

*Příklad* (podposloupnost)

Spočtete, kolikrát se daná jehla vyskytne v daném seně (dva řetězce, seno mnohem delší) jako podposloupnost (možnost proložení jinými znaky). Například v **BARBARAR** je **BAR** 9krát.

┌

*Řešení* (Dynamickým programováním)

Pro každou pozici v jehle si budeme pamatovat, kolika způsoby jsme schopni se do ní dostat (v prvních  $n$  písmenech sena). Tyto hodnoty pak pro  $n+1$  písmen sena přepočítáme jednoduše tak, že k hodnotám na místech, kde je  $n+1$ -ní písmeno sena, přičteme hodnotu na předchozí pozici (-1. pozice má vždy hodnotu 1) (musíme přičítat od poslední pozice v jehle). Po průchodu senem už stačí jen vypsát hodnotu na konci jehly.

Tedy máme časovou složitost součin délky jehly a délky sena (pokud si pro každé písmeno uchováme všechny pozice, kde se vyskytuje, tak se to dá srazit na součet délky jehly a součin počtu výskytů nejčastějšího písmene a délky sena).

└