

I: Tree-Light

原案・解説: 胡桃沢 (@ukuku09)

問題概要

- ・ 頂点数 n の木が与えられる
- ・ 各頂点は $0\sim 9$ のいずれかの数字を一つだけもつ
- ・ 以下の2種類のクエリを合計 q 個処理せよ
 - ・ $\text{count}(r,x,y)$: $T(r)$ にある $x\sim y$ の個数の和を求める
 - ・ $\text{change}(r,x,y)$: $T(r)$ にある x を全て y にする

* $T(r)$ … 頂点 r を根とした部分木

解説

セグ木で殴る。 (以上)

解説

- ・ 最終的にはセグ木で殴るので、数列で考える
 - ・ 数列の長さは n
 - ・ 各要素は $0\sim 9$ のいずれかの数字を一つだけもつ
 - ・ $\text{count}(l,r,x,y)$: $[l,r)$ にある $x\sim y$ の個数の和を求める
 - ・ $\text{change}(l,r,x,y)$: $[l,r)$ にある x を全て y にする

解説

- ・ `count(l,r,x,y)` クエリ
 - ・ 分割した区間ごとに0~9のそれぞれの個数がわかれば良い
 - ・ 各区間に要素数10の配列`A[10]`をもたせる
 - ・ `A[i]`にはその区間にある`i`の個数を格納
 - ・ その区間の`x~y`の個数の和は`A[x]+...+A[y]`

解説

- $\text{change}(l,r,x,y)$ クエリ
 - 各区間のAの再計算は $A[y] += A[x], A[x] = 0$ ($x \neq y$)
 - $[l,r)$ に入るの全ての区間についてやったら遅い
 - $O(\log n)$ くらいでやりたい
 - 遅延評価する (自明)
 - じゃあどうやって変更を子に伝搬させる?

解説

- `change(l,r,x,y)`クエリ
 - 遅延評価にもやっぱり要素数10の配列`B[10]`を使う
 - $B[i] := i$ の変更後の数字
 - $B[i]=x$ であるような全ての $B[i]$ の中身を y にする
 - A の更新は $A'[B[i]] += A[i]$ (A' : 更新後の A)
 - B の変化した部分をうまく子に伝搬

解説

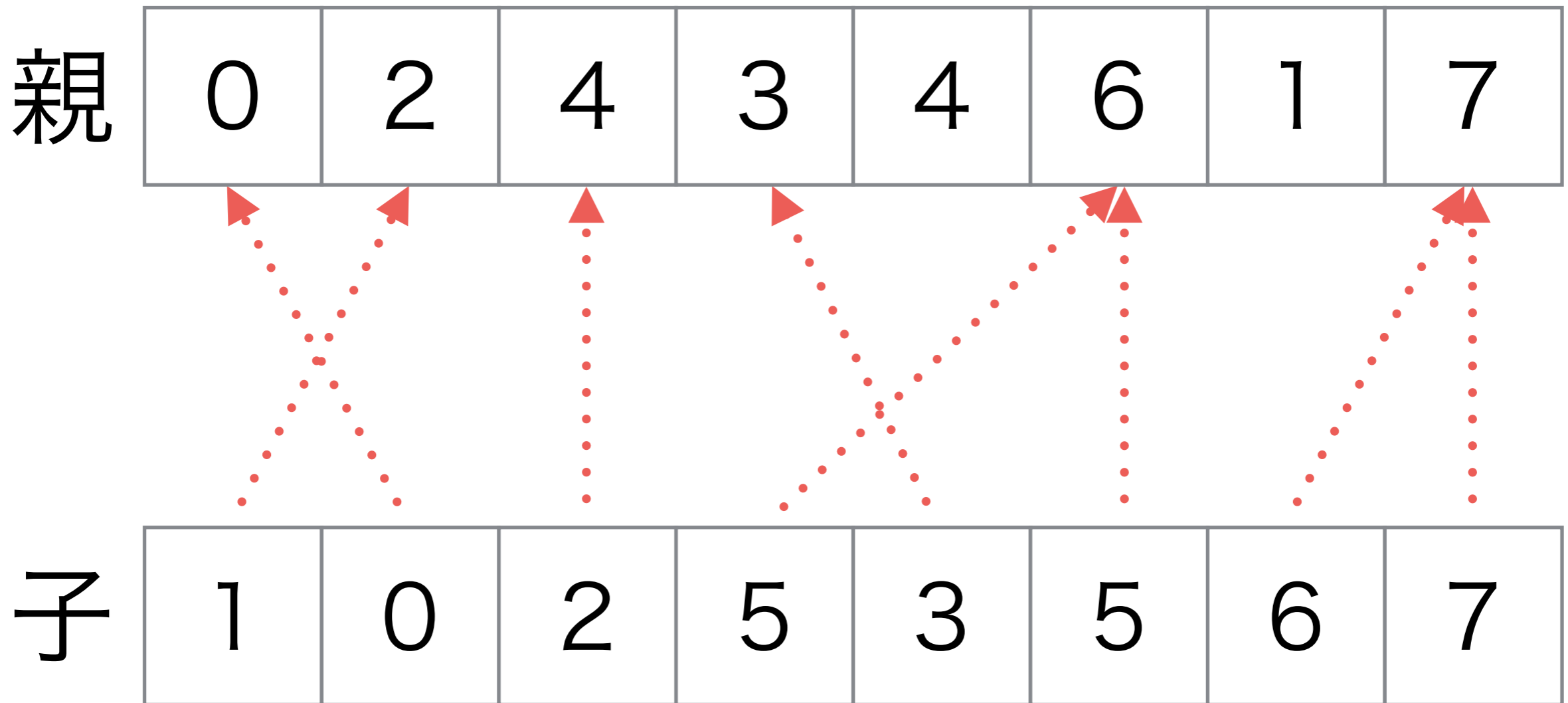
- ・ 伝搬の例

親	0	2	4	3	4	6	1	7
---	---	---	---	---	---	---	---	---

子	1	0	2	5	3	5	6	7
---	---	---	---	---	---	---	---	---

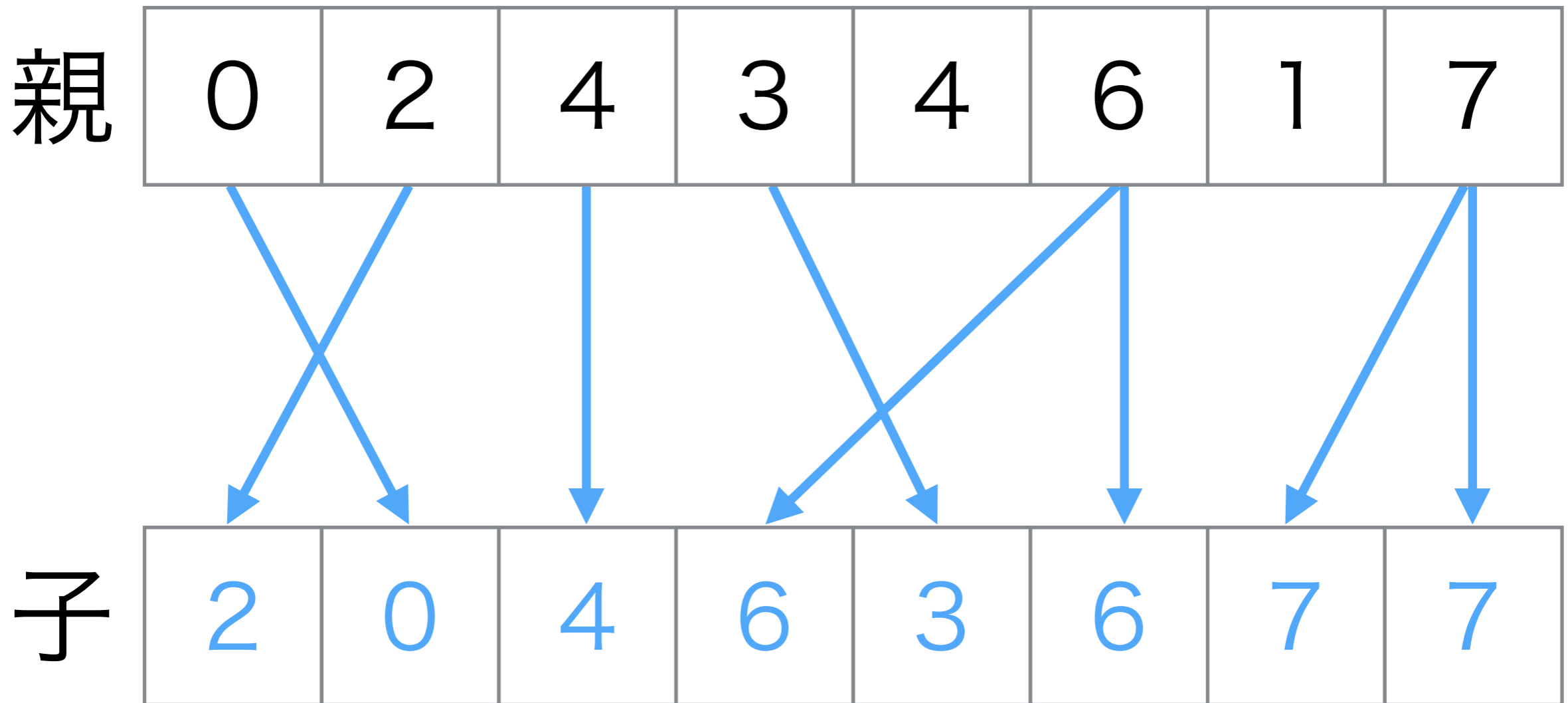
解説

- 子のBの値をインデックスとして親のBを参照



解説

- 親のBから子へ値を伝搬



- 伝搬後、親のBは数列 $0, 1, \dots, N-1$ に書き換えておく

解説

- ・ ここまで数列で考えてきたが、木の場合は？
 - 木を数列に落とし込むテクニック = Euler-Tour
- ・ 根からのDFSでの訪問順に頂点を並べる
- ・ 今回は最初の訪問の順序と部分木の大きさがわかればOK
- ・ $\text{count}(r,x,y) \rightarrow \text{count}(\text{ord}[r], \text{ord}[r] + \text{subsz}[r], x, y)$
- ・ $\text{change}(r,x,y)$ も同様

まとめ

- ・ 与えられた木からDFSの訪問順序で数列をつくる
- ・ 上で得た数列とSegment Treeを対応付ける
- ・ 遅延評価で範囲更新をしながら答えを求める
- ・ 時間計算量: $O(n + q \log n)$
- 空間計算量に注意！

ジヤツジ解

- dohatsu (C++) : 148 lines
- haji (C++) : 101 lines
- kawabys (C++) : 86 lines
- uku (C++) : 122 lines

結果

- First Submission
 - Online: pekempey 41 min
- First Accepted
 - Online: pekempey 41 min
- Success Rate: 25.00 % (4/16)