

## 9. CVIČENÍ Z ADS 1, ČTVRTEK 15:40, LS '24

Vyhledávací stromy: AVL a  $(a, b)$ -stromy

### Trie:

1. *Třídění pomocí trie.* Vymyslete, jak pomocí písmenkového stromu setřídít posloupnost řetězců v čase lineárním vzhledem k součtu jejich délek. Porovnejte s algoritmem přihrádkového třídění. (Uvažujte konstantně velkou abecedu.)
2. *Rýmy.* Navrhněte datovou strukturu pro básníky, která si bude pamatovat slovník a bude umět hledat rýmy. Tedy pro libovolné zadané slovo najde jiné slovo ve slovníku, které má se zadaným co nejdelší společný suffix (koncovku).
3. *Přesmyčky.* Jak reprezentovat slovník (množinu slov), abyste uměli rychle vyhledávat všechny přesmyčky zadaného slova?

### LLRB:

4. *hloubka LLRB.* Spočítejte přesně, jaká může být minimální a maximální hloubka LLRB stromu s  $n$  klíči (přesně znamená určit i multiplikativní konstanty; aditivní konstanty určit přesně nemusíte). Porovnejte maximální hloubku s AVL stromem.

### Intervalové stromy:

5. *Nejbližší větší napravo.* Naučte intervalový strom zjistit nejbližší prvek, který leží napravo od zadaného listu a obsahuje větší hodnotu. Jinými slovy: na vstupu dostaneme index  $i$  v posloupnosti a chceme najít nejmenší  $j > i$  takové, že  $x_j > x_i$ .
6. *Intervalový update.* Naučte (součtový) intervalový strom operaci  $\text{update}(i, j, \Delta)$ , která k hodnotám  $x_i, x_{i+1}, \dots, x_j$  reprezentované posloupnosti přičte  $\Delta$ . Jelikož těchto hodnot může být až  $\Omega(n)$  a my chceme logaritmickou časovou složitost, stačí jen upravit strom tak, aby i po provedení této operace vracel správné výsledky pro dotazy na součet intervalu.
7. *Jeřáb.* Jeřáb se skládá z  $n$  ramen spojených klouby. Pro jednoduchost si ho představíme jako lomenou čáru v rovině. První úsečka je fixní, každá další je připojena kloubem na svou předchůdkyni. Koncový bod poslední úsečky hraje roli háku. Navrhněte datovou strukturu, která si bude pamatovat stav jeřábu a bude nabízet operace „otoč  $i$ -tým kloubem o úhel  $\alpha$ “ a „zjisti aktuální pozici háku“.

*Bonusové úlohy:*

8. *Lexikograficky nejmenší rým.* Upravte datovou strukturu na hledání nejlepších rýmů, aby v případě, že nejlepších rýmů je více, vypsala lexikograficky nejmenší z nich.