

# Datové struktury

pole, seznam, záznam, hash, strom, zásobník, fronta

Adamus Colepaticius Trolo

22. 1. 2025

GEVO

# Obsah

Datová struktura obecně .....	1
Co to je datová struktura .....	2
Operace na datové struktuře .....	3
Konkrétní struktury .....	4
Pole (Array) .....	5
Seznam (List) .....	7

## Co to je datová struktura

Datová struktura je zkrátka řád uložení velkého množství souvisejících dat ve vnitřní paměti.

Při jejím návrhu pracujeme téměř výhradně s von Neumannovým modelem počítače a uvažujeme, že vnitřní paměť je **random access**, tedy na danou adresu je možný okamžitý přístup.

Podle řešeného problému volíme datovou strukturu tak, aby **nejčastější operace** trvaly, co nejkratčeji.

## Operace na datové struktuře

Budeme datové struktury hodnotit z hlediska rychlosti provedení následujících operací:

- uložení (tedy i přepsání) hodnoty na danou pozici
- přečtení hodnoty na dané pozici
- přidání hodnoty na jeden z konců
- odebrání hodnoty z jednoho z konců
- přidání hodnoty na libovolnou pozici
- odebrání hodnoty z libovolné pozice
- nalezení prvku s danou hodnotou

# Obsah

Datová struktura obecně .....	1
Co to je datová struktura .....	2
Operace na datové struktuře .....	3
Konkrétní struktury .....	4
Pole (Array) .....	5
Seznam (List) .....	7

## Pole (Array)

Pole je jednoduchá datová struktura **s danou délkou**, jež ukládá hodnoty do vnitřní paměti bezprostředně za sebou. Přístup k hodnotám probíhá přes **indexy** – vlastně počet míst v paměti od začátku pole.

		ADRESY V PAMĚTI									
		776	777	779	780	781	782	783	784	785	786
HODNOTY		8	25	43	62	42	63	41	18	42	17
	INDEXY V POLI VELIKOSTI 10	0	1	2	3	4	5	6	7	8	9

# Pole (Array)

## Náročnost operací

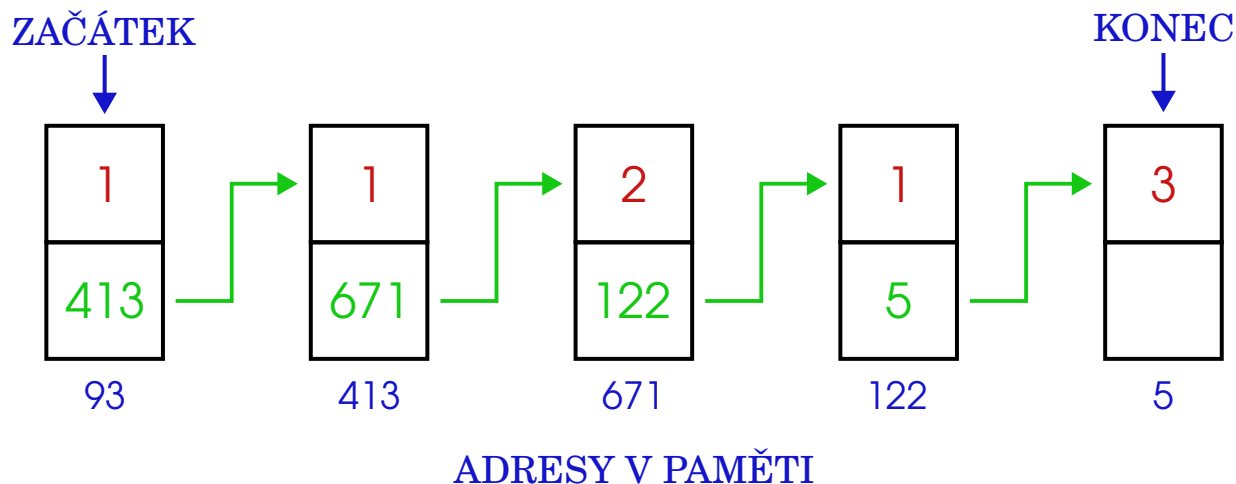
- uložení i přepis hodnoty podle indexu: **instantní**
  - Zkrátka zapíšu hodnotu do RAM na **začátek pole + index**.
- odebrání hodnoty ze zadního konce: **instantní**
  - Zmenším velikost pole o 1.
- odebrání hodnoty na jiné pozici: **úměrné délce pole**
  - Po odebrání je potřeba přesunout všechny hodnoty s vyšším indexem o jeden index doleva.
- přidání prvku kamkoliv: **úměrné délce pole**
  - Může se stát, že pole kolem sebe nemá v paměti místo, takže je potřeba je **překopírovat jinam**.
- nalezení konkrétní hodnoty: **úměrné délce pole**
  - Prostě musím celé pole projít hodnotu po hodnotě.

## Seznam (List)

Datová struktura složená z **uzlů**. Každý uzel obsahuje dvě data:

- **hodnotu**,
- **adresu v paměti** s následujícím uzlem.

Uzly nemusejí být v paměti seřazeny za sebou.





## Seznam (List)

### Náročnost operací

- uložení i přepis hodnoty podle indexu: **úměrné délce seznamu**
  - Musím procházet seznam od **začátku**, dokud se nedostanu na danou pozici.
- přidání na jeden z konců / odebrání z jednoho z konců: **instantní**
  - Stačí připojit další uzel a upravit ten předchozí / následující.
- přidání / odebrání podle indexu: **úměrné délce seznamu**
  - Samotný proces přidání / odebrání je instantní, ale musím se na daný index nejprve dostat.
- nalezení konkrétní hodnoty: **úměrné délce seznamu**
  - Musím seznam procházet od začátku, dokud hodnotu nenajdu.