Sada: 1 Příklad: 2 IV003 Algoritmy a datové struktury II

Jméno: Karel Kubíček UČO: 408351

Jméno: Henrich Lauko UČO: 410438

1. Posloupnost n operací Insert a Min-All má složitost $\mathcal{O}(n)$.
   Neplatí. Můžeme vložit $\frac{n}{2}$ stejných prvků, následně můžeme volat Min-All, která nesmaže žádný prvek, tím pádem můžeme volat tuto operaci $\frac{n}{2}$-krát vždy se složitostí $\mathcal{O}(n)$, tedy $\frac{n}{2} \cdot \mathcal{O}(n) = \mathcal{O}(n^2)$.

2. Posloupnost n operací Insert a Min-one má složitost $\mathcal{O}(n)$.
   Platí. Dokážeme pomocí metody účtů. Operacím přiřadíme následující ceny:

operace	reálná cena (c_i)	amortizovaná cena ($\widehat{c}_i$)
Insert	1	2
Min-One	*délka seznamu* $- 1$	0

   Při Insert vložíme na účet 2 kredity, jeden z nich slouží na zaplacení operace Insert a druhý na zaplacení smazání sám sebe při operaci Min-One. Jelikož v každém momentu bude mít každý prvek jeden kredit, tak účet nemůže jít do mínusu. Tudíž amortizovaná cena může být nanejvýš $2n$ kreditů, kde n je počet operací. Tedy pro n operací je amortizovaná složitost $\mathcal{O}(n)$.

3. Posloupnost n operací Insert a Delete má složitost $\mathcal{O}(n)$.
   Neplatí. Můžeme vložit $\frac{n}{2}$ stejných prvků x, následně můžeme volat Delete(S, x), která nesmaže žádný prvek, tím pádem můžeme volat tuto operaci $\frac{n}{2}$-krát vždy se složitostí $\mathcal{O}(n)$, tedy $\frac{n}{2} \cdot \mathcal{O}(n) = \mathcal{O}(n^2)$.

4. Posloupnost n operací Insert a Delete taková, že při každém volání se operace Delete volá s jiným parametrem i má složitost $\mathcal{O}(n)$.
   Platí. Dokážeme pomocí metody účtů. Přiřadíme následující ceny operacím:

operace	reálná cena (c_i)	amortizovaná cena ($\widehat{c}_i$)
Insert	1	2
Delete	*délka seznamu* $-$ *počet prvků* i	0

   Při Insert vložíme na účet 2 kredity, jeden znich slouží na zaplacení operace Insert a druhý na zaplacení smazání sám sebe při operaci Delete. Jelikož v každém momentu bude mít každý prvek jeden kredit, tak účet nemůže jít do mínusu. Při operaci Delete se smaže jen *délka seznamu* $-$ *počet prvků* i prvků, teda počet prvků i musí zůstat s 1 kreditem. Tudíž amortizovaná cena může být nanejvýš $2n$ kreditů, kde n je počet operací. Teda pro n operací je amortizovaná složitost $\mathcal{O}(n)$.