

8. cvičení

Datové struktury I, 27. 11. 2025

<https://iuuk.mff.cuni.cz/~chmel/2526/ds1/>

Úloha 1 (Špatná verze kukačky)

Proč je následující implementace insertu pro kukačkové hashování problematická? (Implementaci a podmínky pro rehashování pro tento příklad meteme pod koberec.)

```
for i=1 to n
  if T[h1(x)] je prázdné
    T[h1(x)] = x
  return
swap(T[h1(x)], x)
if T[h2(x)] je prázdné
  T[h2(x)] = x
return
swap(T[h2(x)], x)
```

Úloha 2 (Rehashujeme)

Jednoduchá implementace rehashe u kukačkového hashování je, že si všechny hodnoty vložíme do pomocného pole, a potom je po jednom insertujeme. Vymyslete implementaci rehashe, která pomocné pole nepotřebuje. (Pozor na to, že během rehashe můžeme znovu začít s rehashem.)

Úloha 3 (Vyložení praktické systémy)

Uvažme systém funkcí $\mathcal{H}_1 = \{\text{id}\}$, který obsahuje jedinou funkci, jež zobrazí x na x . Je $\mathcal{H}_1$ c -univerzální pro nějaké c ? Je $\mathcal{H}_1$ (k, c) -nezávislý pro nějaká k a c ?

Dále uvažme systém $\mathcal{H}_2 = \{h_a(x) = a : a \in [m]\}$. Dokažte, že tento systém je $(1, 1)$ -nezávislý. Dále ukažte, že $\mathcal{H}_2$ není $(2, c)$ -nezávislý ani c -univerzální pro žádné c .

Úloha 4 (Modulo univerzálního systému nemusí být univerzální)

Ukažte, že pokud máme univerzální systém hashovacích funkcí $\mathcal{H}$, pak systém $\mathcal{H}'$, kde ke každé fci navíc přidáme modulo m , už nemusí být univerzální. Formálně: Dokažte, že pro každé c a $m > c$ existuje univerzum $\mathcal{U}$ a systém $\mathcal{H}$ z $\mathcal{U}$ do $\mathcal{U}$ tak, že $\mathcal{H}$ je univerzální, ale $\mathcal{H}'$ už není c -univerzální.

Úloha 5 (FKS (Fredman, Komlós, Szemerédi))

Ukážeme si konstrukci (statické) hashovací tabulky pro podmnožinu S velikosti n univerza $\mathcal{U}$, která nemá žádné kolize. Možná jste narazili na konstrukci, která potřebovala $\Omega(n^2)$ paměti (přesněji paměťových buněk). My to zvládneme s lineárním počtem paměťových buněk (za předpokladu, že můžeme mít zcela náhodnou hashovací funkci, tu dokážeme sestavit a samplovat v konstantním čase a že si ji dokážeme pamatovat v konstantním prostoru)¹.

Proces stavby tabulky bude probíhat následovně: budeme stavět dvě úrovně. V první úrovni si zcela náhodnou hashovací funkci f rozdělíme prvky S do kyblíků $B_1, \dots, B_n$ (a označíme si $b_i := |B_i|$). V druhé úrovni pak postavíme bezkolizní tabulku pomocí konstrukce, kdy máme pro kyblík B_i tabulku velikosti $2b_i^2$ pro b_i prvků, a zkusíme náhodně volit vhodnou hashovací funkci, dokud nemáme žádné kolize.

První úroveň: V konstantním čase si vybereme náhodnou hashovací funkci $f : \mathcal{U} \rightarrow [n]$, a tou rozdělíme S do kyblíků. Toto opakujeme, dokud neplatí, že $\sum_{i=1}^n b_i^2 \leq \beta n$ pro $\beta = 4$.

Chceme ukázat, že tento krok budeme ve střední hodnotě opakovat nejvýše dvakrát. Označme jako C počet kolizí.

a) Určete $\mathbb{E}[C]$.

b) Určete C v závislosti na b_i .

c) Na základě předchozích dvou hodnot určete $\mathbb{E}[\sum_{i=1}^n b_i^2]$.

d) Aplikujte Markovovu nerovnost na náhodnou veličinu $X = \sum_{i=1}^n b_i^2$ s vhodnou hodnotou, abychom dostali požadovaný výsledek. (Taky se bude hodit střední hodnota geometrického rozdělení.)

¹Totéž jde udělat s rozumně univerzální funkcí, tohle je jenom pro jednoduchost.

Druhá úroveň: Ve druhé úrovni pro každé $i \in [n]$ volíme v i -tém kyblíku univerzální hashovací funkci $g_i : \mathcal{U} \rightarrow [\alpha b_i^2]$ pro $\alpha = 2$. Toto opakujeme, dokud není prostá pro prvky v kyblíku B_i . Označme jako C_x počet kolizí klíče $x \in B_i$ na druhé úrovni.

- Shora odhadněte $\mathbb{E}[C_x]$.
- Použijte Markovovu nerovnost a union bound, abyste shora odhadli pravděpodobnost existence prvku s aspoň jednou kolizí.
- Kolikrát budeme muset proces opakovat? ([Sem si vložte si svou oblíbenou poznámku o střední hodnotě geometrického rozdělení.])

Užitečné pojmy

Definice (c -univerzální systém fcí). Systém $\mathcal{H}$ funkcí $h : \mathcal{U} \rightarrow [m]$ je c -univerzální pro $c > 0$, pokud pro všechna $x \neq y$ platí $\Pr_{h \in \mathcal{H}}[h(x) = h(y)] \leq \frac{c}{m}$.

Systém $\mathcal{H}$ je univerzální, pokud je c -univerzální pro nějaké $c > 0$.

Definice (k -nezávislý systém fcí). Systém $\mathcal{H}$ funkcí $h : \mathcal{U} \rightarrow [m]$ je (k, c) -nezávislý pro nějaká $k \geq 1, c > 0$, pokud $\Pr_{h \in \mathcal{H}}[h(x_1) = a_1 \wedge \dots \wedge h(x_k) = a_k] \leq \frac{c}{m^k}$ pro libovolná $x_1, \dots, x_k$ různá, $a_1, \dots, a_k$ ne nutně různá. Systém $\mathcal{H}$ je k -nezávislý, pokud je (k, c) -nezávislý pro nějakou nezávislou konstantu c .

Tvrzení (Union bound). Pro jevy A_1, A_2 platí, že $P[A_1 \cup A_2] \leq P[A_1] + P[A_2]$.

Věta (Markovova nerovnost). Buď X nezáporná náhodná veličina. Pak $\forall \varepsilon > 0$ platí $P[X \geq \varepsilon] \leq \frac{\mathbb{E}[X]}{\varepsilon}$. Ekvivalentně pro jakékoliv $d \geq 1$, $P[X \geq d \cdot \mathbb{E}[X]] \leq \frac{1}{d}$.