になります。なのでできるだけ多くパーを出せばよいです。グー, パー, グー, パー,... と出すことでこれが実現できて、$\lfloor N/2 \rfloor$ 回出せます。よって答えは $\lfloor N/2 \rfloor - P$ です。

E: AtCoDeer くんと立方体づくり

立方体の上面と底面のパネルとその向きを、どれを使うか全探索します。すると 8 つの頂点の色がすべて決まるので、あとは各側面に対し置けるパネルの数を数えれば良いです。これはパネルを”正規化”した形で表して、回転による差異を無視できるようにしてから map に入れて数を数えるなどの方法でできます。(例えば、入力のように 4 つの数の列でパネルを表すとして、回転によって 4 通りの列が考えられるがそのうち辞書順最小のものでパネルを表す など)

他の部分に使ったパネルは使えないのでちゃんと使った分は引くことと、パネルの向きの置き方が何通りあるか (例えば 4 点同じ色のパネルだと,4 通り置ける) に注意すれば、側面の置き方を数えることが可能です。あとは、同じ立方体を何回重複して数えているか (これは数え方によって異なります) で割れば答えが求まります。計算量は $O(N^2 logN)$ ですが定数倍はかなり重いです。

F: AtCoDeer くんとグラフ色塗り

まず二重 (頂点) 連結成分分解します。異なる辺 x,y に対し, x と y が同じ成分に属する $\Leftrightarrow$ x と y を両方含む単純なサイクルがある なので、各操作はある連結成分内でしか行なえません。なので、各連結成分ごとに独立に考えられます。

1. 成分が辺一本のとき

   塗り方は K 通りです。

2. 成分がサイクルをなすとき

   サイクルを塗る方法なので、これはポリアの定理から、サイクルを n 頂点とすると、

   $$\frac{1}{n}\sum_{i=0}^{n-1} K^{gcd(n,i)}$$

   で計算できます。

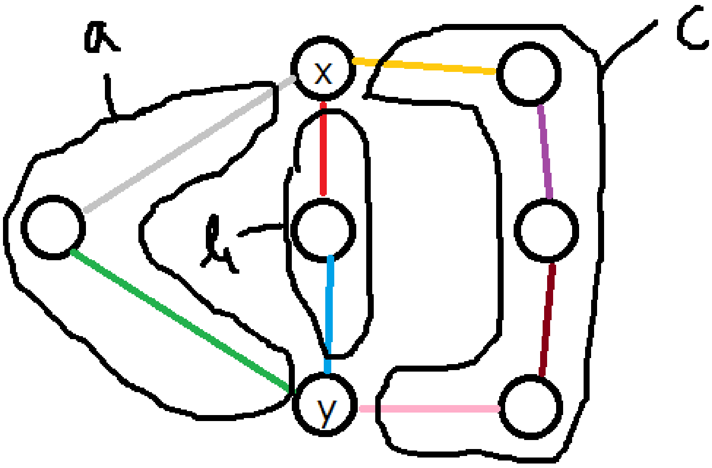
3. それ以外

   この時成分内で任意の置換が作れることがわかります。従って、連結成分の辺の個数を m 本とすると、答えは、${}_{m+K-1}C_{K-1}$ になります。

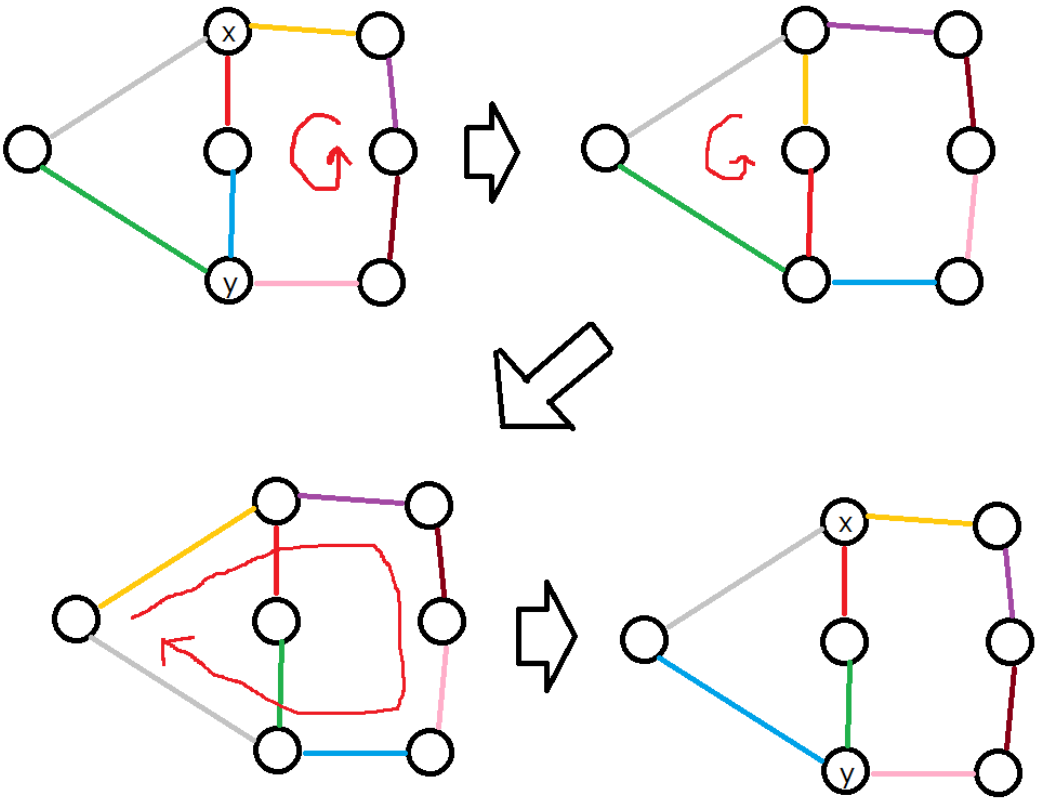
従って各連結成分ごとにこれらの値をかけ合わせると答えが求まります。

3 の場合に任意の置換が作れることの証明をします。

まず次の図のように、異なる頂点 x,y を,3 つの edge disjoint な (互いに辺を共有しない)path a,b,c が結んでいる状況を考えます。このグラフを便宜的に G と呼びます。



このとき、次のように操作をすることで、頂点 y に接続する path a の辺と path b の辺を swap することが出来ます。

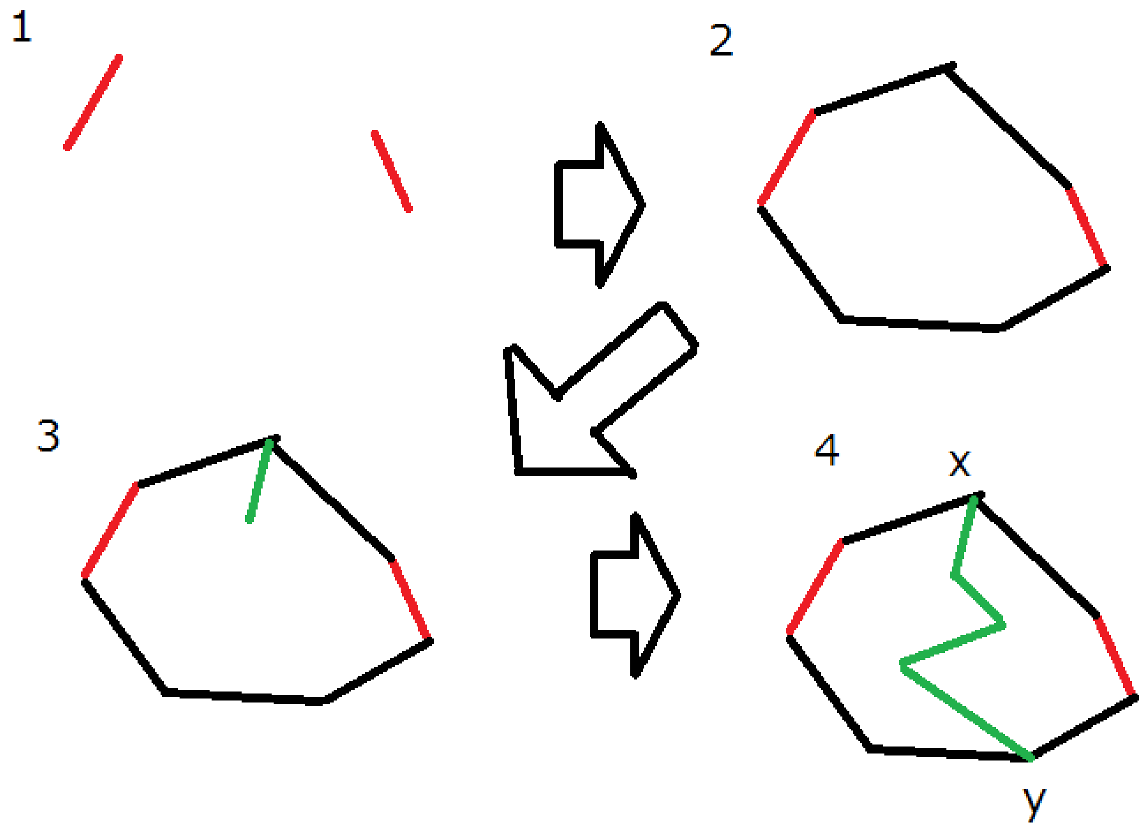


また、この操作を使って、グラフ G の任意の隣接する二辺を swap することが可能です。具体的には、まずその二辺が y に隣接するようにサイクルを回して、上の操作を行い、はじめにサイクルを回した部分の逆操作を行えばよいです。

これを使うと、グラフ G 上の任意の二辺が swap 出来ます。具体的には、その二辺を両端とする path をとり、その path 上の隣接する辺を swap していくことを二回すると出来ます。

あとは、元のグラフの任意の異なる二辺に対して、その二辺を含む、グラフ G のような構造が取れることが示せれば、その二辺は swap できるため結局全ての置換が作れることがわかり、証明は完了します。よってこれを示します。

まず二辺は同じ二重連結であり成分は辺一本ではないので、その二辺を含むサイクルがとれます (図の 2)。また、成分はただのサイクルではないため、そのサイクルとは別に辺が含まれます。当然二重連結成分は連結なので、そのような辺としてサイクルに接続しているものがとれます (図の 3)。この辺とサイクルが同じ二重連結成分に属するためには、また別のサイクルであって、サイクルの辺とこの辺をともに含むサイクルの存在が必要です。そのサイクルの辺を取ってくると、目的の構造が取れていることがわかります (図の 4)。



よって示されました。

二重 (頂点) 連結成分分解は、関節点を求めるのと同様なアルゴリズムで O(N+M) のものがありますが、今回の制約だと、dfs してサイクル基底を得て、各サイクルの辺を unionfind でつなぐ O(MNlogM) なども許されます。