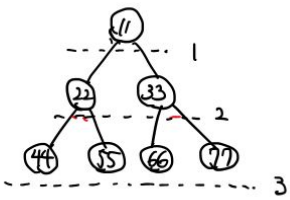
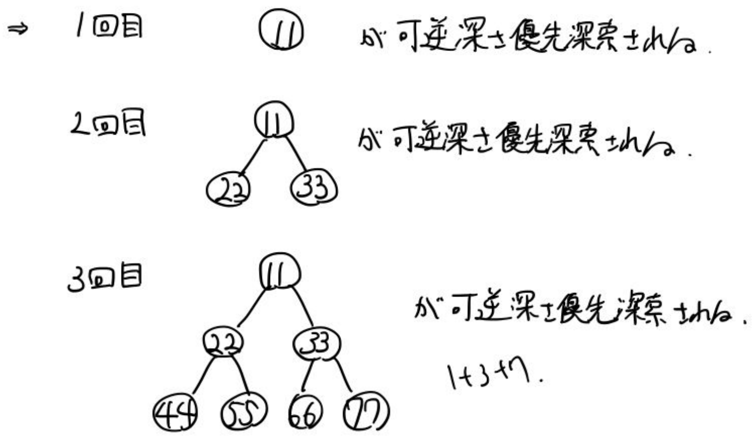
進捗管理・報告(2020/10/13)

1. 現在取り組んでいること
　可逆コンピューティングの研究

2. 進捗状況
・可逆コンピューティングについてと、この分野で行われてきたことの調査

3. 前回からの進捗
・**提案した幅優先探索**の計算量を見積もった
一最良の場合：完全2分木の場合
例)





ここで、可逆深さ優先探索の計算量はO(n)であるので、
例の計算量は 1+3+7

以上を一般化すると、

完全2分木の場合．木の要素の数nに対して、深さは$\lfloor \lg n \rfloor$なので、

$$\sum_{i=1}^{\lfloor \lg n \rfloor +1} \left(\sum_{j=1}^{i} 2^{j-1} \right) \leq O(n)$$

計算の詳細

$$\sum_{i=1}^{\lfloor \lg n \rfloor +1} \left(\sum_{j=1}^{i} 2^{j-1} \right)$$

$$= \sum_{i=1}^{\lfloor \lg n \rfloor +1} (2^i - 1)$$

$$= \sum_{i=1}^{\lfloor \lg n \rfloor +1} 2^i - (\lfloor \lg n \rfloor + 1)$$

$$= \left(\sum_{i=1}^{\lfloor \lg n \rfloor +1} 2 \cdot 2^{i-1} \right) - (\lfloor \lg n \rfloor + 1)$$

$$= (2^{\lfloor \lg n \rfloor + 2} - 2) - (\lfloor \lg n \rfloor + 1)$$

$$= 2^{\lfloor \lg n \rfloor + 2} - \lfloor \lg n \rfloor - 3$$

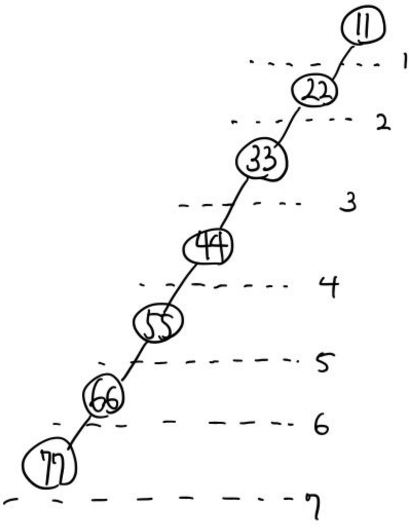
※ここで、$\lfloor \lg n \rfloor + 2 \leq \lg n + 2 = \lg 4n$.

したがって、

$$2^{\lfloor \lg n \rfloor + 2} - \lfloor \lg n \rfloor - 3 \leq 2^{\lg 4n} - \lfloor \lg n \rfloor - 3$$
$$= 4n - \lfloor \lg n \rfloor - 3$$
$$= O(n)$$

一最悪の場合←前回考慮されていなかった

・n個の要素が直線上に並んだ時？



この場合、深さ優先探索はn回行われ、

1回の深さ優先探索で探索される要素数は、

$1 \to 2 \to 3 \to \cdots \to n$、と変化する。

したがって、計算量は

$$1+2+3+\cdots+n = \sum_{i=1}^{n} i = \frac{1}{2} n(n+1)$$
$$= O(n^2)$$

最悪の場合、$O(n^2)$になってしまいそう….。

4. 今後の課題

・提案したアルゴリズムがO(n^2)であることがわかった。

・平均の場合もO(n^2)なのだろうか？