合併　解説

笠浦一海

March 23, 2019

問題概要

- 木が与えられる．それぞれの頂点がいずれかの州に属する．
- 同じ州に属する頂点が同じ集合に属するように木をふたつの連結部分木に分けることができるとき分裂可能
- ふたつの集合をひとつにマージすることを繰り返して木を分裂可能でなくしたい
- マージの最小回数を求めよ

分裂可能性の判定

- 「同じ州に属する頂点が同じ集合に属するように木をふたつの連結部分木に分けることができる」かを判定する方法を考える

分裂可能性の判定

- 「同じ州に属する頂点が同じ集合に属するように木をふたつの連結部分木に分けることができる」かを判定する方法を考える
- 州の数が k のとき頂点の分け方: 2^k 通り

分裂可能性の判定

- 「同じ州に属する頂点が同じ集合に属するように木をふたつの連結部分木に分けることができる」かを判定する方法を考える
- 州の数が k のとき頂点の分け方: 2^k 通り
- 連結判定: $O(N)$

サブタスク 1

- 州のマージの仕方を全列挙

サブタスク 1

- 州のマージの仕方を全列挙
- 簡単な考察で $K!$ を超えないことがわかる

サブタスク 1

- 州のマージの仕方を全列挙
- 簡単な考察で $K!$ を超えないことがわかる
- 全体で $O(K!2^K N)$

サブタスク 1

- 州のマージの仕方を全列挙
- 簡単な考察で $K!$ を超えないことがわかる
- 全体で $O(K!2^K N)$
- $N \leqq 100, K \leqq 7$ のサブタスク 1 なら解ける

分裂可能性の判定（もっと賢く）

- 木をふたつの連結部分木に分ける→ひとつの辺で切る

分裂可能性の判定（もっと賢く）

- 木をふたつの連結部分木に分ける→ひとつの辺で切る
- ひとつの辺で切ってその両方の側の頂点を含む州がなければ良い

分裂可能性の判定（もっと賢く）

- 木をふたつの連結部分木に分ける→ひとつの辺で切る
- ひとつの辺で切ってその両方の側の頂点を含む州がなければ良い
- 判定が $O(N^2)$ でできた

分裂可能性の判定（もっと賢く）

- 木をふたつの連結部分木に分ける→ひとつの辺で切る
- ひとつの辺で切ってその両方の側の頂点を含む州がなければ良い
- 判定が $O(N^2)$ でできた
- 全体で $O(K!N^2)$ → サブタスク 1 のみ

考察

- 「その辺で切ったときにその両方の側の頂点を含む州がない辺」を生きた辺，それ以外を死んだ辺と呼ぶことにする

考察

- 「その辺で切ったときにその両方の側の頂点を含む州がない辺」を生きた辺，それ以外を死んだ辺と呼ぶことにする
- 生きた辺を全部潰したい
- 両方の側から州を一つずつ選んでマージすれば潰せる

別の問題に還元

- 死んだ辺をすべて縮約してできるグラフを考える
- 「ふたつの頂点を選び，それらの間にある辺をすべて縮約する」を繰り返して一点に潰す

別の問題に還元

- 死んだ辺をすべて縮約してできるグラフを考える
- 「ふたつの頂点を選び，それらの間にある辺をすべて縮約する」を繰り返して一点に潰す
- ↑の操作を最低で何回行えば良い…?

答え

- 答え: $(\text{葉の数} + 1)/2$
- ここで葉とは次数 1 の頂点のこと

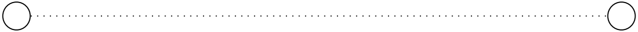
答え

- 答え: (葉の数 + 1)/2
- ここで葉とは次数 1 の頂点のこと
- 証明: 一回の操作で葉は高々 2 個しか消えない→上限

答え

- 答え: (葉の数 + 1)/2
- ここで葉とは次数 1 の頂点のこと
- 証明: 一回の操作で葉は高々 2 個しか消えない→上限
- 逆に葉の数が 3 つでなければ，一回の操作で葉の数を 2 つ減らせることが証明できる

証明

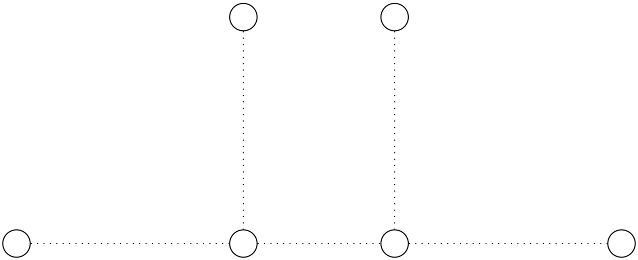


適当に葉を 2 つ選ぶ

証明

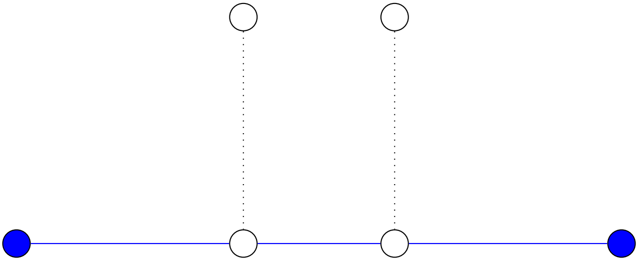
他に葉がないならこれを選べば良い

証明



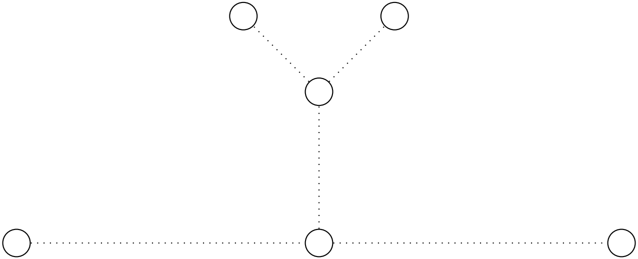
他に葉が 2 つ以上あるとき: パターン 1

証明



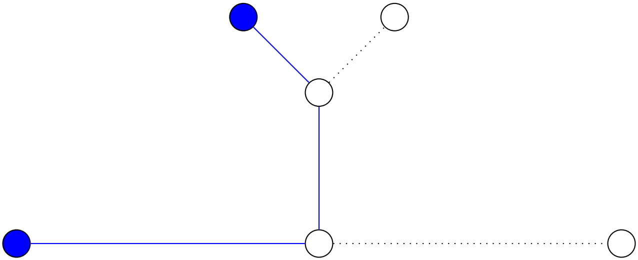
もとの 2 つを選べば良い

証明



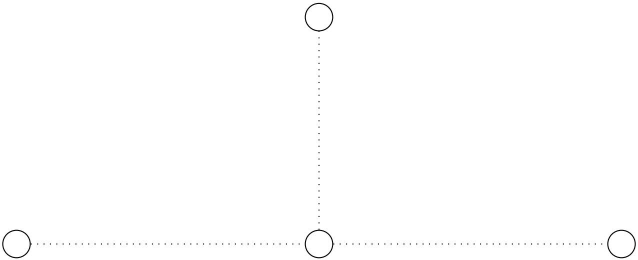
他に葉が 2 つ以上あるとき: パターン 2

証明



この 2 つを選べば良い

証明



葉の数が 3 つの場合はどう選んでも 1 つだけ減る

サブタスク 2

- 各辺について生きてるか死んでるか $O(N)$ で求める $O(N^2)$
- 死んだ辺を潰した木の次数 1 の頂点の数を数える $O(N)$
- 全体で $O(N^2)$: $N \leq 3000$ なら解ける

サブタスク 3

- K が小さい

サブタスク 3

- K が小さい
- 各州について,「それによって死ぬ辺 i.e. その両側にその州の頂点がある辺」を考える
- これはその州の頂点をすべて含む最小の部分木内の辺
- 各州について，これを $O(N)$ で求められればよい

サブタスク 3

- K が小さい
- 各州について,「それによって死ぬ辺 i.e. その両側にその州の頂点がある辺」を考える
- これはその州の頂点をすべて含む最小の部分木内の辺
- 各州について，これを $O(N)$ で求められればよい
- 適当に一頂点選んで DFS すればよい

もう少し考える

- 各州について，それによって死ぬ辺の集合は，その州に属する 2 頂点を結ぶパスすべての Union

もう少し考える

- 各州について，それによって死ぬ辺の集合は，その州に属する 2 頂点を結ぶパスすべての Union
- 3 頂点 v_1, v_2, v_3 について，v_1 と v_3 の間のパスは v_1 と v_2 の間のパスと v_2 と v_3 の間のパスの Union に含まれる

もう少し考える

- 各州について，それによって死ぬ辺の集合は，その州に属する 2 頂点を結ぶパスすべての Union
- 3 頂点 v_1, v_2, v_3 について，v_1 と v_3 の間のパスは v_1 と v_2 の間のパスと v_2 と v_3 の間のパスの Union に含まれる
- 州に含まれる頂点が $v_1, v_2, \ldots, v_n$ なら，v_1 と v_2 の間のパス，v_2 と v_3 の間のパス，$\cdots$ v_{n-1} と v_n の間のパスの Union のみ考えれば良い

サブタスク 4

- パスの長さが小さい→パスに含まれる辺を列挙して良い

サブタスク 4

- パスの長さが小さい→パスに含まれる辺を列挙して良い
- 適当に根付き木にして考えてみる
- 2 頂点の間のパスを求めるには，それらの LCA (最近共通祖先，lowerst common ancestor) まで木を上に登れば良い考えれば良い

満点解法

- パスの長さが長いかもしれない→パスに含まれる辺を列挙できない

満点解法

- パスの長さが長いかもしれない→パスに含まれる辺を列挙できない
- LCA 自体は $O(\log N)$ で求められる (Euler tour + Segment tree, Doubling など)
- パス上の辺すべてを死なせたい

満点解法

- パスの長さが長いかもしれない→パスに含まれる辺を列挙できない
- LCA 自体は $O(\log N)$ で求められる (Euler tour + Segment tree, Doubling など)
- パス上の辺すべてを死なせたい
- 累積和をつかう

満点解法

- パスの長さが長いかもしれない→パスに含まれる辺を列挙できない
- LCA 自体は $O(\log N)$ で求められる (Euler tour + Segment tree, Doubling など)
- パス上の辺すべてを死なせたい
- 累積和をつかう
- 2 頂点に $+1$, LCA の頂点に -2 を加算して，あとで下から累積和を取る
- 各辺について，それらを殺す州の数が定まる

別解

- 死ぬことがわかった時点で辺を縮約していく
- LCA を求めるとき，通った辺はすべて縮約されることに注意

別解

- 死ぬことがわかった時点で辺を縮約していく
- LCA を求めるとき，通った辺はすべて縮約されることに注意
- Union Find を使って，パスの辺をすべて縮約する
- 一度縮約した辺は次回以降スキップされるので，$N - 1$ 回しか操作を行わない

別解

- 死ぬことがわかった時点で辺を縮約していく
- LCA を求めるとき，通った辺はすべて縮約されることに注意
- Union Find を使って，パスの辺をすべて縮約する
- 一度縮約した辺は次回以降スキップされるので，$N-1$ 回しか操作を行わない
- 全体で $O(N\alpha(N))$ の計算量で解ける (α は逆アッカーマン関数)

得点分布