

## 8. cvičení

Datové struktury I, 25. 11. 2024

<https://iuuk.mff.cuni.cz/~chmel/2425/ds1/>

**Úloha 1** (Univerzální a nezávislá rozcvička z definic a formálních důkazů)

Dokažte, že

- $(2, c)$ -nezávislost implikuje  $c$ -univerzalitu.
- $(k, c)$ -nezávislost implikuje  $(k - 1, c)$ -nezávislost.

(Pokud si myslíte, že už jste to někde viděli, tak máte pravdu ... nicméně procvičení nikdy neuškodí :-)

**Úloha 2** (Lineární systém bez konstanty)

Víme, že systém funkcí  $h_{a,b}(x) = (ax + b \pmod p)$  pro  $a, b \in [p]$  je  $(2,1)$ -nezávislý, a po modulo  $m$  se toto změní na  $(2,4)$ -nezávislost (a 2-univerzalitu).

- Zdůvodněte, že tento systém není 3-nezávislý.
- Prozkoumejme, co se stane, když nebudeme přičítat  $b$ : uvažme tedy systém funkcí  $\{h_a(x) = (ax \pmod p) \pmod m : a \in \mathbb{Z}_p \setminus \{0\}\}$ . Je 2-nezávislý? Je  $c$ -univerzální pro nějaké  $c$ ? A co když dovolíme  $a = 0$ ?

**Úloha 3** (Vyloženě 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  $c$ .

**Věta.** *Tabulkové hashování je 3-nezávislé.*

**Úloha 4** (Tuhle větu si dokážeme)

Dokažte předcházející větu s následujícím postupem. Mějme  $a, b, c \in \mathbb{Z}_2^\ell, x \neq y \neq z \neq x \in \mathbb{Z}_2^w$ , a použijeme tabulkové hashování s  $d$  částmi. Pak chceme ukázat, že  $\Pr_{h \in \mathcal{H}}[h(x) = a \wedge h(y) = b \wedge h(z) = c] \leq \frac{1}{m^3}$ .

- Prvně si uvědomme, že pokud máme jen jednu část, a tedy jednu tabulku, tvrzení je triviální. Dále mějme alespoň dvě části. Protože  $x, y, z$  jsou různé, musí se (po dvou) lišit alespoň v jedné části.
- Začneme s případem, kdy existuje část  $i$ , že  $x^i, y^i, z^i$  jsou všechny různé. Mějme jakkoliv zvolené ostatní tabulky, kromě tabulky  $T_i$ . S jakou pravděpodobností můžeme zvolit funkci pro tabulku  $T_i$  tak, že  $h(x) = a, h(y) = b, h(z) = c$ ?
- Jinak existují (BÚNO) části  $i, j$  takové, že  $z^i = x^i \neq y^i$  a  $y^j = x^j \neq z^j$ . Potom máme následující soustavu rovnic, kde  $v_x, v_y, v_z$  jsou vyXORované výsledky z ostatních tabulek:

$$\begin{aligned}T_i[x^i] \oplus T_j[x^j] \oplus v_x &= a \\T_i[y^i] \oplus T_j[y^j] \oplus v_y &= b \\T_i[z^i] \oplus T_j[z^j] \oplus v_z &= c\end{aligned}$$

Opět si představme, že  $v_x, v_y, v_z$  už známe. S jakou pravděpodobností budou náhodně volené tabulky  $T_i, T_j$  splňovat tuto soustavu rovnic?

- Uvědomte si, že toto stačí.

**Úloha 5** (Pokud si ještě chcete hrát s definicemi... 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 6** (Bonus: perfektní hešování dle FKS (Fredman, Komlós, Szemerédi))

Ukážeme si konstrukci (statické) hashovací tabulky pro množinu  $S$  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 sestrojít a samplovat v konstantním čase a že si ji dokážeme pamatovat v konstantním prostoru)<sup>1</sup>.

Proces stavby tabulky bude probíhat následovně: budeme stavět dvě úrovně. V první úrovni si zcela náhodnou hashovací funkcí  $f$  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.

- Určete  $\mathbb{E}[C]$ .
- Určete  $C$  v závislosti na  $b_i$ .
- Na základě předchozích dvou hodnot určete  $\mathbb{E}[\sum_{i=1}^n b_i^2]$ .
- 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.)

**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 definice**

**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  $\Pr_h[h(x) = h(y)] \leq \frac{c}{m}$ .

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  $c$ .

**Definice** (Tabulkové hashování). Představme si, že chceme zahashovat  $n$ -bitové řetězky do  $m$ -bitových řetízků, kde  $n = k \cdot \ell$ . Řetízek  $x \in \{0, 1\}^n$  pak rozložíme do  $k$  částí délky  $\ell$ , které značíme  $x^i$ . Můžeme tedy psát  $x = x^1 x^2 \dots x^k$ . Pak generování naší hashovací funkce  $h : \{0, 1\}^n \rightarrow \{0, 1\}^m$  vypadá tak, že vybereme uniformně náhodně  $k$  funkcí  $T_i : \{0, 1\}^\ell \rightarrow \{0, 1\}^m$  (tyto reprezentujeme tabulkou, proto tabulkové hashování). Vyhodnocujeme pak  $h(x) = \bigoplus_{i=1}^k T_i(x^i) = T_1(x^1) \oplus T_2(x^2) \oplus \dots \oplus T_k(x^k)$ , kde  $\oplus$  značí XOR (po jednotlivých bitech).

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

---

<sup>1</sup>Totéž jde udělat s rozumně univerzální funkcí, tohle je jenom pro jednoduchost.