

KUPC2020 spring F: ボタンの木 解説

writer: drafear

2020 年 3 月 20 日

まず、 a, b については差分のみ考えれば良いので目標変化量 d を $d_i = b_i - a_i$ とします。

次に、辺 (i, j) を合計して移動する美しさ c_{ij} を考えます*¹。辺 (i, j) を取り除いたときに分かれる 2 つの木に含まれる頂点の集合をそれぞれ I, J ($i \in I, j \in J$) とすると

$$c_{ij} = - \sum_{v \in I} d_v = \sum_{v \in J} d_v$$

が成り立つので、 $\sum_{v \in I} d_v$ または $-\sum_{v \in J} d_v$ を計算することで、すべての辺 (i, j) について c_{ij} を求めることができます。これは、入力で与えられた木を頂点 1 を根とした根付き木として考えると、頂点 i の部分木に含まれる頂点集合と I 、または頂点 j の部分木に含まれる頂点集合と J が一致するので、深さ優先探索により各頂点 i についてその部分木に含まれる d_v の総和を計算することにより、線形時間で求められます。

今度は各頂点 i のボタンを押す回数 x_i を考えます。すると

$$x_i - x_j = c_{ij}$$

が成り立ちます。したがって、 $i = 1, 2, \dots, N$ について頂点 i のボタンを押す回数 x_i は、頂点 1 のボタンを押す回数 x_1 とある定数 k_i を用いて $x_i = x_1 + k_i$ の形で表せます。ここで、 k_1, \dots, k_N は頂点 1 から深さ優先探索で求めることができます。 k_1, \dots, k_N が求まると、 x_1, \dots, x_N が解となる条件は $x_1, \dots, x_N \geq 0$ なので、 $x_1 = -\min_i \{k_i\}$ とすれば $\sum_i x_i = Nx_1 + \sum_i k_i$ は条件を満たす中で最小になります。

*¹ c_{ij} が負のときは j から i に $-c_{ij}$ 移動すると考えます。つまり $c_{ij} = -c_{ji}$ が成り立ちます。