

会津合宿 2018 Day3 F 問題

01 文字列と窓 (Binary String with Slit)

原案: tsutaj

問題文: TAB

解答: tsutaj · rsk0315 · tsukasa_diary

解説: tsutaj

2018 年 9 月 21 日

01 文字列と窓 (Binary String with Slit)

- 01 文字列 S に対する操作 1 回分を以下で定義
 - S を 2 進数として見た時の LSB を含むように幅 2 のスリットを置く
 - スリット内の数値を +1 するか -1 する
 - ただし、 $1 \leq$ 変更後のスリット内の数値 ≤ 3
- 01 文字列 S_i を T_i に変えるための操作回数の最小値を求める
- このようなクエリが Q 回来るので処理する
- 制約
 - $1 \leq Q \leq 10^5$
 - $2 \leq |S_i| = |T_i| \leq 50$
 - S_i, T_i は 0 と 1 のみからなる
 - S_i, T_i はともに 1 を少なくとも 1 つ含む

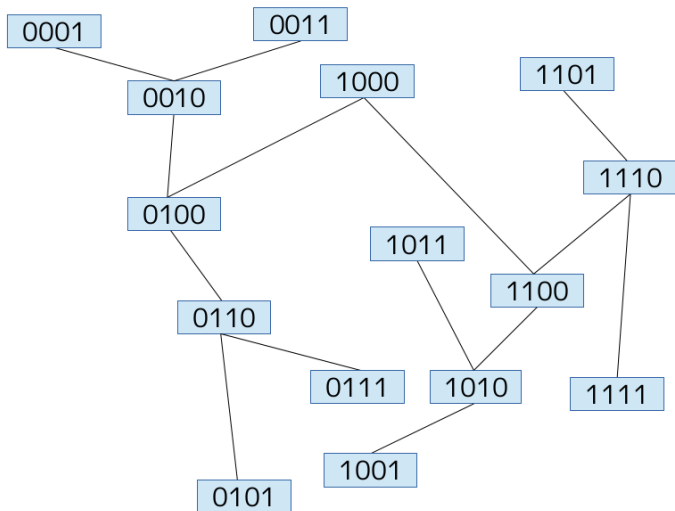
以下、 N は文字列の長さとする

- BFS などでも毎回求める
 - クエリ 1 回につきオーダーが $O(2^N)$ だし、それを 10^5 回やることが要求されているし、まず無理
- 全点对最短経路問題
 - グラフの頂点数を M とすると $O(M^3)$ かかり、この問題において頂点は 2^N 個存在するのだから、この問題においては $O(2^{N^3})$ かかる。まず無理

- この問題は気づけば一瞬です
- スリット内の数値の Before / After を考えよう
- パターンはかなり少ない (たった 4 通り)
 - ① "10" → "01"
 - ② "10" → "11"
 - ③ "11" → "10"
 - ④ "01" → "10"
- 各 01 文字列を頂点と見なし、1 回の操作で到達できるもの同士に辺を張ってグラフを作ってみよう

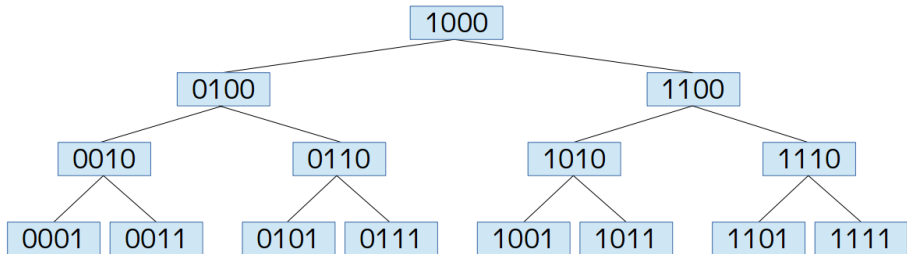
アプローチ

こんな感じになるが・・・もう少し綺麗に書いてみよう



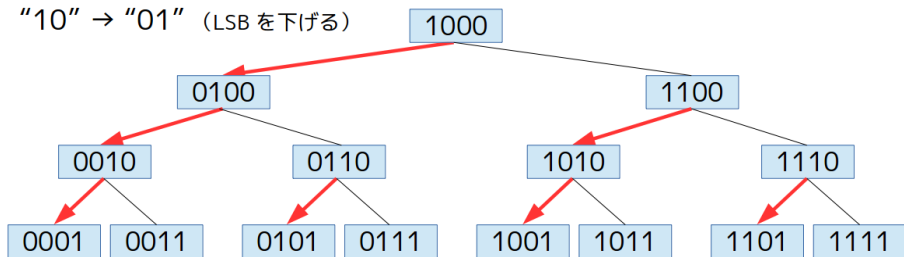
アプローチ

整形すると、このように**完全二分木** (それぞれの辺は相互に移動可能) になる



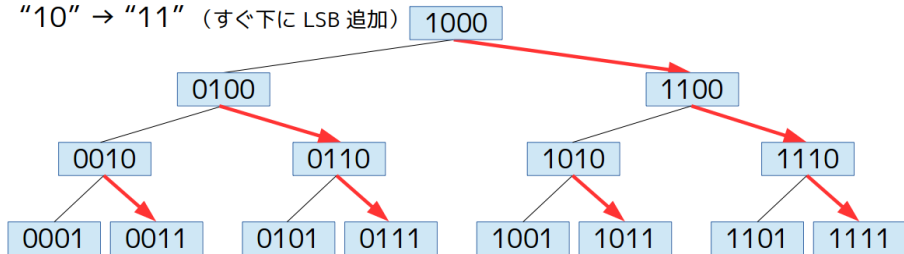
遷移 (1)

“10” → “01” (LSB を下げる)



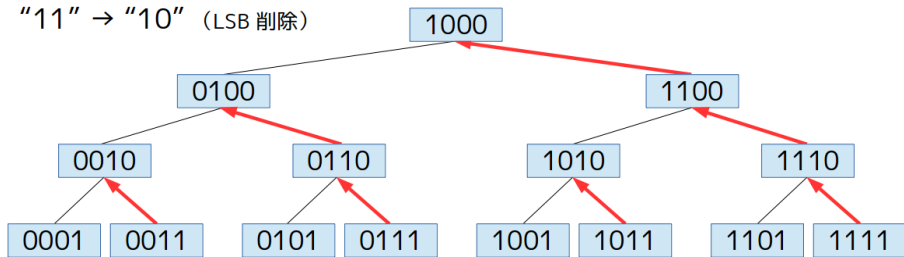
遷移 (2)

“10” → “11” (すぐ下に LSB 追加)



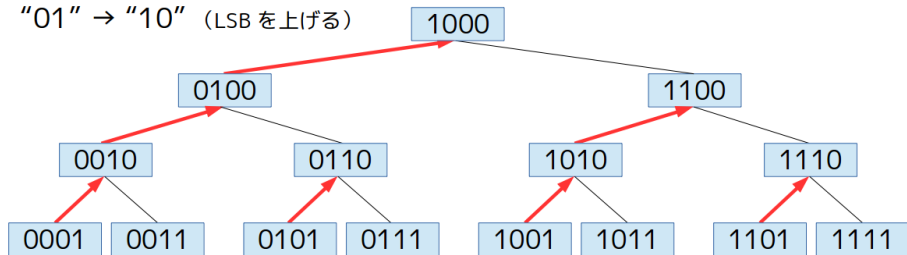
遷移 (3)

“11” → “10” (LSB 削除)



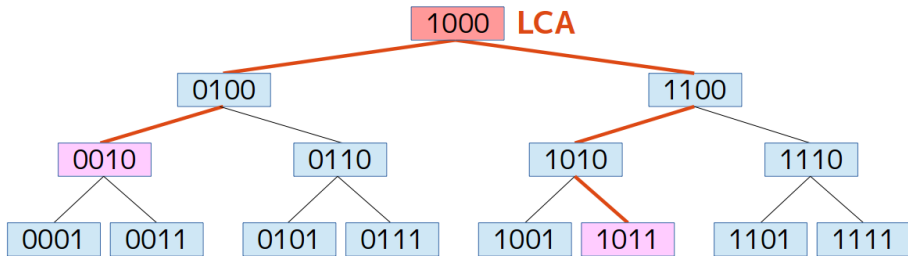
遷移 (4)

“01” → “10” (LSB を上げる)



遷移

- クエリに答えるときは、以下のように S_i と T_i の LCA を求め、パスの長さを求めれば良い
- 木が大きすぎてダブリングで LCA を求めることはできないが、木が平衡なので高さは高々 50 程度
- つまりパスの長さは高々 100 程度なので愚直に求めても間に合う
- $Q = 10^5$ でもこれなら余裕



- Writer 解
 - tsutaj (C++ · 43 行 · 954 bytes)
 - rsk0315 (C++ · 29 行 · 513 bytes)
 - tsukasa_diary (C++ · 37 行 · 687 bytes)
- 統計
 - AC / tried: 14 / 36 (38.9 %)
 - First AC
 - On-site: acpc_Fixstars (69 min)
 - On-line: LLma (93 min)