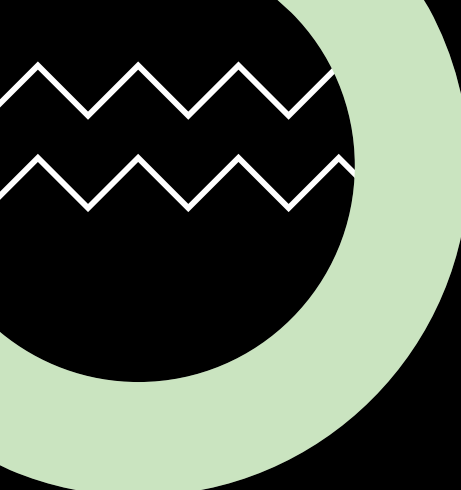


Sekvenční vyhledávání

Ondřej Chovanec





Co je to sekvenční vyhledávání?



Sekvenční vyhledávání (nebo lineární vyhledávání) je **jednoduchý algoritmus** pro hledání prvku v seznamu.



Vhodný pro **neuspořádané nebo malé seznamy**.



Příklad použití: Vyhledávání položky v neřazeném seznamu hodnot



Jak funguje?

- Algoritmus prochází seznam **prvek po prvku**.
- Porovnává **aktuální prvek s hledaným klíčem**.
- **Pokud najde shodu**, proces se Neukončí, pokračuje až do konce seznamu.
- **Pokud shodu nenajde**, pokračuje k dalšímu prvku až do konce seznamu.



```
public static int sekvenčniVyhledavani(int[] pole, int x)
{
    for (int i = 0; i < pole.Length; i++)
    {
        if (pole[i] == x)
            return i; // vraci index, pokud jsme hledany prvek nalezli
    }
    return -1; // pri neuspechu vracime -1
}
```

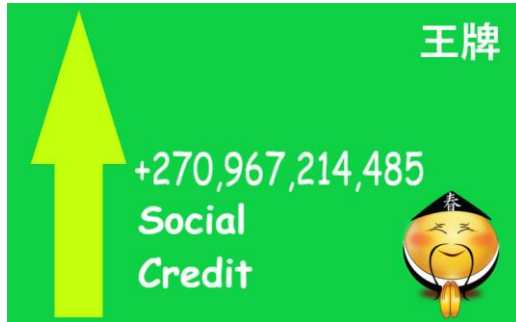
Časová složitost

- $\text{Min} \approx \text{Max} \approx \text{Avg} \approx N$
- Sekvenční vyhledávání není efektivní pro **velké nebo řazené seznamy**.



Výhody a nevýhody

Výhody:



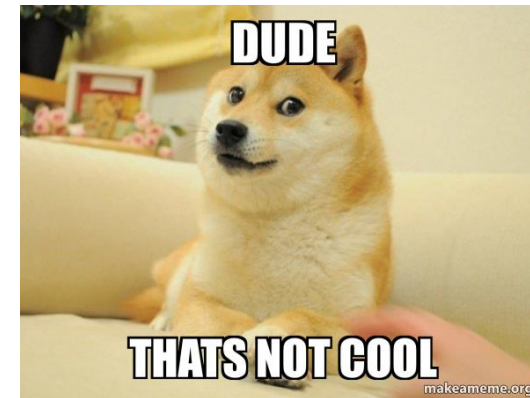
Jednoduchá implementace –
žádné složité podmínky
nebo třídění.



**Vhodné pro neuspořádané
seznamy.**

Nevýhody:

- **Lineární časová složitost** – je nutné procházet celý seznam. (z toho důvodu je pomalejší pro velké seznamy)



Kde se používá?

- V malých databázích, kde není výhodné složitě třídit data.
- Při **vyhledávání konkrétní hodnoty v neuspořádaném seznamu**.
- **Testování nebo prototypování** jednoduchých algoritmů.



Shrnutí

- Sekvenční vyhledávání je **základní vyhledávací algoritmus** procházející každý prvek.
- **Efektivní pro malé seznamy**, ale nevhodné pro velká data.
- Důležité rozumět jeho principům, protože tvoří základ pro složitější algoritmy.



Děkuji za pozornost