Datové struktury – Úvod

Navrhněte co nejjednodušší datovou strukturu, která podporuje následující operace:

1. INSERT a DELETE v $\mathcal{O}(n)$,
   SEARCH v $\mathcal{O}(\log n)$;

Datové struktury – Úvod

Navrhněte co nejjednodušší datovou strukturu, která podporuje následující operace:

1. Insert a Delete v $\mathcal{O}(n)$,
   Search v $\mathcal{O}(\log n)$;

2. Insert a Delete v $\mathcal{O}(n)$,
   Predecessor a Successor v $\mathcal{O}(1)$;

Datové struktury – Úvod

Navrhněte co nejjednodušší datovou strukturu, která podporuje následující operace:

1. Insert a Delete v $\mathcal{O}(n)$,
   Search v $\mathcal{O}(\log n)$;

2. Insert a Delete v $\mathcal{O}(n)$,
   Predecessor a Successor v $\mathcal{O}(1)$;

3. Insert a Delete v $\mathcal{O}(1)$,
   Minimum a Maximum(S) v $\mathcal{O}(n)$.

Datové struktury – Příklad 1

Klíč a priorita

Mějme prvky tvaru (k, p), kde k je klíč a p je priorita. Navrhněte datovou strukturu takovou, aby následující tři operace měly složitost $\mathcal{O}(\log n)$, kde n je počet prvků ve struktuře.

- $\textsc{Insert}(k, p)$ – pokud prvek s klíčem k ve struktuře neexistuje, vloží nový prvek do struktury; pokud už prvek s klíčem k existuje, změní jeho prioritu
- $\textsc{Delete}(k)$ – odstraní prvek s klíčem k
- $\textsc{Get}(k)$ – najde prvek s nejmenší prioritou z těch, které mají klíč $\leq k$

Datové struktury – Příklad 2

Navrhněte datovou strukturu vytvořenou ze sekvence n hodnot $x_1, \ldots, x_n$ a která má podporovat následující operaci:

- $\textsc{Min}(i, j)$ – nejmenší hodnota z $x_i, x_{i+1}, \ldots, x_j$.

Jaký nejlepší poměr časové složitosti vytvoření dané struktury a dotazů $\textsc{Min}$ jste schopni dosáhnout?

Datové struktury – Vyváženost stromů

Rozhodněte (a dokažte), zda následující podmínky zaručují vyváženost binárních stromů, tj. zda mají všechny binární stromy splňující danou podmínku výšku $\mathcal{O}(\log n)$, kde n je počet uzlů stromu.

1. Každý uzel je buď list nebo má právě dva potomky.

Datové struktury – Vyváženost stromů

Rozhodněte (a dokažte), zda následující podmínky zaručují vyváženost binárních stromů, tj. zda mají všechny binární stromy splňující danou podmínku výšku $\mathcal{O}(\log n)$, kde n je počet uzlů stromu.

1. Každý uzel je buď list nebo má právě dva potomky.
2. Velikost každého podstromu je právě $2^k - 1$, kde $k \in \mathbb{N}$. (k nemusí být stejné pro všechny podstromy.)

Datové struktury – Vyváženost stromů

Rozhodněte (a dokažte), zda následující podmínky zaručují vyváženost binárních stromů, tj. zda mají všechny binární stromy splňující danou podmínku výšku $\mathcal{O}(\log n)$, kde n je počet uzlů stromu.

1. Každý uzel je buď list nebo má právě dva potomky.
2. Velikost každého podstromu je právě $2^k - 1$, kde $k \in \mathbb{N}$. (k nemusí být stejné pro všechny podstromy.)
3. Existuje $0 < c < 1$ takové, že pro každý uzel platí, že velikost jeho menšího podstromu je alespoň $c$násobek velikosti jeho většího podstromu. (Má-li uzel jen jednoho potomka, uvažujeme, že jeho menší podstrom má velikost 0).

Datové struktury – Vyváženost stromů

Rozhodněte (a dokažte), zda následující podmínky zaručují vyváženost binárních stromů, tj. zda mají všechny binární stromy splňující danou podmínku výšku $\mathcal{O}(\log n)$, kde n je počet uzlů stromu.

1. Každý uzel je buď list nebo má právě dva potomky.
2. Velikost každého podstromu je právě $2^k - 1$, kde $k \in \mathbb{N}$. (k nemusí být stejné pro všechny podstromy.)
3. Existuje $0 < c < 1$ takové, že pro každý uzel platí, že velikost jeho menšího podstromu je alespoň $c$násobek velikosti jeho většího podstromu. (Má-li uzel jen jednoho potomka, uvažujeme, že jeho menší podstrom má velikost 0).
4. Existuje c takové, že pro každý uzel platí, že výška podstromů jeho dvou potomků se liší maximálně o c.

Datové struktury – Vyváženost stromů

Rozhodněte (a dokažte), zda následující podmínky zaručují vyváženost binárních stromů, tj. zda mají všechny binární stromy splňující danou podmínku výšku $\mathcal{O}(\log n)$, kde n je počet uzlů stromu.

1. Každý uzel je buď list nebo má právě dva potomky.
2. Velikost každého podstromu je právě $2^k - 1$, kde $k \in \mathbb{N}$. (k nemusí být stejné pro všechny podstromy.)
3. Existuje $0 < c < 1$ takové, že pro každý uzel platí, že velikost jeho menšího podstromu je alespoň $c$násobek velikosti jeho většího podstromu. (Má-li uzel jen jednoho potomka, uvažujeme, že jeho menší podstrom má velikost 0).
4. Existuje c takové, že pro každý uzel platí, že výška podstromů jeho dvou potomků se liší maximálně o c.
5. Průměrná hloubka uzlu je $\mathcal{O}(\log n)$. (Hloubka uzlu je vzdálenost od kořene.)