

1. *Počítání pestrých řetězců:*

Pestřý budeme říkat takovému řetězci, jehož všechny rotace jsou navzájem různé. Kolik existuje pestrých řetězců v Σ^n pro konečnou abecedu Σ a prvočíslo n ?

2. *Nejdelší palindrom jako prefix:*

Jak v řetězci najít nejdelší prefix, který je *palindrom* (popředu a pozpátku se čte stejně)?

3. *Zkratky zrychlují algoritmus:*

Uvažujme zjednodušený algoritmus AC, který nepoužívá zkratkové hrany a vždy projde po zpětných hranách až do kořene. Ukažte vhodnými příklady vstupů, že tento algoritmus je asymptoticky pomalejší.

4. *Množiny:*

Jednoduchý způsob, jak si poradit s hlášením výskytů, je předpočítat si pro každý stav s množinu $M(s)$ slov k ohlášení. Dokažte, že tyto množiny není možné sestojit v lineárním čase s velikostí slovníku, protože součet jejich velikostí může být pro některé vstupy superlineární.

5. *Chytré množiny:*

Rozmyslete si, že množiny $M(s)$ z předchozího příkladu by bylo možné reprezentovat jako srůstající spojové seznamy – tedy takové, kde si každý prvek pamatuje ukazatel na svého následníka, který ovšem může ležet v jiném seznamu. Přesvědčte se, že námi zavedené zkratkové hrany lze interpretovat jako ukazatele ve srůstajících seznamech.

6. *Počet zpětných hran do kořene:*

Dokažte, že do kořene stromu vede aspoň tolik zpětných hran, kolik různých znaků se vyskytuje v jehlách.

7. *Jeden výskyt:*

Co kdybychom chtěli pro každou pozici v seně hlásit jenom jeden výskyt jehly? Mohl by to být třeba ten nejdelší, který na dané pozici končí. Ukažte, jak to zařídit bez vyjmenování všech výskytů. Jak by se situace změnila, kdybychom místo nejdelšího hledali nejkratší?

8. *Fibonacciho slova:*

Definujme *Fibonacciho slova* takto: $F_0 = \mathbf{a}$, $F_1 = \mathbf{b}$, $F_{n+2} = F_n F_{n+1}$. Jak v zadaném řetězci nad abecedou $\{\mathbf{a}, \mathbf{b}\}$ najít nejdelší Fibonacciho podslovo?