

1. *Ekvivalence podle rotací:*

*Rotací* řetězce  $\alpha$  o  $K$  pozic nazýváme řetězec  $\alpha[K : ]\alpha[: K]$ . Jak o dvou řetězcích zjistit, zda je jeden rotací druhého?

2. *Rotace s konstantní pamětí:*

Jak v lineárním čase zrotovat řetězec, dostačuje-li paměť počítače jen na uložení jednoho řetězce a  $\mathcal{O}(1)$  pomocných proměnných?

3. *Testování periodicity:*

Jak zjistit, zda je zadané slovo  $\alpha$  *periodické*? Tím myslíme zda existuje slovo  $\beta$  a číslo  $k > 1$  takové, že  $\alpha = \beta^k$  (zřetězení  $k$  kopií řetězce  $\beta$ ).

4. *Počítání pestrých řetězců:*

*Pestrý* budeme říkat takovému řetězci, jehož všechny rotace jsou navzájem různé. Kolik existuje pestrých řetězců v  $\Sigma^n$  pro konečnou abecedu  $\Sigma$  a prvočíslo  $n$ ?

5. *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?

6. *Nejdelší palindrom jako prefix:*

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

7. *Počet řetězců bez jehly:*

Spočítejte, kolik řetězců délky  $n$  nad abecedou  $\Sigma$  *neobsahuje* zadanou jehlu.

8. \* *Ukládání rotujících řetězců:*

Dva řetězce jsou ekvivalentní, pokud je jeden rotací druhého. Navrhněte datovou strukturu pro uložení množiny řetězců, která umí přidat řetězec do množiny a dále umí otestovat pro zadaný řetězec, zda se již v množině nachází řetězec jemu ekvivalentní.

*Nápověda:* Nejdříve vymyslete algoritmus, který v lineárním čase nalezne tu z rotací zadaného řetězce, jež je lexikograficky minimální.