

Hashovací tabulka

Vladyslav Kovalets

Vysoké učenie technické v Brne
Fakulta informačných technológií

2. 5. 2021

1 Co to je

- Definice
- Jak to vypadá

2 Jak to funguje

- Princip vyhledávání
- Princip fungování
- Asymptotická složitost

3 Implementace

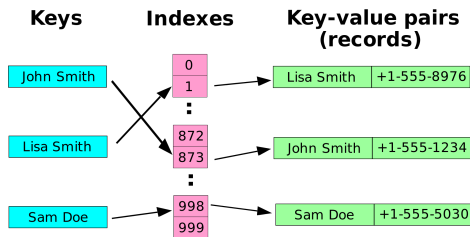
- Pomocí pole

4 Výhody a nevýhody

- **Hashovací tabulka** je datová struktura pro ukládání dvojic (klíč, hodnota) nabízející dobrý kompromis mezi rychlostí vyhledávání a paměťovou náročností.

- **Hashovací tabulka** je datová struktura pro ukládání dvojic (klíč, hodnota) nabízející dobrý kompromis mezi rychlostí vyhledávání a paměťovou náročností.
- Obecně řečeno, hashovací tabulka je datová struktura se schopností efektivně vkládat, mazat a hledat datové záznamy podle klíče.

Jak to vypadá



Obrázek: Malý telefonní seznam jako hashovací tabulka

Hashovací tabulka uvnitř obsahuje pole tzv. **slotů**, do kterých lze ukládat záznamy. Hashovací tabulka v podstatě dělá jen to, že na základě klíče vybere vhodný slot a operaci provede v něm. Abychom docílili efektivity, snažíme se, aby všechny sloty byly využity rovnoměrně, tedy aby různé klíče ideálně padaly do různých slotů. Samozřejmě to není zcela možné, protože množina klíčů je mnohem větší než počet slotů. Takové situaci říkáme kolize a existují různé metody, jak tyto kolize řešit:

Hashovací tabulka uvnitř obsahuje pole tzv. **slotů**, do kterých lze ukládat záznamy. Hashovací tabulka v podstatě dělá jen to, že na základě klíče vybere vhodný slot a operaci provede v něm. Abychom docílili efektivity, snažíme se, aby všechny sloty byly využity rovnoměrně, tedy aby různé klíče ideálně padaly do různých slotů. Samozřejmě to není zcela možné, protože množina klíčů je mnohem větší než počet slotů. Takové situaci říkáme kolize a existují různé metody, jak tyto kolize řešit:

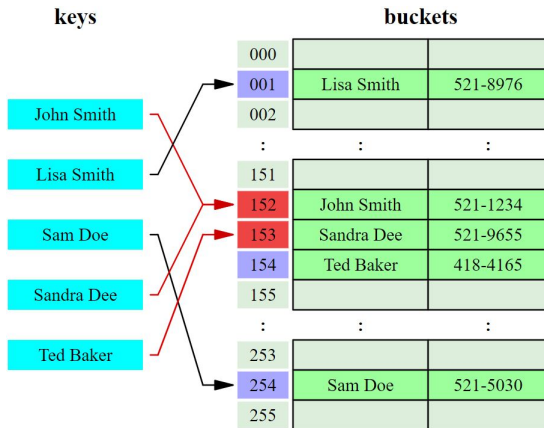
- **zřetězení záznamů** - každý slot obsahuje spojový seznam, do kterého se postupně řetězí prvky patřící do stejného slotu.

Hashovací tabulka uvnitř obsahuje pole tzv. **slotů**, do kterých lze ukládat záznamy. Hashovací tabulka v podstatě dělá jen to, že na základě klíče vybere vhodný slot a operaci provede v něm. Abychom docílili efektivity, snažíme se, aby všechny sloty byly využity rovnoměrně, tedy aby různé klíče ideálně padaly do různých slotů. Samozřejmě to není zcela možné, protože množina klíčů je mnohem větší než počet slotů. Takové situaci říkáme kolize a existují různé metody, jak tyto kolize řešit:

- **zřetězení záznamů** - každý slot obsahuje spojový seznam, do kterého se postupně řetězí prvky patřící do stejného slotu.
- **otevřená adresace** - obsah všech slotů je umístěn v jednom poli a tak mohou data z jednoho slotu "přetékat" i do jiných slotů a tím zabírat volné místo pro jejich budoucí prvky, což se minimalizuje různými dalšími technikami (linear probing, double hashing...)

Asymptotická složitost

Operace	Typický případ	Nejhorší případ
<i>vyhledávání podle klíče</i>	$O(1)$	$O(n)$
<i>vkládání záznamu</i>	$O(1)$	$O(n)$
<i>mazání záznamu</i>	$O(1)$	$O(n)$



Obrázek: hashovací tabulka realizovaná jedním polem

C++ implementace

```
class HashEntry {  
private:  
    int key;  
    int value;  
public:  
    HashEntry(int key, int value) {  
        this->key = key;  
        this->value = value;  
    }  
    int getKey() {  
        return key;  
    }  
    int getValue() {  
        return value;  
    }  
};
```

Výhody a nevýhody

Výhody

- hašování je v porovnání s jinými vyhledávacími metodami, např. vyhledávacími stromy (nejen binárními), v průměrném případě asymptoticky rychlejší.

Nevýhody

Výhody a nevýhody

Výhody

- hašování je v porovnání s jinými vyhledávacími metodami, např. vyhledávacími stromy (nejen binárními), v průměrném případě asymptoticky rychlejší.

Nevýhody

- neposkytuje záruky pro nejhorší případ.

Výhody

- hašování je v porovnání s jinými vyhledávacími metodami, např. vyhledávacími stromy (nejen binárními), v průměrném případě asymptoticky rychlejší.

Nevýhody

- neposkytuje záruky pro nejhorší případ.
- potřebuje úplný klíč, tj. neumožňuje dobré vyhledávání pro částečně známý klíč pro strukturované klíče nebo klíče ve vícerozměrném prostoru.

Výhody

- hašování je v porovnání s jinými vyhledávacími metodami, např. vyhledávacími stromy (nejen binárními), v průměrném případě asymptoticky rychlejší.

Nevýhody

- neposkytuje záruky pro nejhorší případ.
- potřebuje úplný klíč, tj. neumožňuje dobré vyhledávání pro částečně známý klíč pro strukturované klíče nebo klíče ve vícerozměrném prostoru.
- neposkytuje intervalové dotazy, tj. vyhledání všech klíčů v daném intervalu.



Voho.eu: *Hashovací tabulka*. [online], [citováno 2. 5. 2021].

Dostupné z:

<http://voho.eu/wiki/datova-struktura-hash-tabulka/>



Wikipedia.org: *Hašovací tabulka*. [online], [citováno 2. 5. 2021].

Dostupné z: https://cs.wikipedia.org/wiki/Hašovací_tabulka