

# Hashing

Igor Čanadi, 12.12.2009.  
Bruno Rahle, 7.1.2011.

# Zadatak Cenzura

---

- ▶ Google intervju
- ▶ Vaše ideje?



# Hash tablica (tablica raspršenog adresiranja)

---

## ▶ Struktura podataka koja podržava operacije:

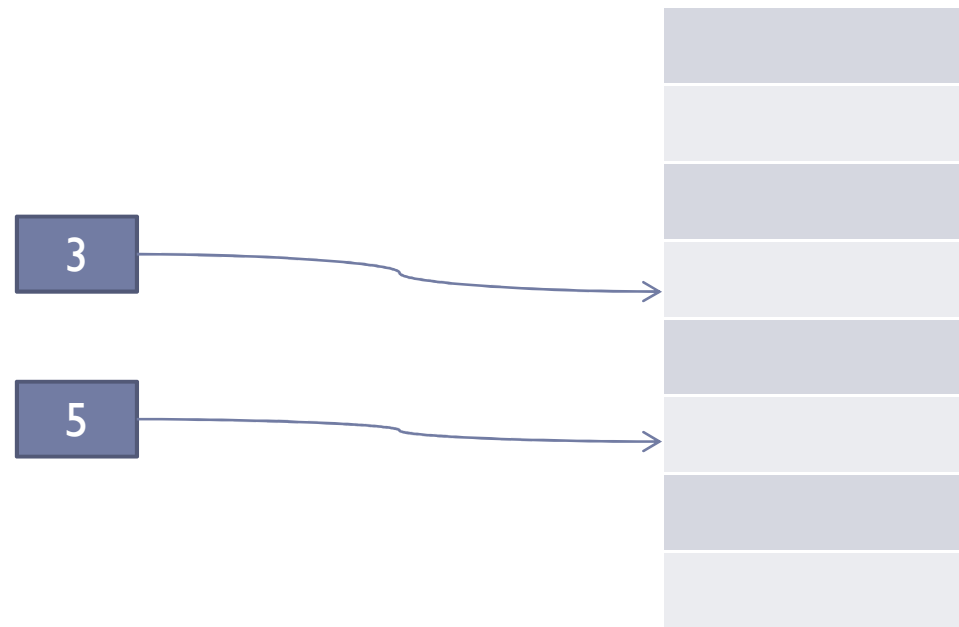
- ▶ Ubaci( $S, x$ ) :  $S = S \cup \{x\}$
- ▶ Izbaci( $x$ ) :  $S = S \setminus \{x\}$
- ▶ Pronađi( $k$ ) :  $x \in S$ , ako  $x.key = k$   
NULL, inače



# Direct access

---

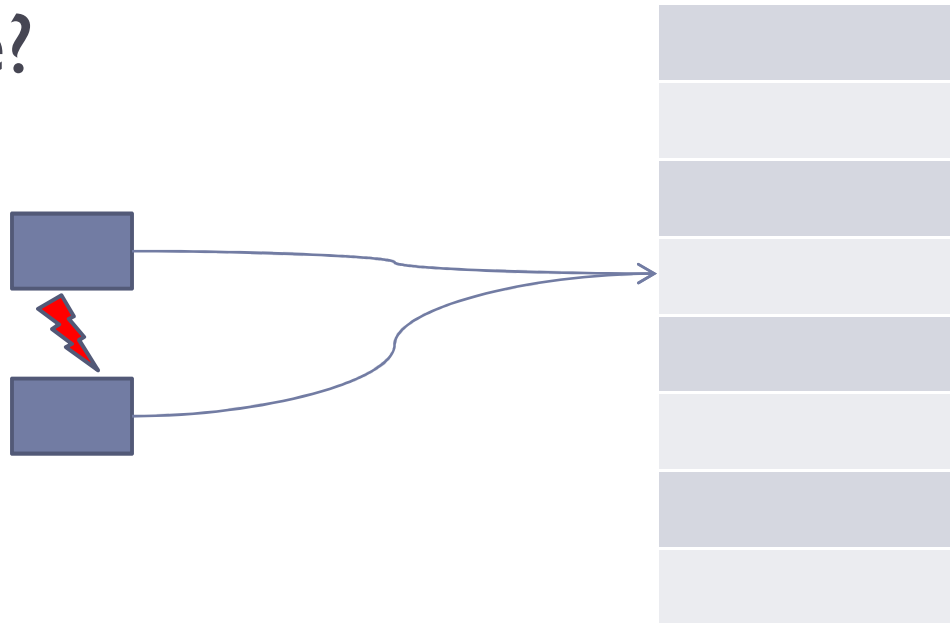
- ▶ Ako su svi ključevi **različiti** elementi skupa  $\{0, 1, \dots, m-1\}$
- ▶ Jednostavna implementacija s poljem
- ▶ Problem?



# Hashing

---

- ▶ Ključevi (ili cijeli objekti) se preslikavaju na skup  $\{0, 1, \dots, m-1\}$
- ▶ Problemi:
  - ▶ Preslikavanje?
  - ▶ Kolizije?



# Faktor opterećenja

---

▶  $\alpha = \text{broj elemenata} / \text{broj mjesta}$

# Preslikavanje – hash funkcija

---

- ▶ **Uvjeti:**
  - ▶ Uniformnost
  - ▶ Pravilnost ključeva ne smije utjecati na uniformnost

# Metoda dijeljenja

---

- ▶  $h(k) = k \bmod m$
- ▶  $m$  ne smije imat male djelitelje
- ▶  $m$  ne smije biti potencija brojeva 2 ili 10 (a nije dobro niti ako je blizu)

## Metoda množenja

---

- ▶  $h(k) = (A \cdot k \bmod 2^w) \text{ rsh } (w - r)$
- ▶  $w =$  širina riječi
- ▶  $m = 2^r$
- ▶  $2^{w-1} < A < 2^w$
- ▶  $A$  je neparan i nije blizu granica

## Otvoreno adresiranje

---

- ▶  $h : U \times \{0, \dots, m - 1\} \rightarrow \{0, \dots, m - 1\}$
- ▶ uniformnost - svaka  $h(k)$  je neka permutacija koja se odabire s jednakom vjerojatnošću

# Linear probing

---

- ▶  $h(k, i) = (h(k, 0) + i) \bmod m$
- ▶ Loše – zašto?

## Quadratic probing

---

▶  $h(k, i) = (h(k, 0) + i + i^2) \bmod m$

## Duplo hashiranje

---

- ▶  $h(k, i) = (h_1(k) + i \cdot h_2(k)) \bmod m$
- ▶ Super
- ▶  $m = 2^l$
- ▶  $h_2(k)$  je neparan
  
- ▶ Uz uniformnu hash funkciju
  - ▶  $E[\text{broj pokušaja}] \leq 1/(1 - \alpha)$

# Slabost

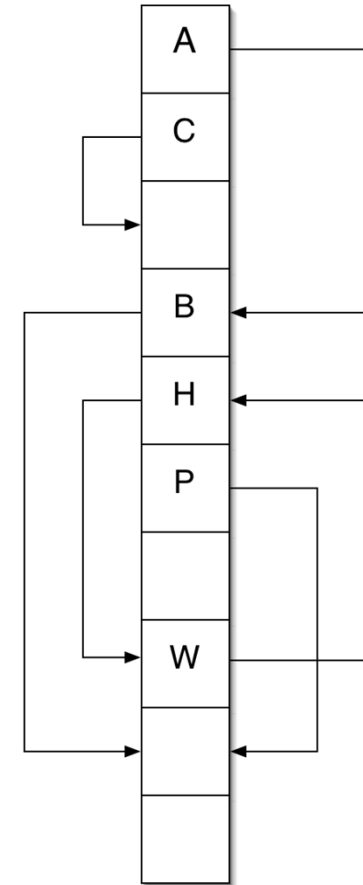
---

- ▶ Za svaku hash funkciju postoji skup loših ključeva
- ▶ Rješenje – random hash funkcija

# Kukavičji hashing

---

- ▶ Dvije hash funkcije
- ▶ U slučaju kolizije izbacuje drugi element



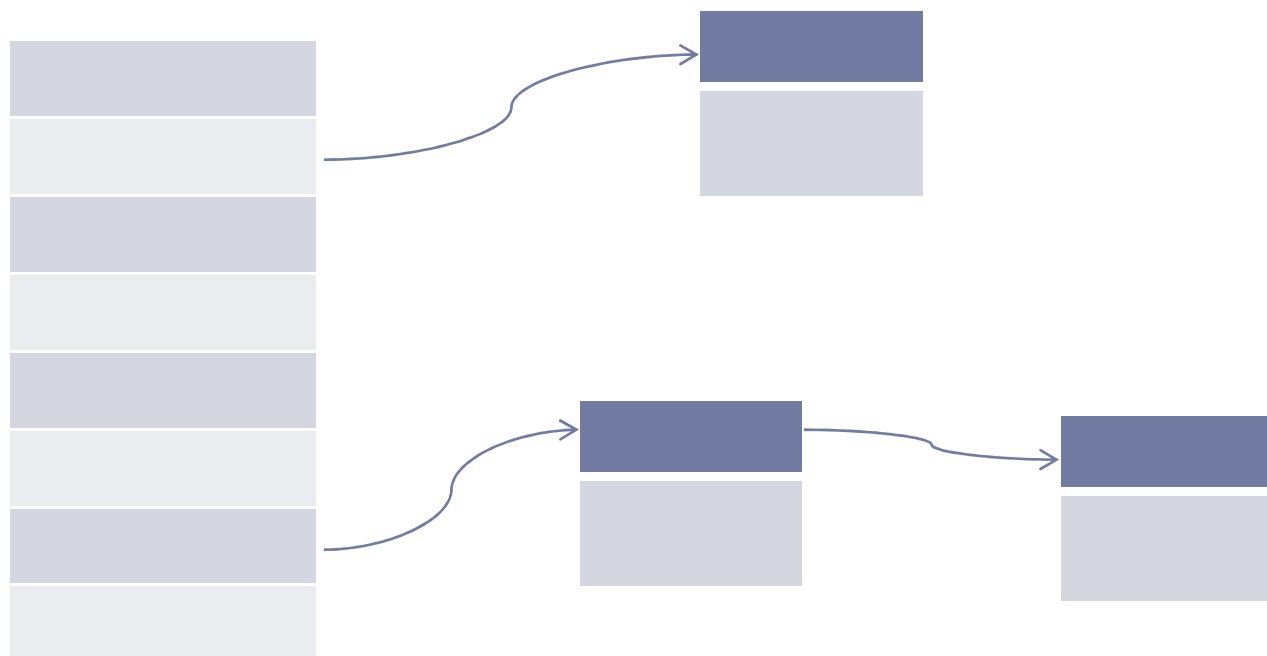
# Odvojeno ulančavanje

---

- ▶ Svaki pretinac je lista
- ▶ Na kraj liste dodaje se novi element
- ▶ Super!

# Ulančavanje

---



# Rolling hash function

---

- ▶ Pronalazak uzastopnih podnizova
- ▶ Fiksna duljina
- ▶ Iz  $h(x, x + l)$  možemo izračunati  $h(x + 1, x + l + 1)$  u konstantnom vremenu
- ▶ Rabin Karp

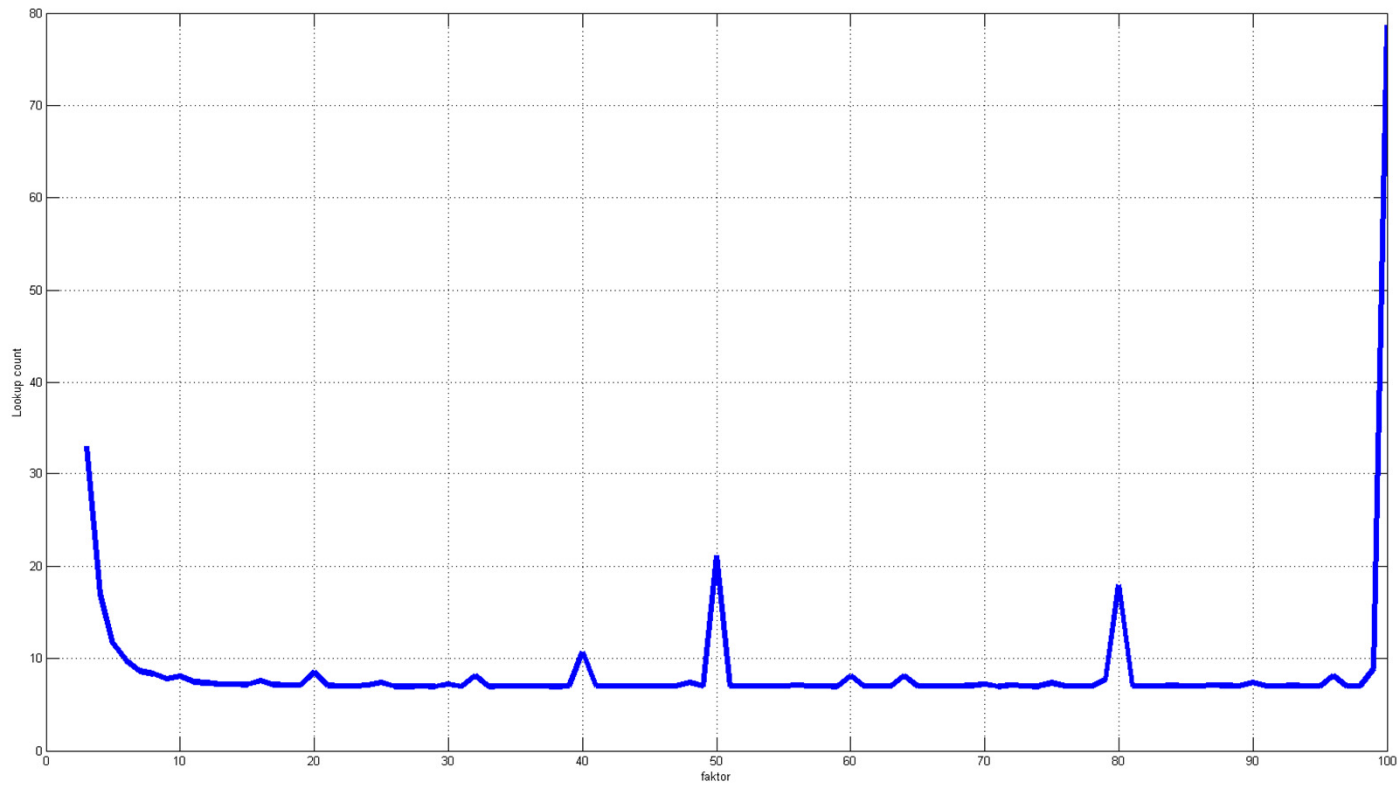
# Primjer

---

- ▶ Odvojeno ulančavanje
- ▶ Veličina tablice ( $m$ ): 100,000
- ▶ Broj ubačenih elemenata ( $n$ ): 1,000,000
- ▶ Dužina riječi: 1-10 znakova
- ▶ Load faktor: 7.0743
  - ▶ Što to znači?
- ▶ Hash funkcija:  $h(x, i) = (h(x, i - 1) p + x[i]) \% m$

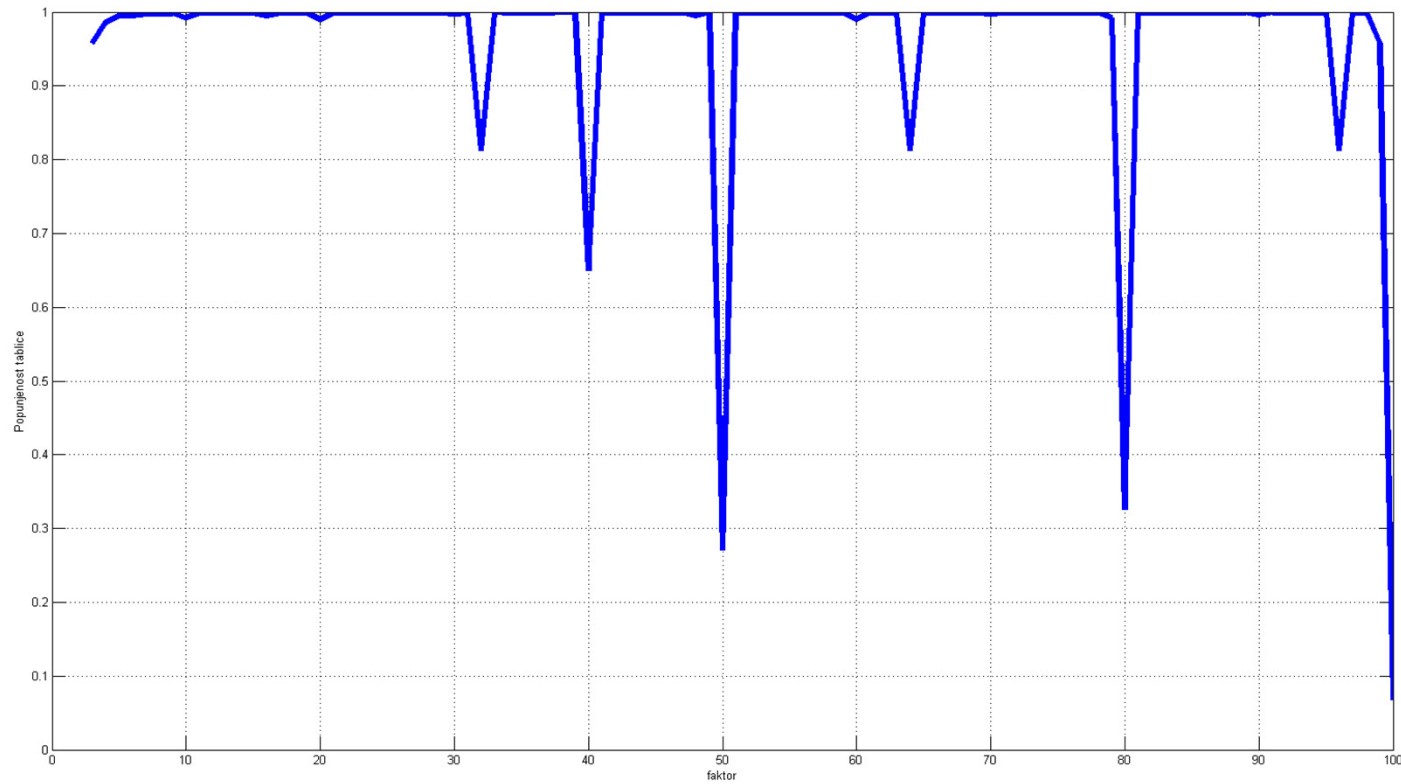
# Prosječan lookup count

---



# Prosječna popunjenost tablice

---



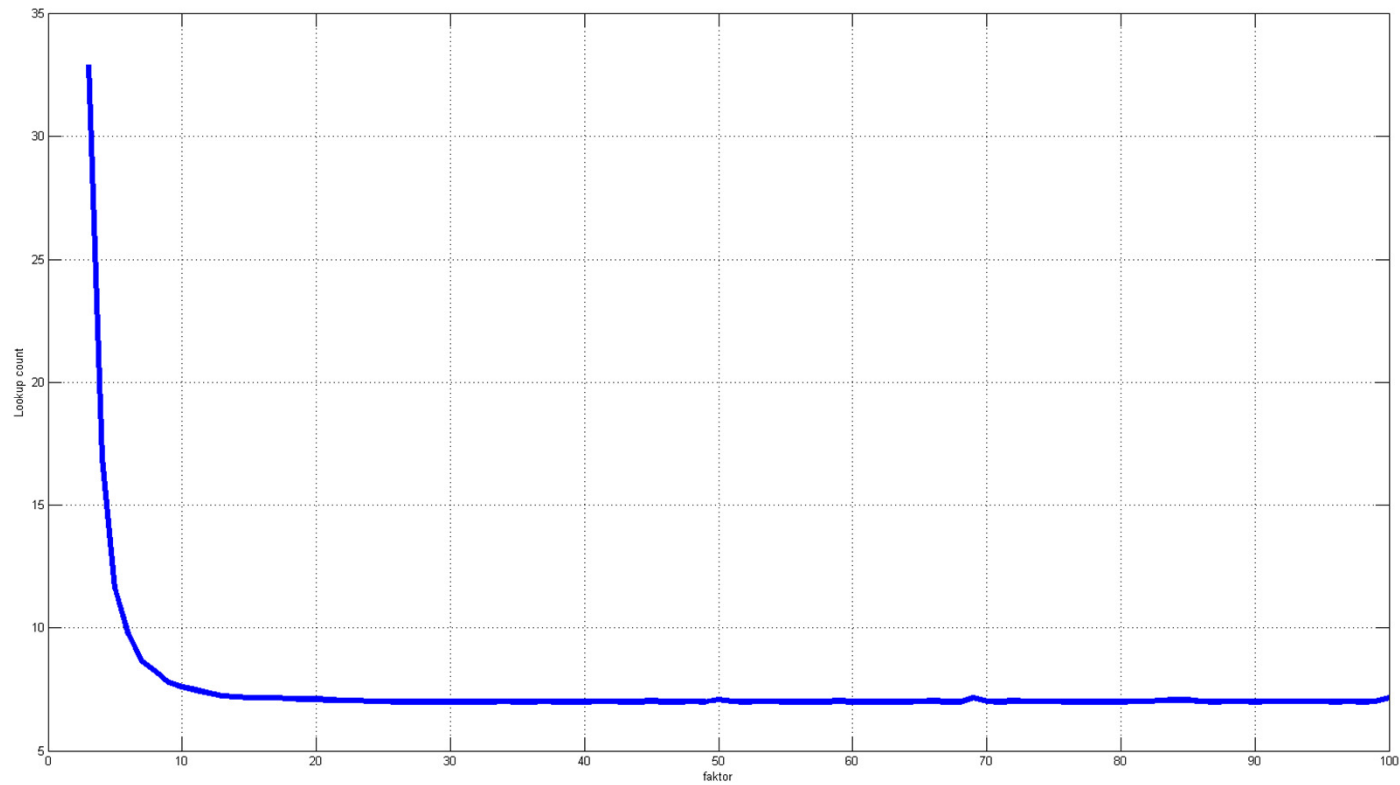
# Primjer

---

- ▶ Odvojeno ulančavanje
- ▶ Veličina tablice ( $m$ ): 99,989
- ▶ Broj ubačenih elemenata ( $n$ ): 1,000,000
- ▶ Dužina riječi: 1-10 znakova
- ▶ Load faktor: 7.082
  - ▶ Što to znači?
- ▶ Hash funkcija:  $h(x, i) = (h(x, i - 1) p + x[i]) \% m$

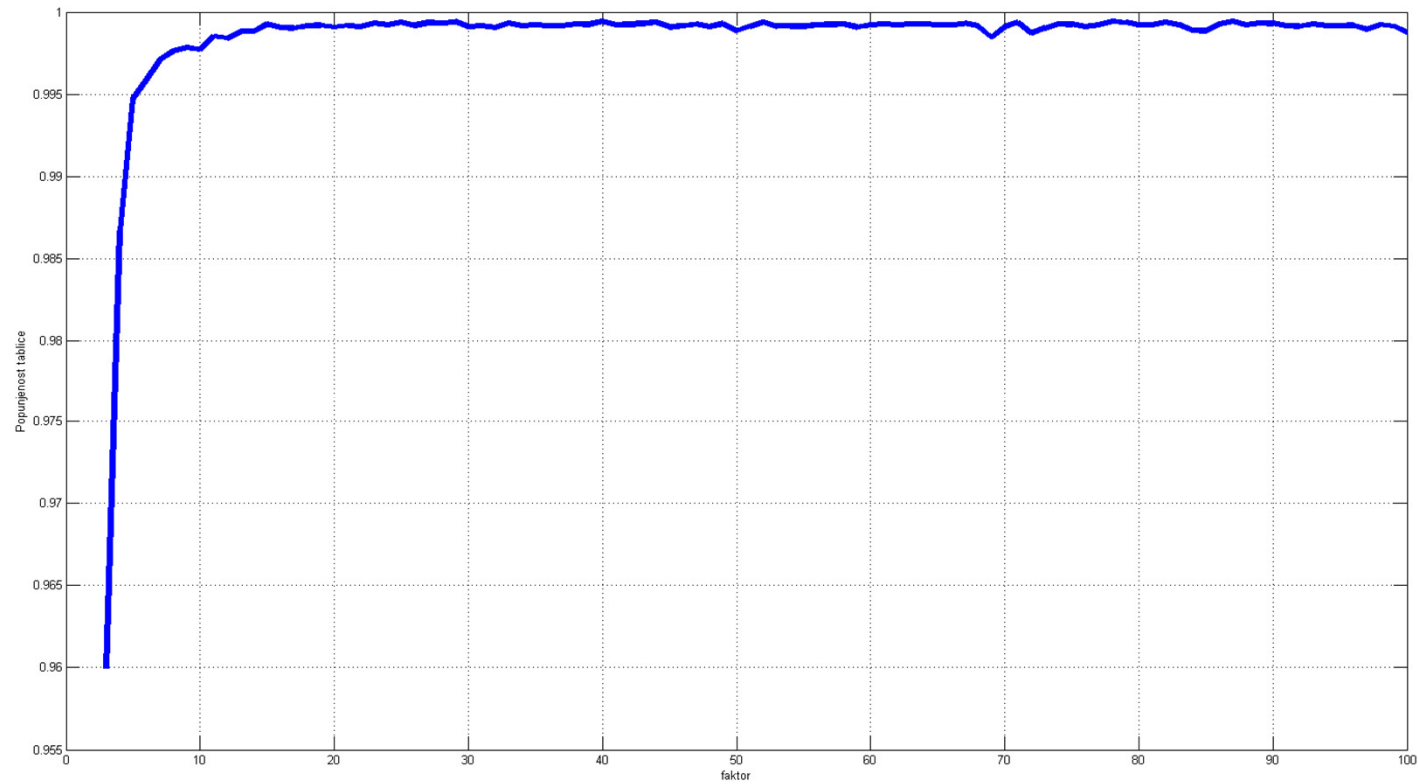
# Prosječan lookup count

---



# Prosječna popunjenost tablice

---



## Zadatak cenzura

---

- ▶ Hash funkcija s posebnim svojstima
  - ▶ Anagrami **uvijek** daju koliziju
  - ▶ Ne-anagrami **rijetko** daju koliziju
- ▶ Polinomna hash funkcija sortirane riječi

## Zadatak matrice (9. DZ)

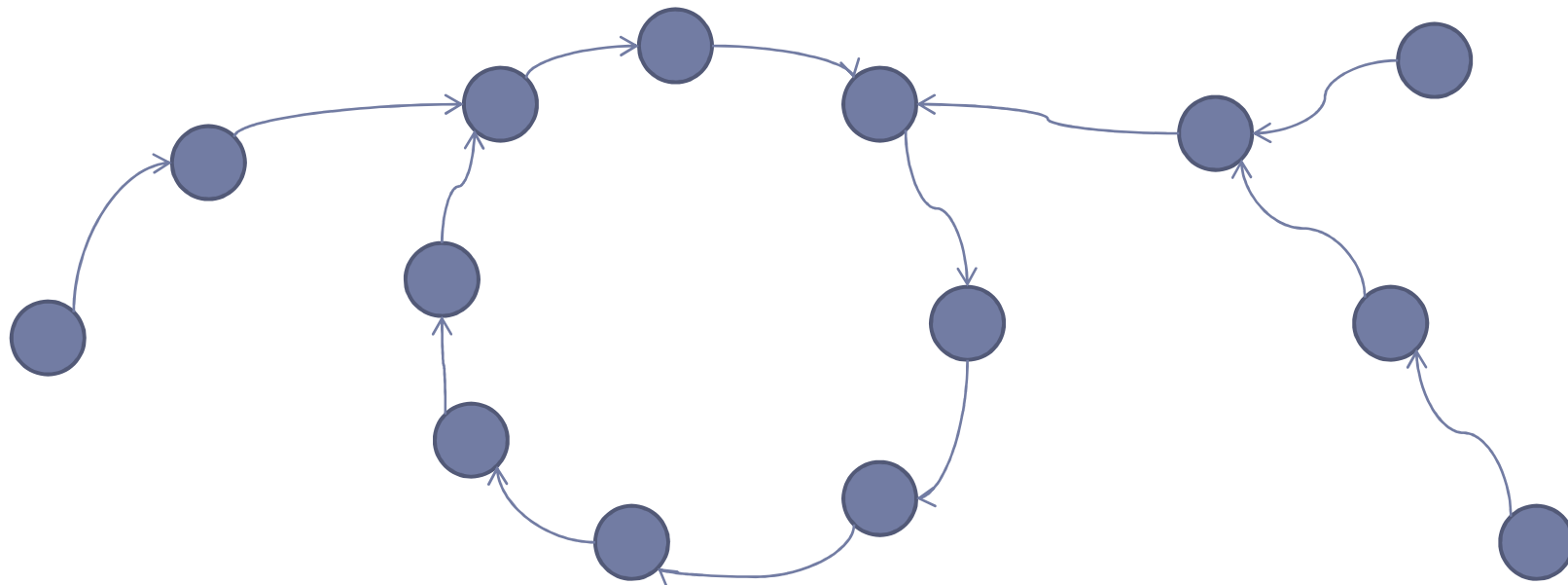
---

- ▶ Pronalazak malih matrica u velikoj
- ▶ Hashiranje malih matrica i svih podmatrica u velikoj
- ▶ Brže hashiranje podmatrica

# Wandering flea trainers

---

- ▶ <http://www.oi.edu.pl/php/show.php?ac=e100000&module=show&file=zadania/oi8/pchzad>



# Hvala na pozornosti!

---

