

Универзитет у Новом Саду, Факултет техничких наука

ОАС Софтверско инжењерство и информационе технологије

Организација података

# Статичка расута организација датотеке

# Садржај

- **Увод**
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- Ресурси

- Статичка расута датотека (Mogin, 2008)
  - при формирању датотеци бива додељено  $Q = bV$  локација, где  $N \leq Q$ , а  $Q$  остаје непроменљиво
    - $b$  – фактор бакетирања
    - $V$  – број бакета
    - $N$  – број слогова
  - применом трансформације вредности идентификатора слога у адресу добија се адреса бакета, у којем би тај слог требало да се налази
  - логичке везе између слогова нису меморисане, а слогови су у логичком погледу расути по меморијском простору датотеке

# Увод

- Врсте статичких расутих датотека (Mogin, 2008)
  - **статичке расуте датотеке с детерминистичком трансформацијом**
    - статичке расуте датотеке детерминистичке методе
  - **статичке расуте датотеке с пробабилистичком трансформацијом**
    - статичке расуте датотеке пробабилистичке методе

- Врсте статичких расутих датотека – примери (Mogin, 2008)

## статичке расуте датотеке

статичке расуте датотеке  
детерминистичке методе

↓  
директне датотеке

↓  
директне датотеке  
с машинским  
адресама

↓  
директне датотеке  
с релативним  
адресама

статичке расуте датотеке  
пробабилистичке методе

↓  
статичке расуте датотеке  
пробабилистичке методе  
с јединственим адресним  
простором

↓  
статичке расуте  
датотеке с отвореним  
адресирањем

↓  
статичке расуте  
датотеке с линеарним  
тражењем

↓  
статичке расуте  
датотеке са случајним  
тражењем

↓  
статичке расуте  
датотеке са спрезањем  
у једној зони

↓  
статичке расуте  
датотеке са спрезањем  
у примарној зони

↓  
статичке расуте  
датотеке  
пробабилистичке  
методе с две зоне

↓  
статичке расуте  
датотеке са зоном  
прекорачења

↓  
статичке расуте  
датотеке са спрезањем  
у зони прекорачења

↓  
статичке расуте  
датотеке са серијском  
зоном прекорачења

# Садржај

- Увод
- **Статичке расуте датотеке детерминистичке методе**
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- Ресурси

# Статичке расуте датотеке детерминистичке методе

- Врсте статичких расутих датотека – примери (Mogin, 2008)

## статичке расуте датотеке

статичке расуте датотеке  
детерминистичке методе

директне датотеке

директне датотеке  
с машинским  
адресама

директне датотеке  
с релативним  
адресама

статичке расуте датотеке  
пробабилистичке методе

статичке расуте датотеке  
пробабилистичке методе  
с јединственим адресним  
простором

статичке расуте  
датотеке  
пробабилистичке  
методе с две зоне

статичке расуте  
датотеке с отвореним  
адресирањем

статичке расуте  
датотеке са спрезањем  
у једној зони

статичке расуте  
датотеке са зоном  
прекорачења

статичке расуте  
датотеке с линеарним  
тражењем

статичке расуте  
датотеке са случајним  
тражењем

статичке расуте  
датотеке са спрезањем  
у примарној зони

статичке расуте  
датотеке са спрезањем  
у зони прекорачења

статичке расуте  
датотеке са серијском  
зоном прекорачења

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке <sup>(Mogin, 2008)</sup>
  - једна врста статичких расутих датотека с детерминистичком трансформацијом
    - примена тривијалне трансформације
      - вредност идентификатора је адреса бакета
        - $A_i = k_i$
    - идентификатор није део слога
  - врсте директних датотека
    - **директне датотеке с машинским адресама**
      - вредност идентификатора је машинска адреса
    - **директне датотеке с релативним адресама**
      - вредност идентификатора је релативна адреса

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с машинским адресама (Mogin, 2008)
  - вредност идентификатора је машинска адреса
    - општи облик у случају коришћења диска –  $(c, t, s)$ 
      - број цилиндра ( $c$ ), број стазе ( $t$ ), број сектора ( $s$ )
  - недостаци
    - чврста повезаност слогова датотеке с физичким карактеристикама меморијског уређаја
    - одсуство логичке везе између вредности идентификатора и садржаја слога
    - немогућност употребе путем виших програмских језика
      - потребно у програму задати машинску адресу
  - историјски превазиђена врста датотеке

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - вредност идентификатора је релативна адреса
    - локацијама меморијског простора датотеке придружени су редни бројеви од 1 до  $Q$  (или од 0 до  $Q - 1$ )
  - релативна адреса мора бити трансформисана у машинску
    - трансформација у оквиру релативне методе приступа
  - потребно дефинисати правило за придруживање вредности идентификатора слоговима
  - директне датотеке с релативним адресама додатно су познате као релативне датотеке

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - отклоњени неки недостаци директних датотека с машинским адресама
    - нема чврсте повезаности слогова датотеке с физичким карактеристикама уређаја
      - услед коришћења релативних адреса
    - могућа употреба путем виших програмских језика
      - постоји одређена подршка за релативне датотеке

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама
  - пример

табела идентификатора

$k(S_i)$	18	7	11	19	22	6	9
$S_i$	$S_1$	$S_2$	$S_3$	$S_4$	$S_5$	$S_6$	$S_7$
$k_i$	$A_1^1$	$A_1^2$	$A_1^3$	$A_2^1$	$A_2^2$	$A_2^3$	$A_3^1$

правило за придруживање вредности  
идентификатора слоговима

ДАТОТЕКА

	$A_1^1$	$A_1^2$	$A_1^3$
$A_1$	$\frac{18}{n(S_1)}$	$\frac{7}{n(S_2)}$	$\frac{11}{n(S_3)}$
	$A_2^1$	$A_2^2$	$A_2^3$
$A_2$	$\frac{19}{n(S_4)}$	$\frac{22}{n(S_5)}$	$\frac{6}{n(S_6)}$
	$A_3^1$	$A_3^2$	$A_3^3$
$A_3$	$\frac{9}{n(S_7)}$	$\frac{*}{\quad}$	$\frac{\quad}{\quad}$

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - релативна метода приступа
    - присутна подршка код оперативних система мејнфрејм рачунара и неких програмских језика, нпр. језика *Cobol*
      - стандардизовано формирање, коришћење и ажурирање
      - поглед на датотеку као на низ слогова
        - релативна адреса слога је редни број слога
      - пружање услуга на нивоу блока
        - једини могући фактор блокирања је 1
      - у неким ситуацијама потребно на нивоу апликативног програма имплементирати одређене активности

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - релативна метода приступа
    - нема посебне подршке код оперативног система *Unix* и неких програмских језика, нпр. језика *C*
      - стандардне библиотечке функције
      - основни поглед на датотеку као на низ бајтова
      - нема разликовања организација датотека
      - на нивоу апликативног програма потребно имплементирати активности које се тичу конкретне организације

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - формирање
    - читавање узастопних слогова из серијске датотеке
    - придруживање вредности идентификатора слоговима
    - смештање слогова у датотеку која бива формирана
      - смештање у локације чије релативне адресе представљају вредности идентификатора

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - тражење
    - основни начин тражења је применом методе трансформације вредности идентификатора у адресу
      - кораци тражења
        - трансформација је тривијална и вредност идентификатора представља релативну адресу
        - на основу добијене адресе бива учитан одговарајући бакет
        - ако слога има у учитаном бакету, тражење је успешно, а, ако га тамо нема, тражење је неуспешно
      - тражење случајно одабраног слога и тражење логички наредног слога изводе се на исти начин
    - потенцијални начин тражења је и применом методе линеарног тражења

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - тражење – перформансе
    - број приступа за успешно тражење  $R_u$   
 $R_u = 1$
    - број поређења за успешно тражење  $U_u$ 
      - случај када је слогу придружена адреса локације  
 $U_u = 1$
      - случај када је слогу придружена адреса бакета  
 $1 \leq U_u \leq b$

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - тражење – перформансе
    - тражење подразумева коришћење табеле идентификатора
      - табела идентификатора је обично релативно малог обима
        - могућност непрекидног држања табеле идентификатора у оперативној меморији током рада над датотеком

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - ажурирање
    - упис
      - новом слогу бива придружена вредност идентификатора, што је релативна адреса и слог бива уписан у датотеку у одговарајући бакет, под условом да има слободне локације
      - може бити проверено да ли у датотеци већ постоји слог с истом вредношћу кључа као код новог слога
    - модификација
      - извођење би могло бити налик извођењу логичког брисања али уз измену садржаја одговарајућег поља а не поља статуса
    - брисање
      - примена логичког брисања
        - читање одговарајућег бакета и, након провере вредности кључа и измене садржаја поља статуса, писање бакета

# Статичке расуте датотеке детерминистичке методе

- Директне датотеке с релативним адресама (Mogin, 2008)
  - одлике и примена
    - перформансе тражења
      - тражење логички наредног слога
        - већа ефикасност него код серијских датотека
        - мања ефикасност него код секвенцијалних датотека
      - ограничена могућност шире примене услед одсуства логичке везе између вредности идентификатора и садржаја слога
        - потенцијално решење је употреба вредности бројчаног кључа као вредности идентификатора, која одговара адреси
          - искоришћење меморијског простора може бити веома лоше
            - потребна засебна локација за сваку могућу вредност кључа
            - број могућих вредности кључа је често много већи од броја актуелних вредности кључа

# Садржај

- Увод
- Статичке расуте датотеке детерминистичке методе
- **Методе пробабилистичке трансформације**
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- Ресурси

# Методе пробабилистичке трансформације

- Методе пробабилистичке трансформације вредности кључа у адресу (Mogin, 2008)
  - вредност кључа може бити једна од  $v^p$  (или  $v^p - 1$ ) вредности
    - $v$  – основа бројног система (број знакова алфавета)
    - $p$  – општи број позиција
    - пример
      - бројчани кључ од девет позиција ( $p = 9$ ) и децимални систем ( $v = 10$ )
        - постоји  $v^p = 10^9 = 1000000000$  могућих вредности кључа
          - од 0 до 999999999

# Методе пробабилистичке трансформације

- Методе пробабилистичке трансформације вредности кључа у адресу (Mogin, 2008)

- у пракси је број могућих вредности кључа обично много већи од укупног броја локација у датотеци и броја актуелних слогова у датотеци

$$v^p \gg Q \geq N$$

- детерминистичка трансформација неодговарајућа
  - код такве трансформације с једном адресом може бити повезана највише једна вредност кључа
- пробабилистичка трансформација може бити искоришћена
  - код такве трансформације с једном адресом може бити повезано више различитих вредности кључа

# Методе пробабилистичке трансформације

- Методе пробабилистичке трансформације вредности кључа у адресу (Mogin, 2008)
  - општи облик трансформације вредности кључа у адресу
    - функција  $h$  која пресликава домен кључа  $K$  на скуп адреса  $A$ 
      - $h : \text{dom}(K) \rightarrow A$
    - $A_i = h(k_i)$  представља адресу бакета у који би требало сместити слог  $S_i$ 
      - $i \in \{1, \dots, N\}$
      - $k_i$  – вредност кључа слога  $S_i$
  - циљ да трансформација вредности кључа у адресу буде што равномернија (случајнија)
    - ослањање на неку методу за генерисање псеудослучајних бројева

# Методе пробабилистичке трансформације

- Методе пробабилистичке трансформације вредности кључа у адресу (Mogin, 2008)
  - четири општа корака трансформације
    - 1) претварање ненумеричке у нумеричку вредност кључа
      - у случају да је вредност кључа изворно ненумеричка
    - 2) претварање нумеричке вредности кључа у псеудослучајан број
      - примена формуле за генерисање псеудослучајних бројева
    - 3) превођење вредности псеудослучајног броја у опсег дозвољених вредности релативне адресе
      - кориговање псеудослучајног броја одговарајућим фактором чиме се добија исправна релативна адреса
    - 4) претварање релативне у машинску адресу
      - помоћу релативне методе приступа

# Методе пробабилистичке трансформације

- Методе пробабилистичке трансформације вредности кључа у адресу (Mogin, 2008)
  - примери метода пробабилистичке трансформације
    - **метода остатка при дељењу**
    - **метода централних цифара квадрата кључа**
    - **метода преклапања**

# Методе пробабилистичке трансформације

- Метода остатка при дељењу (Mogin, 2008)
  - преглед корака
    - адреса  $A$  се добија на основу остатка при дељењу вредности кључа  $k(S)$  одабраним бројем  $m$

# Методе пробабилистичке трансформације

- Метода остатка при дељењу (Mogin, 2008)
  - формални кораци
    - остатак при дељењу вредности кључа  $k(S)$  бројем  $m$  бива увећан за један
$$A = 1 + k(S) \pmod{m}$$
      - примена случаја  $m = B$ 
        - није потребно додатно довођење у опсег дозвољених вредности адресе
      - одабир броја  $m$ 
        - $m$  да не буде паран број
        - $m$  да не одговара степену основе бројног система
        - $m$  да буде прост број или непаран број с релативно великим простим чиниоцима

# Методе пробабилистичке трансформације

- Метода остатка при дељењу

- пример

- вредности кључа из опсега од 0 до 999 ( $v = 10, p = 3$ )
    - датотека обухвата  $B = 7$  бакета (адресе у опсегу од 1 до 7)
    - рачунање адресе

- вредност кључа  $k(S) = 24$ 
        - одговарајућа адреса за  $m = B$

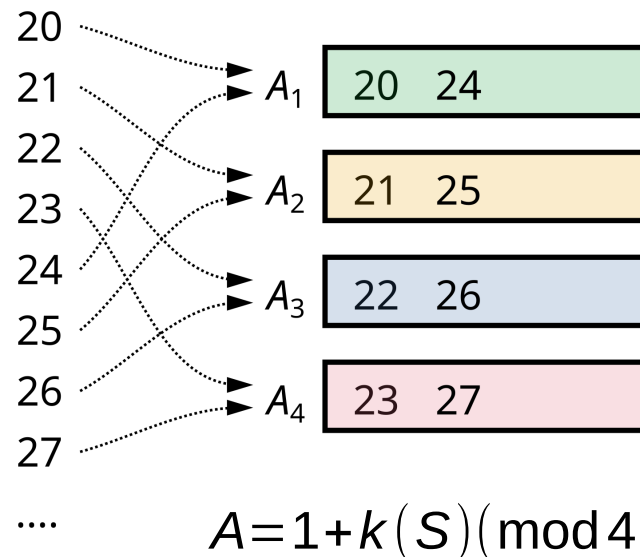
$$A = 1 + k(S) \pmod{m} = 1 + 24 \pmod{7} = 1 + 3 = 4$$

- вредност кључа  $k(S) = 179$ 
        - одговарајућа адреса за  $m = B$

$$A = 1 + k(S) \pmod{m} = 1 + 179 \pmod{7} = 1 + 4 = 5$$

# Методе пробабилистичке трансформације

- Метода остатка при дељењу (Mogin, 2008)
  - примена
    - за случајеве појаве вредности кључа у пакетима
      - пакет обухвата узастопне вредности кључа
      - за узастопне вредности кључа метода генерише узастопне адресе (циркуларно посматрано)
        - бакети бивају равномерно попуњавани



# Методе пробабилистичке трансформације

- Метода централних цифара квадрата кључа (Mogin, 2008)
  - преглед корака
    - вредност кључа  $k(S)$  бива квадрирана
      - $v$  – основа бројног система
      - $p$  – општи број позиција у вредности кључа
    - бива формиран псеудослучајни број  $T$  од онолико централних цифара квадрата колико је потребно за адресу
      - $n$  – општи број позиција у адреси
$$n = \lceil \log_v B \rceil$$
    - по потреби, формиран псеудослучајни број  $T$  бива коригован како би била добијена адреса из дозвољеног опсега

# Методе пробабилистичке трансформације

- Метода централних цифара квадрата кључа (Mogin, 2008)

- формални кораци, део 1

- вредност кључа у полиномном облику

$$k(S) = \sum_{i=0}^{p-1} a_i v^i, \quad a_i \in \{0, 1, \dots, v-1\}$$

- квадрат вредности кључа у полиномном облику

$$(k(S))^2 = \sum_{i=0}^{2p-1} c_i v^i, \quad c_i \in \{0, 1, \dots, v-1\}$$

- централне цифре квадрата вредности кључа

$$(c_{t+n-1}, c_{t+n-2}, \dots, c_{t+1}, c_t)$$

$$t = \left\lfloor p - \frac{n}{2} \right\rfloor$$

# Методе пробабилистичке трансформације

- Метода централних цифара квадрата кључа (Mogin, 2008)
  - формални кораци, део 2
    - псеудослучајни број формиран на основу централних цифара

$$T = \sum_{i=0}^{n-1} c_{t+i} v^i$$

- адреса добијена кориговањем псеудослучајног броја

$$A = 1 + \left\lfloor \frac{B}{v^n} T \right\rfloor$$

# Методе пробабилистичке трансформације

- Метода централних цифара квадрата кључа

- пример

- вредност кључа из опсега од 0 до 999 ( $v = 10, p = 3$ )
    - датотека обухвата  $B = 150$  бакета (адресе у опсегу од 1 до 150)  
 $n = \lceil \log_v B \rceil = \lceil \log_{10} 150 \rceil = \lceil 2,176 \dots \rceil = 3$
    - рачунање адресе

- вредност кључа  $k(S) = 37$

$$k(S) = 37 = 7 \cdot 10^0 + 3 \cdot 10^1 + 0 \cdot 10^2$$

$$(k(S))^2 = 1369 = 9 \cdot 10^0 + 6 \cdot 10^1 + 3 \cdot 10^2 + 1 \cdot 10^3 + 0 \cdot 10^4 + 0 \cdot 10^5$$

$$t = \left\lfloor p - \frac{n}{2} \right\rfloor = \left\lfloor 3 - \frac{3}{2} \right\rfloor = 1 \quad (k(S))^2 = 00 \mathbf{1369}$$

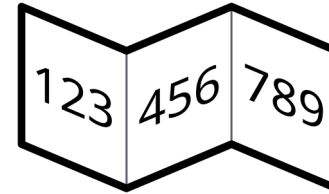
$$T = 6 \cdot 10^0 + 3 \cdot 10^1 + 1 \cdot 10^2 = 136$$

$$A = 1 + \left\lfloor \frac{B}{v^n} T \right\rfloor = 1 + \left\lfloor \frac{150}{10^3} 136 \right\rfloor = 21$$

# Методе пробабилистичке трансформације

- Метода преклапања (Mogin, 2008)

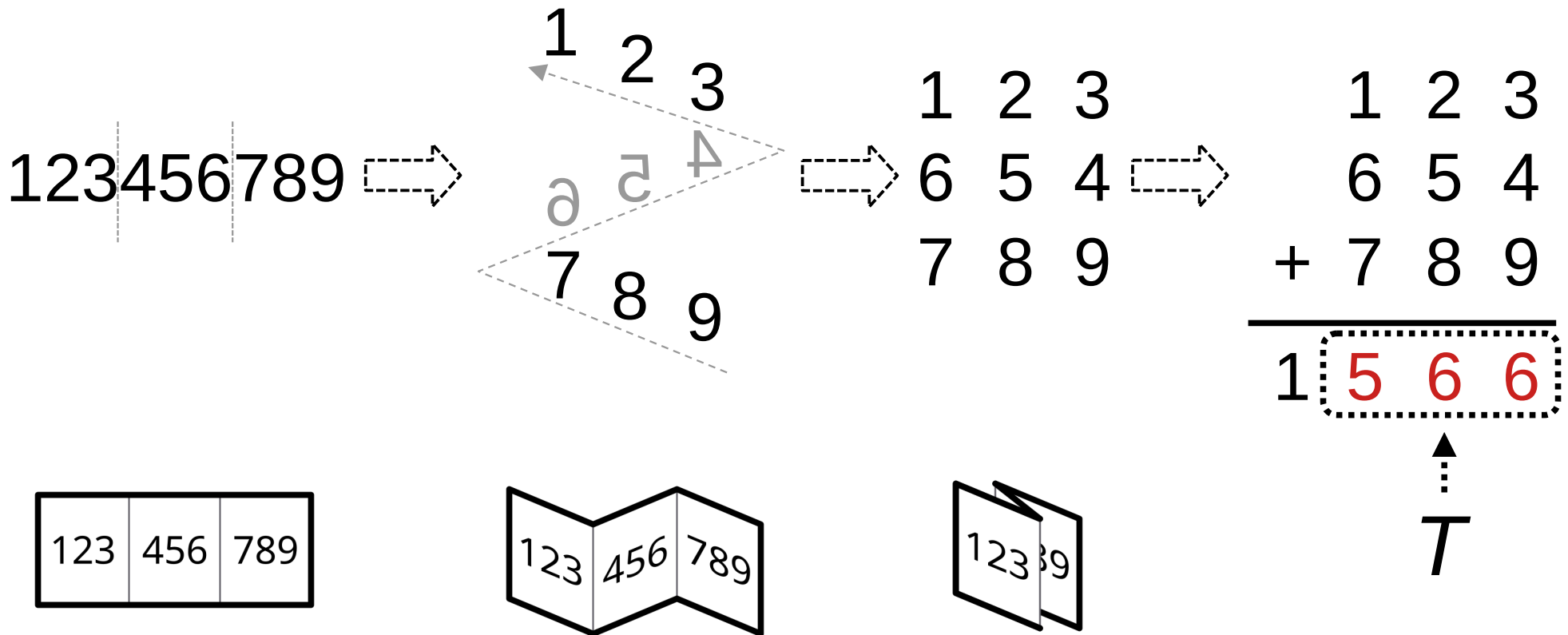
- преглед корака



- цифре вредности кључа бивају преклапане по сегментима
      - број сегмената (сваки сегмент дужине  $n$ )
        - $\left\lfloor \frac{p}{n} \right\rfloor$
      - посматрано с десне стране, у сваком другом сегменту цифре су у обрнутом редоследу у односу на редослед у вредности кључа
    - за сваки сегмент бива формиран број комбиновањем цифара
    - бива формиран псеудослучајни број  $T$  сабирањем формираних бројева по модулу  $v^n$
    - по потреби, формиран псеудослучајни број  $T$  бива коригован како би била добијена адреса из дозвољеног опсега

# Методе пробабилистичке трансформације

- Метода преклапања
  - преглед корака – илустрација преклапања и сабирања



# Методе пробабилистичке трансформације

- Метода преклапања (Mogin, 2008)

- формални кораци, део 1

- вредност кључа у полиномном облику

$$k(S) = \sum_{i=0}^{p-1} a_i v^i, \quad a_i \in \{0, 1, \dots, v-1\}$$

- псеудослучајни број формиран преклапањем и сабирањем цифара вредности кључа

$$T = \left( \sum_{k=0}^q \sum_{i=0}^{n-1} c_r v^i + \sum_{k=1}^q \sum_{i=0}^{n-1} c_s v^i \right) \pmod{v^n}$$

$$q = \left\lfloor \frac{p}{2n} \right\rfloor \quad \begin{array}{l} r = 2kn + i \\ s = 2kn - i - 1 \end{array} \quad c_r = \begin{cases} a_r, & r < p \\ 0, & r \geq p \end{cases} \quad c_s = \begin{cases} a_s, & s < p \\ 0, & s \geq p \end{cases}$$

# Методе пробабилистичке трансформације

- Метода преклапања (Mogin, 2008)
  - формални кораци, део 2
    - адреса добијена кориговањем псеудослучајног броја

$$A = 1 + \left\lfloor \frac{B}{v^n} T \right\rfloor$$

# Методе пробабилистичке трансформације

- Метода преклапања

- пример

- вредност кључа из опсега од 0 до 9999999999 ( $v = 10, p = 9$ )
    - датотека обухвата  $B = 175$  бакета (адресе у опсегу од 1 до 175)  
 $n = \lceil \log_v B \rceil = \lceil \log_{10} 175 \rceil = \lceil 2,243... \rceil = 3$
    - рачунање адресе

- вредност кључа  $k(S) = 123456789$

$$\begin{aligned} T &= (((9 \cdot 10^0 + 8 \cdot 10^1 + 7 \cdot 10^2) + (3 \cdot 10^0 + 2 \cdot 10^1 + 1 \cdot 10^2)) + \\ &\quad (4 \cdot 10^0 + 5 \cdot 10^1 + 6 \cdot 10^2)) \pmod{10^3} \\ &= ((789 + 123) + (654)) \pmod{1000} \\ &= (1566) \pmod{1000} \\ &= 566 \end{aligned}$$

$$A = 1 + \left\lfloor \frac{B}{v^n} T \right\rfloor = 1 + \left\lfloor \frac{175}{10^3} 566 \right\rfloor = 100$$

# Методе пробабилистичке трансформације

- Метода преклапања (Mogin, 2008)
  - примена
    - за случајеве када је општи број позиција у вредности кључа  $p$  много већи од општег броја позиција у адреси  $n$

# Садржај

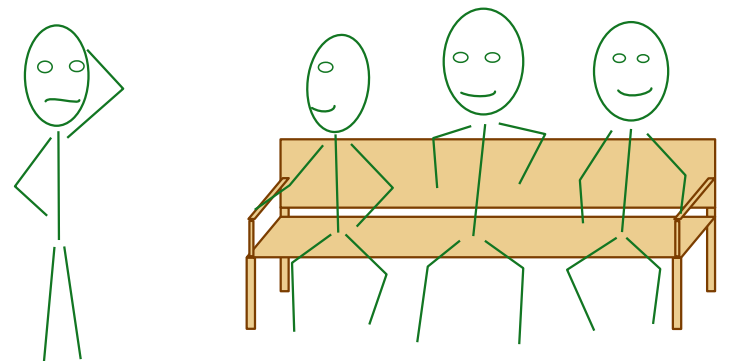
- Увод
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- **Статичке расуте датотеке пробабилистичке методе**
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- Ресурси

# Статичке расуте датотеке пробабилистичке методе

- Статичке расуте датотеке с пробабилистичком трансформацијом (Mogin, 2008)
  - примена статичке доделе простора за датотеку
  - примена пробабилистичке трансформације вредности кључа у адресу при одређивању локације за слог
    - превазилажење одсуства логичке везе између вредности идентификатора и садржаја слога
      - употреба кључа као идентификатора
    - превазилажење ограничења које прати примену детерминистичке трансформације
      - није неопходно да за сваку могућу вредност кључа буде предвиђена другачија адреса, па ни да унапред буде одређен простор довољно велик за смештање скупа слогова у којем је свака могућа вредност кључа искоришћена

# Статичке расуте датотеке пробабилистичке методе

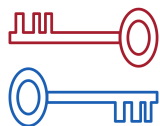
- Статичке расуте датотеке с пробабилистичком трансформацијом (Mogin, 2008)
  - примена статичке доделе простора за датотеку
  - примена пробабилистичке трансформације вредности кључа у адресу при одређивању локације за слог
    - отварање другачијег проблема
      - могуће је да за неки слог нема места на одговарајућој адреси, јер је тамошњи простор већ сасвим искоришћен



# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)

- **колизија кључева**



- појава да за две различите вредности кључа резултат трансформације буде иста адреса

$$k_i \neq k_j \wedge h(k_i) = h(k_j)$$

- $i \neq j$
- $i, j \in \{1, \dots, N\}$
- појава синонима

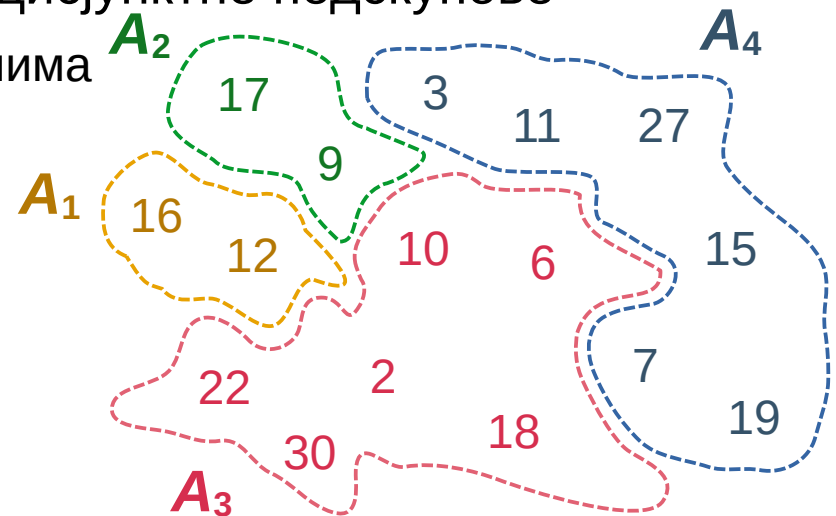
# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)

- **СИНОНИМИ**

- синоними су слогови који имају различите вредности кључа али којима према трансформацији вредности кључа у адресу одговара иста адреса
- на основу примене методе трансформације скуп слогова датотеке може бити подељен на дисјунктне подскупове

- сваки подскуп је један скуп синонима
- подскуп може бити празан
- подскупова има колико и бакета



$$A = 1 + k(S) \pmod{4}$$

# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)
  - **матични бакет**
    - матични бакет за дати слог је бакет чија се адреса добија као резултат трансформације вредности кључа тог слога
    - слог би требало да се нађе у свом матичном бакету, али, услед попуњености матичног бакета, могуће је да слог буде негде другде
  - врсте слогова у датотеци према бакету у којем се налазе
    - **примарни слогови**
      - слогови који се налазе у својим матичним бакетима
    - **прекорачиоци (слогови прекорачиоци)**
      - слогови који се не налазе у својим матичним бакетима

# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)
  - **прекорачиоци**
    - непожељни услед негативног утицаја на перформансе у раду
      - приликом уписа потребно је и пронаћи слободну локацију за смештај прекорачиоца
      - приликом тражења слога који је прекорачилац време приступа је продужено
    - обично неизбежни

# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)

- **фактор попуњености  $q$**

- однос броја слогова  $N$  и броја локација датотеке  $Q$

$$q = \frac{N}{Q}, \quad 0 < q \leq 1$$

$$Q = bV$$

# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)
  - **фактор бакетирања  $b$** 
    - број локација за смештај слогова у једном бакету
      - бакети може садржати слободне локације
        - начини за распознавање слободних локација
          - употреба индекса слободних локација, уз спрезање слободних локација
            - увођење посебног показивача на почетак листе слободних локација
            - проширење локације пољем показивача, за потребе спрезања
          - употреба посебне вредности статуса у пољу статуса
            - увођење вредности статуса за слободну локацију
          - употреба специјалних вредности као садржаја локације
            - нпр. ознака \* у пољу кључа

# Статичке расуте датотеке пробабилистичке методе

- Карактеристични појмови (Mogin, 2008)

- број бакета ***B***

- одређивање броја бакета на основу броја слогова *N* и пројектованих вредности *q* и *b*

$$B = \left\lceil \frac{N}{bq} \right\rceil$$

# Статичке расуте датотеке пробабилистичке методе

- Пројектовање датотеке (Mogin, 2008)
  - пројектантски задаци
    - одређивање фактора попуњености меморијског простора  $q$
    - одређивање фактора бакетирања  $b$
    - одабир методе трансформације вредности кључа у адресу
    - одабир поступка за смештање прекорачилаца
  - главни чиниоци у извршавању пројектантских задатака
    - расподела вредности кључа у опсегу његових могућих вредности
    - величина слога
    - обим и карактер ажурирања датотеке
    - задати средњи број приступа датотеци при успешном и неуспешном тражењу

# Статичке расуте датотеке пробабилистичке методе

- Пројектовање датотеке (Mogin, 2008)
  - пројектантски задаци – фактор попуњености  $q$ 
    - критеријуми при одређивању  $q$ 
      - вероватноћа појаве више слогова у једном скупу синонима (мање је боље)
        - расте с повећањем  $q$
        - искоришћеност меморијског простора (веће је боље)
          - расте с повећањем  $q$
      - избор вредности  $q$  у пракси
        - $q \leq 0,8$

# Статичке расуте датотеке пробабилистичке методе

- Пројектовање датотеке (Mogin, 2008)
  - пројектантски задаци – фактор бакетирања  $b$ 
    - критеријуми при одређивању  $b$ 
      - вероватноћа појаве прекорачилаца по бакету, за дато  $q$  (мање је боље)
        - опада с повећањем  $b$ 
          - под претпоставком да је иста вероватноћа доделе сваке од расположивих адреса сваком од слогова датотеке и да додељивања адреса представљају међусобно независне догађаје
        - време размене података између оперативне меморије и екстерног меморијског уређаја, као и величина потребног бафера у оперативној меморији (мање је боље)
          - расте с повећањем  $b$
      - избор вредности  $b$  у пракси
        - $b \leq 10$

# Статичке расуте датотеке пробабилистичке методе

- Пројектовање датотеке (Mogin, 2008)
  - пројектантски задаци – метода трансформације
    - зависно од расподеле вредности кључа у конкретном случају
      - циљ је постићи најравномернију расподелу слогова по бакетима

# Статичке расуте датотеке пробабилистичке методе

- Пројектовање датотеке (Mogin, 2008)
  - пројектантски задаци – смештање прекорачилаца
    - врсте расутих датотека према општој организацији простора у односу на смештање прекорачилаца
      - **датотеке с јединственим адресним простором**
        - у заједнички простор бивају смештани и примарни слогови и прекорачиоци
      - **датотеке с две зоне**
        - садржане зоне
          - примарна зона – намењена смештању примарних слогова
          - зона прекорачења – намењена смештању прекорачилаца
        - две зоне могу бити формиране као две датотеке

# Статичке расуте датотеке пробабилистичке методе

- Пројектовање датотеке (Mogin, 2008)
  - пројектантски задаци – смештање прекорачилаца
    - поступци за смештање прекорачилаца
      - случај 1 – датотека с јединственим адресним простором
        - отворено адресирање
          - линеарно тражење локације за смештај прекорачилаца
          - случајно тражење локације за смештај прекорачилаца
        - спрезање прекорачилаца у јединственом адресном простору
      - случај 2 – датотека с две зоне (примарном зоном и зоном прекорачења)
        - спрезање прекорачилаца у зони прекорачења
        - смештања прекорачилаца у серијску зону прекорачења

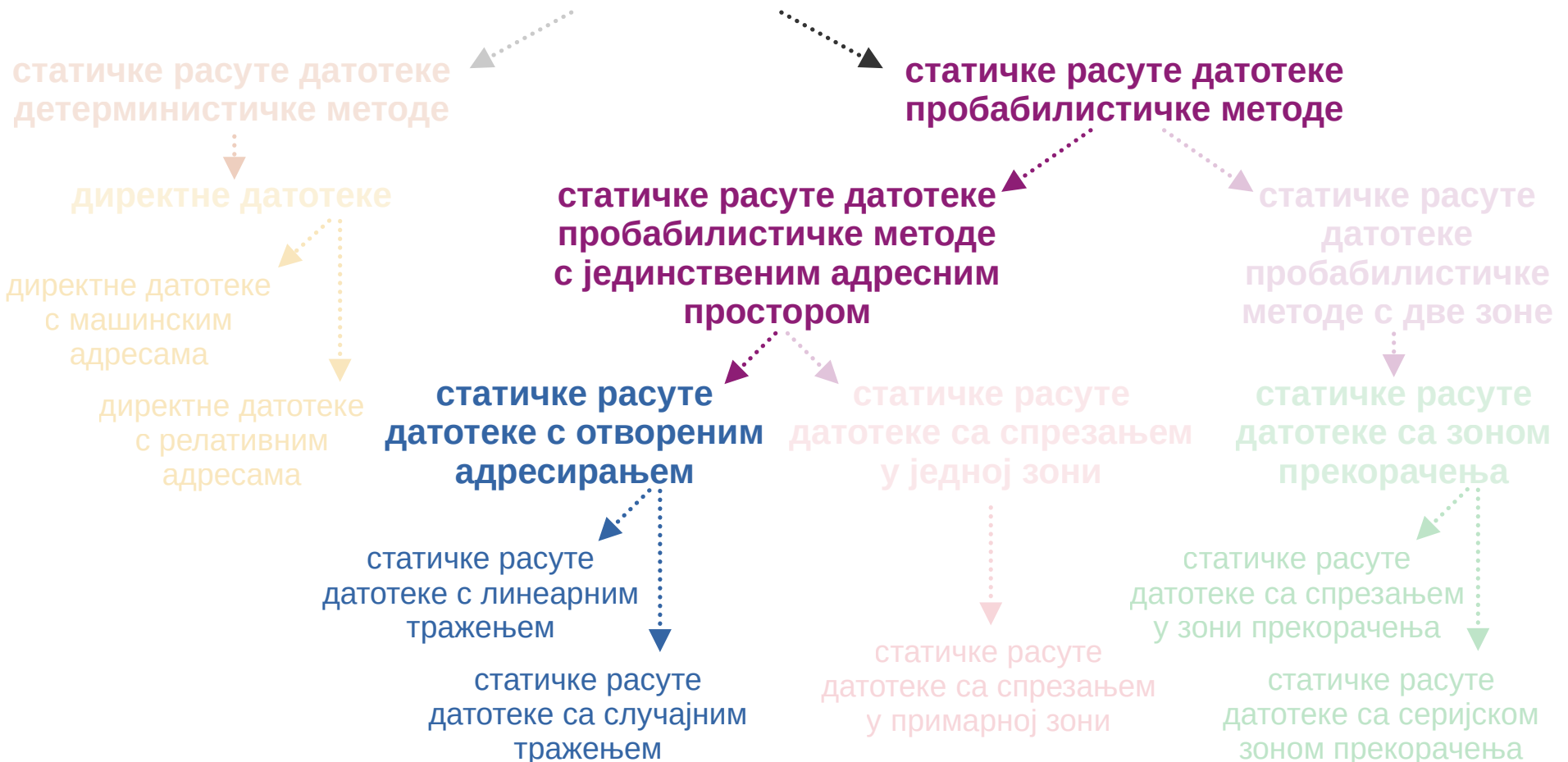
# Садржај

- Увод
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- **Статичке расуте датотеке с отвореним адресирањем**
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- Ресурси

# Статичке расуте датотеке с отвореним адресирањем

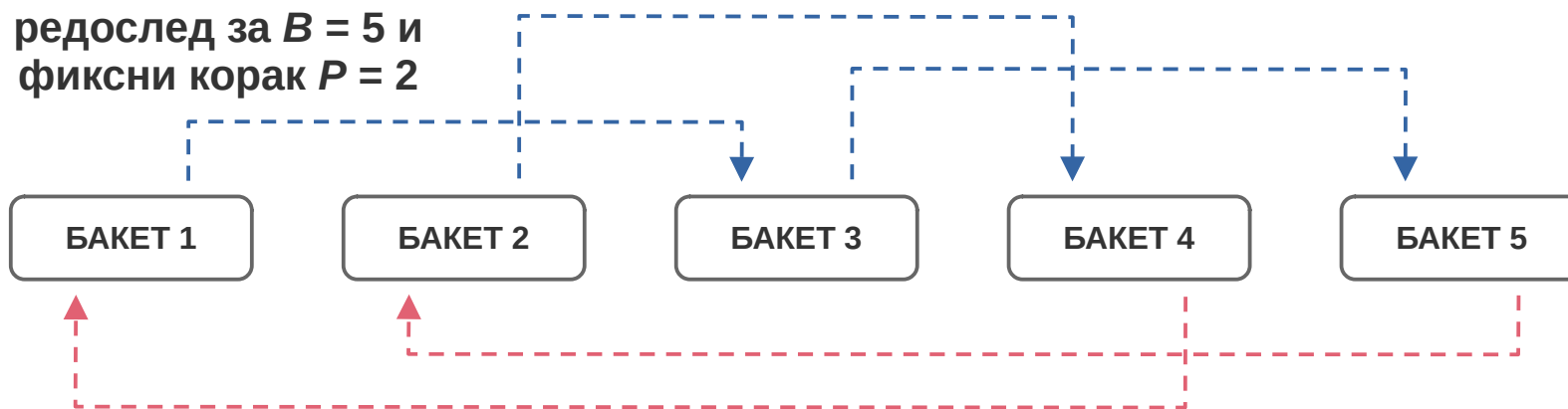
- Врсте статичких расутих датотека – примери (Mogin, 2008)

## статичке расуте датотеке



# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с отвореним адресирањем (Mogin, 2008)
  - сви слогови датотеке су у јединственом адресном простору
  - начин смештања прекорачилаца
    - тражење места за смештај прекорачиоца одвија се по нематичним бакетима применом корака (помака) тражења  $P$ 
      - општи редослед проверавања нематичних бакета одговара редоследу који настаје узастопном применом корака  $P$  на позицију бакета, при чему примена корака креће од матичног бакета и не иде преко поновног доласка до матичног бакета



# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с отвореним адресирањем (Mogin, 2008)
  - врсте статичких расутих датотека с отвореним адресирањем
    - статичке расуте датотеке с линеарним тражењем
      - примена линеарног тражења локације за смештај прекорачилаца
        - корак тражења  $P$  је фиксан на нивоу датотеке ( $P = 1$  или  $P > 1$ )
    - статичке расуте датотеке са случајним тражењем
      - примена случајног тражења локације за смештај прекорачилаца
        - корак тражења  $P$  зависи од вредности кључа

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - сви слогови датотеке су у јединственом адресном простору
  - смештање прекорачилаца се заснива на линеарном тражењу локације
    - тражење места за смештај прекорачиоца одвија се линеарно с фиксним кораком  $P$  по нематичним бакетима

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - додела локација слоговима
    - општи поступак
      - ако у матичном бакету постоји слободна локација, слог бива смештен тамо
      - ако у матичном бакету не постоји слободна локација, отпочиње линеарни пролазак кроз остале бакете у потрази за слободном локацијом за смештај слога
        - редослед проласка кроз остале бакете зависи од задатог фиксног корака (помака)  $P$ 
          - две главне могућности за подешавање корака  $P$ 
            - корак  $P$  дужине 1 ( $P = 1$ )
            - корак  $P$  дужине веће од 1 ( $P > 1$ )
        - ако у неком од нематичних бакета постоји слободна локација, слог бива смештен тамо
        - ако нема слободне локације, слог не може бити смештен

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - додела локација слоговима
    - поступак – варијанта за корак  $P$  дужине 1 ( $P = 1$ )
      - одређивање одговарајуће адресе матичног бакета  $A_0$  за слог  $S$   
 $A_0^r = h(k(S))$
      - ако у матичном бакету постоји слободна локација, слог бива смештен у прву слободну локацију у матичном бакету
      - ако у матичном бакету не постоји слободна локација, редом бивају проверавани остали бакети на адресама  $A_1^r, \dots, A_{B-1}^r$   
 $A_n^r = 1 + A_{n-1}^r \pmod{B}, n = 1, \dots, B-1$ 
        - у први од преосталих бакета у којем постоји слободна локација слог бива смештен
          - у прву слободну локацију тог бакета
        - ако нигде не може бити пронађена слободна локација, слог не може бити смештен

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - додела локација слоговима
    - поступак – варијанта за корак  $P$  дужине веће од 1 ( $P > 1$ )
      - слично као поступак за корак  $P = 1$  осим што се на другачији начин одређује редослед провере нематичних бакета
        - ако у матичном бакету не постоји слободна локација, редом бивају проверавани остали бакети на адресама  $A_1^r, \dots, A_{B-1}^r$ 
$$A_n^r = 1 + (P - 1 + A_{n-1}^r) \pmod{B}, \quad n = 1, \dots, B - 1$$
      - избор корака  $P$ 
        - корак  $P$  је природан број релативно прост у односу на број бакета  $B$ 
          - на тај начин могуће је извршити проверу свих преосталих бакета

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

  - пример 1

- датотека је бакетирана
  - фактор бакетирања  $b = 5$
- постоје укупно  $B = 3$  бакета
  - бакети  $A_1, A_2, A_3$
- постоји  $N = 13$  слогова
  - 13 обичних слогова
    - слогови  $S_1, \dots, S_{13}$
- структура слога
  - вредност кључа (цео број)
  - вредности некључних обележја
    - $n(S_i)$ ,  $i$  је ознака слога
- постоје 2 слободне локације
  - специјална ознака у пољу кључа (\*)

$$P=1$$

$$A_0^r = 1 + k(S) \pmod{3}$$

$$A_1$$

6	3	44	17	21
$n(S_5)$	$n(S_7)$	$n(S_9)$	$n(S_{10})$	$n(S_{11})$

$$A_2$$

13	36	15	*	*
$n(S_8)$	$n(S_{12})$	$n(S_{13})$		

$$A_3$$

14	8	5	11	26
$n(S_1)$	$n(S_2)$	$n(S_3)$	$n(S_4)$	$n(S_6)$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

- пример 1

- постоје 3 скупа синонима
  - $\{S_5, S_7, S_{11}, S_{12}, S_{13}\}$
  - $\{S_8\}$
  - $\{S_1, S_2, S_3, S_4, S_6, S_9, S_{10}\}$
- постоји 9 примарних слогова
  - $S_5, S_7, S_{11}, S_8, S_1, S_2, S_3, S_4, S_6$
- постоје 4 слога прекорачиоца
  - $S_9, S_{10}, S_{12}, S_{13}$

$$P=1$$

$$A_0^r = 1 + k(S) \pmod{3}$$

$A_1$	6	3	44	17	21
	$n(S_5)$	$n(S_7)$	$n(S_9)$	$n(S_{10})$	$n(S_{11})$
$A_2$	13	36	15	*	*
	$n(S_8)$	$n(S_{12})$	$n(S_{13})$		
$A_3$	14	8	5	11	26
	$n(S_1)$	$n(S_2)$	$n(S_3)$	$n(S_4)$	$n(S_6)$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

  - пример 2

- датотека је бакетирана
  - фактор бакетирања  $b = 5$
- постоје укупно  $B = 3$  бакета
  - бакети  $A_1, A_2, A_3$
- постоји  $N = 13$  слогова
  - 13 обичних слогова
    - слогови  $S_1, \dots, S_{13}$
- структура слога
  - вредност кључа (цео број)
  - вредности некључних обележја
    - $n(S_i)$ ,  $i$  је ознака слога
- постоје 2 слободне локације
  - специјална ознака у пољу кључа (\*)

$$P=2$$

$$A_0^r = 1 + k(S) \pmod{3}$$

$$A_1$$

6	3	21	36	15
$n(S_5)$	$n(S_7)$	$n(S_{11})$	$n(S_{12})$	$n(S_{13})$

$$A_2$$

13	44	17	*	*
$n(S_8)$	$n(S_9)$	$n(S_{10})$		

$$A_3$$

14	8	5	11	26
$n(S_1)$	$n(S_2)$	$n(S_3)$	$n(S_4)$	$n(S_6)$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

- пример 2

- постоје 3 скупа синонима
  - $\{S_5, S_7, S_{11}, S_{12}, S_{13}\}$
  - $\{S_8\}$
  - $\{S_1, S_2, S_3, S_4, S_6, S_9, S_{10}\}$
- постоји 11 примарних слогова
  - $S_5, S_7, S_{11}, S_{12}, S_{13}, S_8, S_1, S_2, S_3, S_4, S_6$
- постоје 2 слога прекорачиоца
  - $S_9, S_{10}$

$$P=2$$

$$A_0^r = 1 + k(S) \pmod{3}$$

$A_1$	6	3	21	36	15
	$n(S_5)$	$n(S_7)$	$n(S_{11})$	$n(S_{12})$	$n(S_{13})$
$A_2$	13	44	17	*	*
	$n(S_8)$	$n(S_9)$	$n(S_{10})$		
$A_3$	14	8	5	11	26
	$n(S_1)$	$n(S_2)$	$n(S_3)$	$n(S_4)$	$n(S_6)$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - формирање
    - формирање на основу садржаја непразне улазне серијске датотеке
    - главни начини формирања
      - формирање у једном пролазу
      - формирање у два пролаза

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - формирање – формирање у једном пролазу
    - кораци
      - формирање празне расуте датотеке ( $D_{rasu}$ )
        - упис  $B$  блокова у расуту датотеку
        - у сваком блоку свака локација је слободна
          - специјална ознака у пољу кључа локације указује на то да је локација слободна (нпр. ознака  $*$ )
      - слогови непразне улазне серијске датотеке ( $D_{ulaz}$ ) редом бивају читани и смештани у расуту датотеку
        - упису прочитаног слога претходи неуспешно тражење у расутој датотеци
        - прочитани слог добија локацију у расутој датотеци у складу с општим поступком доделе локација слоговима
        - поступак бива раније завршен ако понестане слободног места у расутој датотеци

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - формирање – формирање у једном пролазу
    - преглед



# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - формирање – формирање у једном пролазу
    - пример

## СЕРИЈСКА ДАТОТЕКА

$$A_1 \begin{array}{|c|c|c|c|} \hline 11 & 4 & 42 & 21 \\ \hline n(S_1) & n(S_2) & n(S_3) & n(S_4) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|} \hline 3 & 50 & 25 & 2 \\ \hline n(S_5) & n(S_6) & n(S_7) & n(S_8) \\ \hline \end{array}$$

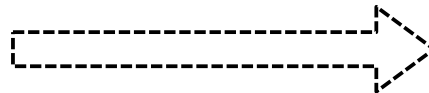
$$A_3 \begin{array}{|c|c|c|c|} \hline 16 & 46 & 14 & 37 \\ \hline n(S_9) & n(S_{10}) & n(S_{11}) & n(S_{12}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|} \hline 1 & 41 & 6 & 49 \\ \hline n(S_{13}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|} \hline 34 & 54 & 23 & 22 \\ \hline n(S_{17}) & n(S_{18}) & n(S_{19}) & n(S_{20}) \\ \hline \end{array}$$

$$A_6 \begin{array}{|c|c|c|c|} \hline 17 & 10 & 15 & * \\ \hline n(S_{21}) & n(S_{22}) & n(S_{23}) & \\ \hline \end{array}$$

формирање



## РАСУТА ДАТОТЕКА

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 22 & 17 & 10 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 34 & 54 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{17}) & n(S_{18}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 15 & * & * \\ \hline n(S_5) & n(S_{19}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 41 & 6 & 49 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - формирање – формирање у једном пролазу
    - пример

## СЕРИЈСКА ДАТОТЕКА

$$A_1 \begin{array}{|c|c|c|c|} \hline 11 & 4 & 42 & 21 \\ \hline n(S_1) & n(S_2) & n(S_3) & n(S_4) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|} \hline 3 & 50 & 25 & 2 \\ \hline n(S_5) & n(S_6) & n(S_7) & n(S_8) \\ \hline \end{array}$$

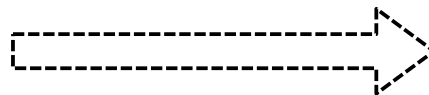
$$A_3 \begin{array}{|c|c|c|c|} \hline 16 & 46 & 14 & 37 \\ \hline n(S_9) & n(S_{10}) & n(S_{11}) & n(S_{12}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|} \hline 1 & 41 & 6 & 49 \\ \hline n(S_{13}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|} \hline 34 & 54 & 23 & 22 \\ \hline n(S_{17}) & n(S_{18}) & n(S_{19}) & n(S_{20}) \\ \hline \end{array}$$

$$A_6 \begin{array}{|c|c|c|c|} \hline 17 & 10 & 15 & * \\ \hline n(S_{21}) & n(S_{22}) & n(S_{23}) & \\ \hline \end{array}$$

формирање



број слогова прекорачилаца: ?

## РАСУТА ДАТОТЕКА

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 22 & 17 & 10 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 34 & 54 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{17}) & n(S_{18}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 15 & * & * \\ \hline n(S_5) & n(S_{19}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 41 & 6 & 49 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - формирање – формирање у једном пролазу
    - пример

## СЕРИЈСКА ДАТОТЕКА

$$A_1 \begin{array}{|c|c|c|c|} \hline 11 & 4 & 42 & 21 \\ \hline n(S_1) & n(S_2) & n(S_3) & n(S_4) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|} \hline 3 & 50 & 25 & 2 \\ \hline n(S_5) & n(S_6) & n(S_7) & n(S_8) \\ \hline \end{array}$$

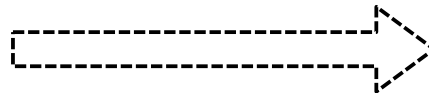
$$A_3 \begin{array}{|c|c|c|c|} \hline 16 & 46 & 14 & 37 \\ \hline n(S_9) & n(S_{10}) & n(S_{11}) & n(S_{12}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|} \hline 1 & 41 & 6 & 49 \\ \hline n(S_{13}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|} \hline 34 & 54 & 23 & 22 \\ \hline n(S_{17}) & n(S_{18}) & n(S_{19}) & n(S_{20}) \\ \hline \end{array}$$

$$A_6 \begin{array}{|c|c|c|c|} \hline 17 & 10 & 15 & * \\ \hline n(S_{21}) & n(S_{22}) & n(S_{23}) & \\ \hline \end{array}$$

формирање



број слогова прекорачилаца: 7

- $S_{20}, S_{21}, S_{17}, S_{18}, S_{23}, S_{14}, S_{15}$

## РАСУТА ДАТОТЕКА

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 22 & 17 & 10 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 34 & 54 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{17}) & n(S_{18}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 15 & * & * \\ \hline n(S_5) & n(S_{19}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 41 & 6 & 49 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

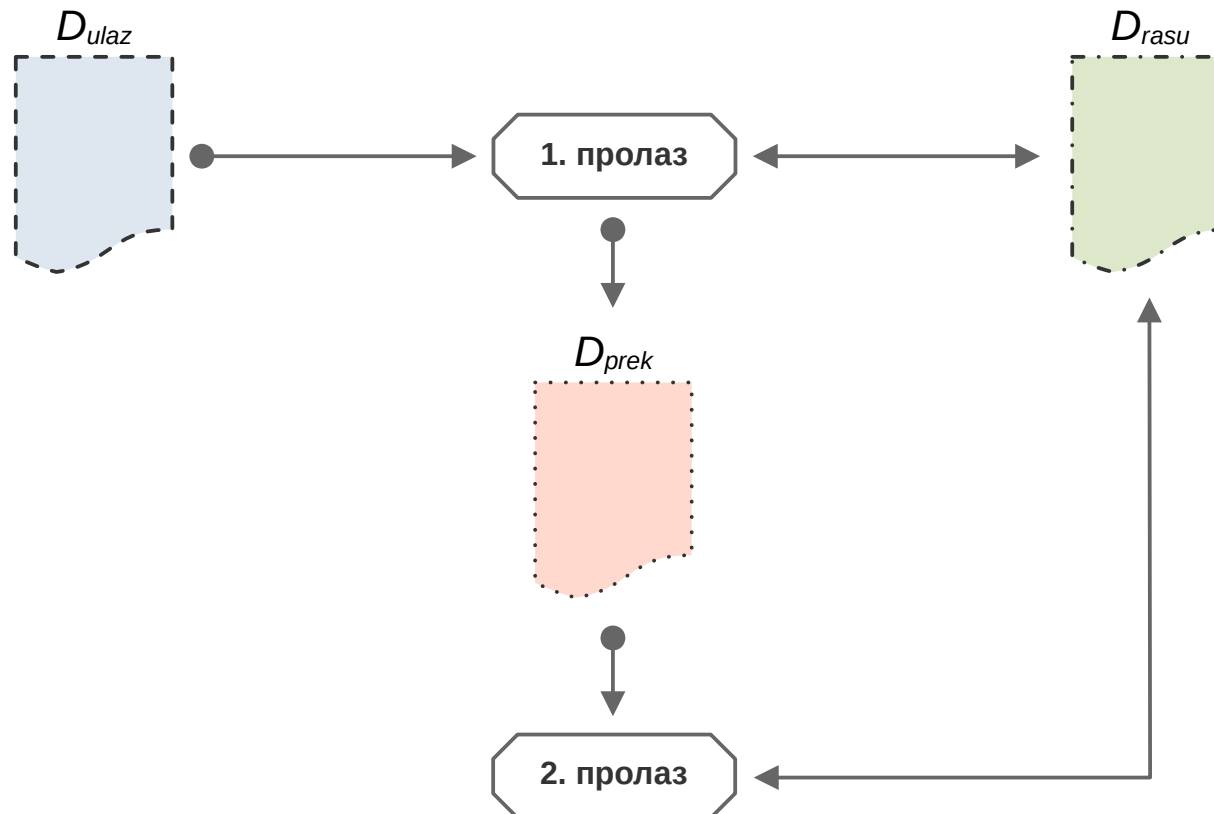
$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - формирање – формирање у два пролаза
    - кораци
      - први пролаз
        - слогови улазне датотеке ( $D_{ulaz}$ ) редом бивају читани
        - ако за прочитани слог постоји у расутој датотеци ( $D_{rasu}$ ) слободно место у матичном бакету, слог бива тамо смештен
        - ако за прочитани слог не постоји у расутој датотеци слободно место у матичном бакету, слог бива смештен у датотеку прекорачилаца ( $D_{prek}$ )
      - други пролаз
        - слогови датотеке прекорачилаца редом бивају читани и смештани у расуту датотеку
    - мотивација
      - покушај смањења појаве прекорачилаца током формирања

# Статичке расуте датотеке с отвореним адресирањем

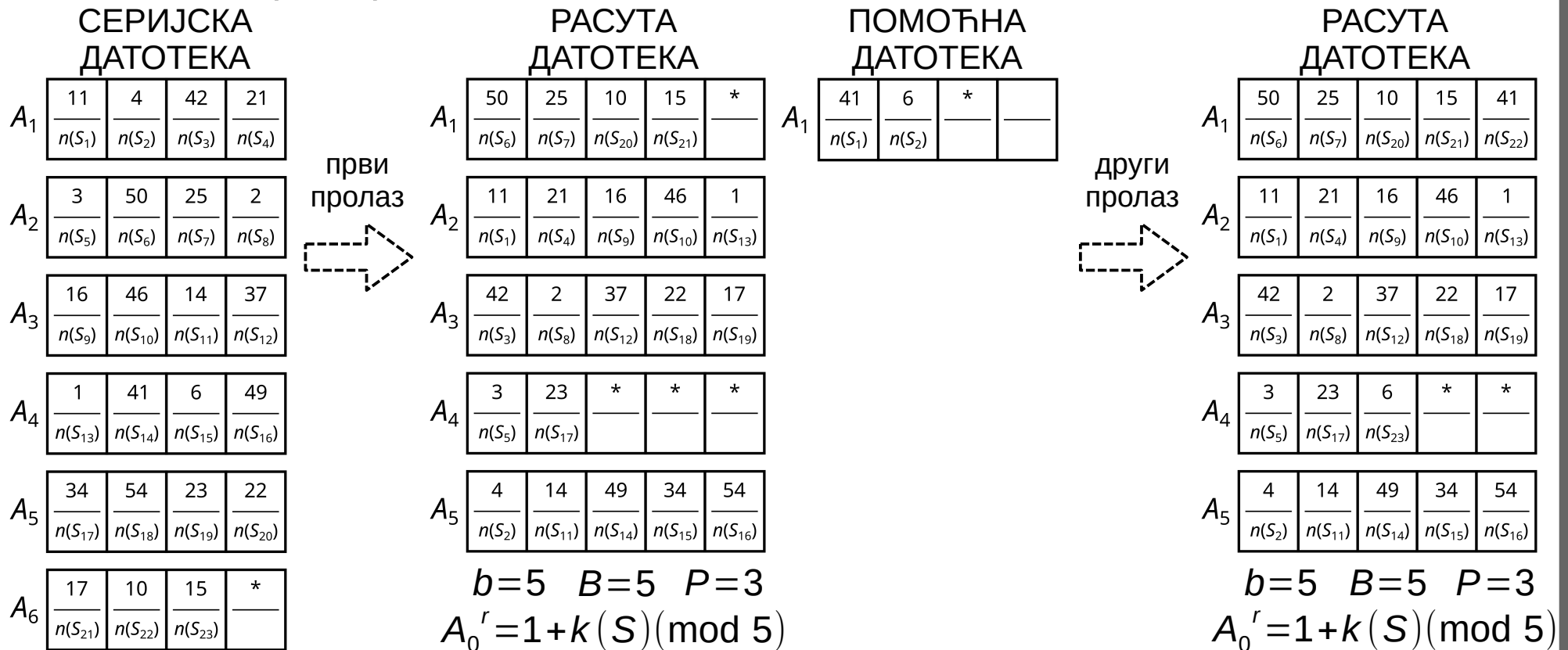
- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - формирање – формирање у два пролаза
    - преглед



# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - формирање – формирање у два пролаза

○ пример



# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - формирање – формирање у два пролаза

○ пример

број слогова прекорачилаца: 2

- $S_{22}, S_{23}$

**СЕРИЈСКА ДАТОТЕКА**

$A_1$	$\frac{11}{n(S_1)}$	$\frac{4}{n(S_2)}$	$\frac{42}{n(S_3)}$	$\frac{21}{n(S_4)}$
$A_2$	$\frac{3}{n(S_5)}$	$\frac{50}{n(S_6)}$	$\frac{25}{n(S_7)}$	$\frac{2}{n(S_8)}$
$A_3$	$\frac{16}{n(S_9)}$	$\frac{46}{n(S_{10})}$	$\frac{14}{n(S_{11})}$	$\frac{37}{n(S_{12})}$
$A_4$	$\frac{1}{n(S_{13})}$	$\frac{41}{n(S_{14})}$	$\frac{6}{n(S_{15})}$	$\frac{49}{n(S_{16})}$
$A_5$	$\frac{34}{n(S_{17})}$	$\frac{54}{n(S_{18})}$	$\frac{23}{n(S_{19})}$	$\frac{22}{n(S_{20})}$
$A_6$	$\frac{17}{n(S_{21})}$	$\frac{10}{n(S_{22})}$	$\frac{15}{n(S_{23})}$	$\frac{*}{n(S_{24})}$

први пролаз



**РАСУТА ДАТОТЕКА**

$A_1$	$\frac{50}{n(S_6)}$	$\frac{25}{n(S_7)}$	$\frac{10}{n(S_{20})}$	$\frac{15}{n(S_{21})}$	$\frac{*}{n(S_{24})}$
$A_2$	$\frac{11}{n(S_1)}$	$\frac{21}{n(S_4)}$	$\frac{16}{n(S_9)}$	$\frac{46}{n(S_{10})}$	$\frac{1}{n(S_{13})}$
$A_3$	$\frac{42}{n(S_3)}$	$\frac{2}{n(S_8)}$	$\frac{37}{n(S_{12})}$	$\frac{22}{n(S_{18})}$	$\frac{17}{n(S_{19})}$
$A_4$	$\frac{3}{n(S_5)}$	$\frac{23}{n(S_{17})}$	$\frac{*}{n(S_{22})}$	$\frac{*}{n(S_{23})}$	$\frac{*}{n(S_{24})}$
$A_5$	$\frac{4}{n(S_2)}$	$\frac{14}{n(S_{11})}$	$\frac{49}{n(S_{14})}$	$\frac{34}{n(S_{15})}$	$\frac{54}{n(S_{16})}$

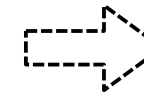
$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

**ПОМОЋНА ДАТОТЕКА**

$A_1$	$\frac{41}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{*}{n(S_3)}$	$\frac{*}{n(S_4)}$
-------	---------------------	--------------------	--------------------	--------------------

други пролаз



**РАСУТА ДАТОТЕКА**

$A_1$	$\frac{50}{n(S_6)}$	$\frac{25}{n(S_7)}$	$\frac{10}{n(S_{20})}$	$\frac{15}{n(S_{21})}$	$\frac{41}{n(S_{22})}$
$A_2$	$\frac{11}{n(S_1)}$	$\frac{21}{n(S_4)}$	$\frac{16}{n(S_9)}$	$\frac{46}{n(S_{10})}$	$\frac{1}{n(S_{13})}$
$A_3$	$\frac{42}{n(S_3)}$	$\frac{2}{n(S_8)}$	$\frac{37}{n(S_{12})}$	$\frac{22}{n(S_{18})}$	$\frac{17}{n(S_{19})}$
$A_4$	$\frac{3}{n(S_5)}$	$\frac{23}{n(S_{17})}$	$\frac{6}{n(S_{23})}$	$\frac{*}{n(S_{24})}$	$\frac{*}{n(S_{25})}$
$A_5$	$\frac{4}{n(S_2)}$	$\frac{14}{n(S_{11})}$	$\frac{49}{n(S_{14})}$	$\frac{34}{n(S_{15})}$	$\frac{54}{n(S_{16})}$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)

- тражење

- примена методе трансформације аргумента у адресу

- 1) тражење креће од матичног бакета, чијој адреси одговара резултат трансформације

$$A_0^r = h(k(S))$$

- ако тражени слог јесте пронађен у матичном бакету, тражење се завршава успешно
- ако тражени слог није пронађен у матичном бакету и матични бакет садржи бар једну слободну локацију, тражење се завршава неуспешно

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)

- тражење

- примена методе трансформације аргумента у адресу

- 2) ако тражени слог није у матичном бакету и матични бакет нема ниједну слободну локацију, тражење се даље одвија линеарно по бакетима у одређеном редоследу

$$A_n^r = 1 + (P - 1 + A_{n-1}^r) \pmod{B}, \quad n = 1, \dots, B - 1$$

- ако тражени слог јесте пронађен у датом бакету, тражење се завршава успешно
- ако тражени слог није пронађен у датом бакету и дати бакет садржи бар једну слободну локацију, тражење се завршава неуспешно

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - тражење
    - примена методе трансформације аргумента у адресу
      - 3) ако тражени слог није пронађен ни у једном бакету, тражење се завршава неуспешно

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

- тражење – пример

- тражење слога с вредношћу кључа  $k = 37$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 41$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 18$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 6$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 19$ 
      - исход: ?
      - број приступа: ?

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$
$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$
$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$
$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & * & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & & \\ \hline \end{array}$$
$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$
$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

- тражење – пример

- тражење слога с вредношћу кључа  $k = 37$ 
      - исход: *успешно*
      - број приступа: 1
    - тражење слога с вредношћу кључа  $k = 41$ 
      - исход: *успешно*
      - број приступа: 4
    - тражење слога с вредношћу кључа  $k = 18$ 
      - исход: *неуспешно*
      - број приступа: 1
    - тражење слога с вредношћу кључа  $k = 6$ 
      - исход: *успешно*
      - број приступа: 5
    - тражење слога с вредношћу кључа  $k = 19$ 
      - исход: *неуспешно*
      - број приступа: 4

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & * & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)

- тражење – перформансе

- број приступа за успешно тражење  $R_u$

$$1 \leq R_u \leq 1 + \lceil l/b \rceil$$

- $l$  – дужина најдужег низа узастопних заузетих локација што почиње од прве локације унутар низа нематичних бакета, при чему су ти бакети у поретку који одговара редоследу њиховог обиласка приликом опште примене тражења од матичног бакета с кораком  $P$

- број приступа за неуспешно тражење  $R_n$

$$R_n = \begin{cases} 1, & m < b \\ 1 + \lceil l/b \rceil, & m = b \end{cases}$$

- $m$  – број слогова у одговарајућем матичном бакету

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање
    - упис
      - нови слог бива уписан у локацију на којој се неуспешно тражење зауставило, под условом да у датотеци има слободног места
      - нови слог не може бити уписан ако не постоји ниједна слободна локација у датотеци

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - ажурирање
    - упис – пример

## СТАРО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

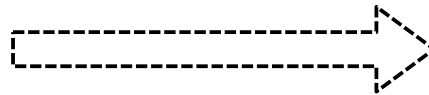
$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & * & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

упис слога с вредношћу  
кључа  $k = 51$



## НОВО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & 51 & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & n(S_{24}) & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - ажурирање
    - упис – пример

## СТАРО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

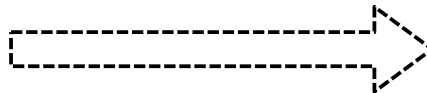
$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & * & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

упис слога с вредношћу  
кључа  $k = 51$



исход: успешно

број приступа: ?

## НОВО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & 51 & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & n(S_{24}) & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем

- ажурирање

- упис – пример

## СТАРО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

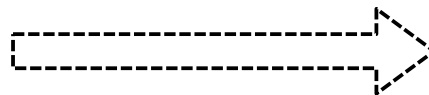
$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & * & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

упис слога с вредношћу  
кључа  $k = 51$



исход: успешно

број приступа: 6

- 5 за неуспешно тражење
- 1 за упис у локацију

## НОВО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 50 & 25 & 10 & 15 & 41 \\ \hline n(S_6) & n(S_7) & n(S_{20}) & n(S_{21}) & n(S_{22}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 11 & 21 & 16 & 46 & 1 \\ \hline n(S_1) & n(S_4) & n(S_9) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 42 & 2 & 37 & 22 & 17 \\ \hline n(S_3) & n(S_8) & n(S_{12}) & n(S_{18}) & n(S_{19}) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|c|c|} \hline 3 & 23 & 6 & 51 & * \\ \hline n(S_5) & n(S_{17}) & n(S_{23}) & n(S_{24}) & \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|c|c|} \hline 4 & 14 & 49 & 34 & 54 \\ \hline n(S_2) & n(S_{11}) & n(S_{14}) & n(S_{15}) & n(S_{16}) \\ \hline \end{array}$$

$$b=5 \quad B=5 \quad P=3$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање
    - модификација
      - као у случају модификације за серијску или секвенцијалну организацију, али уз ослањање на тражење које се користи у статичкој расутој датотеци с линеарним тражењем

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање
    - логичко брисање
      - потребно увођење поља статуса у структуру слога
        - пример могућих ознака у пољу статуса
          - А – актуелан слог
          - О – логички обрисан слог (неактуелан слог)
          - S – слободна локација
        - потребно прилогађење поступка тражења
          - при тражењу не долази до заустављања у случају наиласка на локацију која садржи логички обрисан слог
      - након успешног проналажења слога за брисање, статус у локацији тог слога бива промењен на логички обрисан слог

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање
    - физичко брисање
      - 1) након успешног тражења слога за брисање, потребно је прво проверити и потенцијално променити садржај бакета тог слога
        - долази до померања садржаја унутар бакета слога за брисање, под условом да слог за брисање није у последњој локацији
          - садржаји који су у опсегу од прве локације иза локације слога за брисање до последње локације у бакету бивају померени за једну локацију налево
          - локација која је пре почетка померања била слободна остаје слободна
        - ако је у бакету слога за брисање било слободног места пре почетка брисања, није потребно проверити остале бакете и брисање је завршено

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање
    - физичко брисање
      - 2) ако у бакету слога за брисање није било слободног места пре почетка брисања, у последњу локацију тог бакета може бити премештен погодан слог из неког од осталих бакета
        - ради налажења слога за премештање, остали бакети бивају проверени у одређеном редоследу
          - редослед обиласка бакета при тражењу које креће од бакета слога за брисање с кораком  $P$  и не иде даље од првог бакета са слободним местом, ако таквог бакета има
        - слог за премештање је први слог који може бити премештен тако да смештањем у нову локацију не буде у бакету који по примењеном редоследу претходи матичном бакету тог слога
        - ако слог за премештање није пронађен, последња локација бакета слога за брисање је слободна и брисање је завршено

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање
    - физичко брисање
      - 3) ако је пронађен слог за премештање у неком од осталих бакета, долази до одређених промена у датотеци
        - слог за премештање бива смештен у последњу локацију бакета слога за брисање и уклоњен из бакета где је пронађен
          - уклањање из бакета бива извршено слично као за слог за брисање, заједно с потенцијалним попуњавањем последње локације тог бакета
            - потенцијално проналажење новог слога за премештање који може попити последњу локацију бакета већ премештеног слога одвија се у остатку осталих бакета
            - уклањање може довести до ланчаног премештања слогова из све мањег остатка осталих бакета

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - ажурирање
    - физичко брисање – пример

СТАРО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|} \hline 10 & 20 & * \\ \hline n(S_5) & n(S_9) & \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|} \hline 11 & 6 & 26 \\ \hline n(S_2) & n(S_7) & n(S_{11}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|} \hline 17 & 7 & 22 \\ \hline n(S_3) & n(S_6) & n(S_8) \\ \hline \end{array}$$

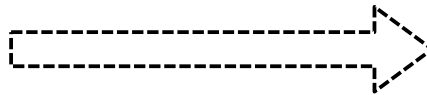
$$A_4 \begin{array}{|c|c|c|} \hline 18 & 2 & 16 \\ \hline n(S_4) & n(S_{10}) & n(S_{12}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|} \hline 14 & 8 & 19 \\ \hline n(S_1) & n(S_{13}) & n(S_{14}) \\ \hline \end{array}$$

$$b=3 \quad B=5 \quad P=1$$

$$A_0^r = 1 + k(S) \pmod{5}$$

физичко брисање слога с вредношћу кључа  $k = 6$



НОВО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|} \hline 10 & 20 & * \\ \hline n(S_5) & n(S_9) & \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|} \hline 11 & 26 & 16 \\ \hline n(S_2) & n(S_{11}) & n(S_{12}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|} \hline 17 & 7 & 22 \\ \hline n(S_3) & n(S_6) & n(S_8) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|} \hline 18 & 2 & 8 \\ \hline n(S_4) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|} \hline 14 & 19 & * \\ \hline n(S_1) & n(S_{14}) & \\ \hline \end{array}$$

$$b=3 \quad B=5 \quad P=1$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - ажурирање
    - физичко брисање – пример

СТАРО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|} \hline 10 & 20 & * \\ \hline n(S_5) & n(S_9) & \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|} \hline 11 & 6 & 26 \\ \hline n(S_2) & n(S_7) & n(S_{11}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|} \hline 17 & 7 & 22 \\ \hline n(S_3) & n(S_6) & n(S_8) \\ \hline \end{array}$$

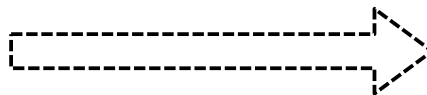
$$A_4 \begin{array}{|c|c|c|} \hline 18 & 2 & 16 \\ \hline n(S_4) & n(S_{10}) & n(S_{12}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|} \hline 14 & 8 & 19 \\ \hline n(S_1) & n(S_{13}) & n(S_{14}) \\ \hline \end{array}$$

$$b=3 \quad B=5 \quad P=1$$

$$A_0^r = 1 + k(S) \pmod{5}$$

физичко брисање слога с вредношћу кључа  $k = 6$



исход: успешно

број приступа: ?

НОВО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|} \hline 10 & 20 & * \\ \hline n(S_5) & n(S_9) & \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|} \hline 11 & 26 & 16 \\ \hline n(S_2) & n(S_{11}) & n(S_{12}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|} \hline 17 & 7 & 22 \\ \hline n(S_3) & n(S_6) & n(S_8) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|} \hline 18 & 2 & 8 \\ \hline n(S_4) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|} \hline 14 & 19 & * \\ \hline n(S_1) & n(S_{14}) & \\ \hline \end{array}$$

$$b=3 \quad B=5 \quad P=1$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем
  - ажурирање
    - физичко брисање – пример

СТАРО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|} \hline 10 & 20 & * \\ \hline n(S_5) & n(S_9) & \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|} \hline 11 & 6 & 26 \\ \hline n(S_2) & n(S_7) & n(S_{11}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|} \hline 17 & 7 & 22 \\ \hline n(S_3) & n(S_6) & n(S_8) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|} \hline 18 & 2 & 16 \\ \hline n(S_4) & n(S_{10}) & n(S_{12}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|} \hline 14 & 8 & 19 \\ \hline n(S_1) & n(S_{13}) & n(S_{14}) \\ \hline \end{array}$$

$$b=3 \quad B=5 \quad P=1$$

$$A_0^r = 1 + k(S) \pmod{5}$$

физичко брисање слога с вредношћу кључа  $k = 6$



исход: успешно

број приступа: 8

- 1 за успешно тражење
- 7 за уклањање, провере и премештања

НОВО СТАЊЕ

$$A_1 \begin{array}{|c|c|c|} \hline 10 & 20 & * \\ \hline n(S_5) & n(S_9) & \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|} \hline 11 & 26 & 16 \\ \hline n(S_2) & n(S_{11}) & n(S_{12}) \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|} \hline 17 & 7 & 22 \\ \hline n(S_3) & n(S_6) & n(S_8) \\ \hline \end{array}$$

$$A_4 \begin{array}{|c|c|c|} \hline 18 & 2 & 8 \\ \hline n(S_4) & n(S_{10}) & n(S_{13}) \\ \hline \end{array}$$

$$A_5 \begin{array}{|c|c|c|} \hline 14 & 19 & * \\ \hline n(S_1) & n(S_{14}) & \\ \hline \end{array}$$

$$b=3 \quad B=5 \quad P=1$$

$$A_0^r = 1 + k(S) \pmod{5}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - ажурирање – перформансе
    - логичко брисање
      - логичко брисање негативно делује на перформансе тражења
        - очекивано повећање броја приступа и при успешном и при неуспешном тражењу
      - негативни ефекти логичког брисања могу бити решавани повременим извршавањем реорганизације датотеке
    - физичко брисање
      - физичко брисање не делује негативно на перформансе тражења као што је случај с логичким брисањем
      - током физичког брисања практично долази до динамичке реорганизације датотеке
        - неки прекорачиоци могу бити приближени матичним бакетима
        - део заузетог простора бива ослобођен

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с линеарним тражењем (Mogin, 2008)
  - одлике
    - варијанта с кораком тражења  $P$  дужине 1 ( $P = 1$ )
      - појава нагомилавања
        - настајање дугих низова заузетих локација
      - нагомилавање негативно утиче на тражење
        - тражење све чешће постаје релативно дуготрајно
      - мера против негативног утицаја нагомилавања
        - примена фактора попуњености  $q < 0,7$
    - варијанта с кораком тражења  $P$  дужине веће од 1 ( $P > 1$ )
      - сличних општих одлика као варијанта с кораком  $P = 1$
      - примена корака  $P > 1$  делује против негативног утицаја нагомилавања
        - настају низови удаљени  $P$  бакета један од другог и мање је вероватно да дође до њиховог повезивања

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем (Mogin, 2008)
  - сви слогови датотеке су у јединственом адресном простору
  - смештање прекорачилаца се заснива на случајном тражењу локације
    - за разлику од линеарног тражења локације, корак тражења није фиксан на нивоу датотеке

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем (Mogin, 2008)
  - додела локација слоговима
    - поступак
      - модификација општег поступка који се примењује код статичке расуте датотеке с линеарним тражењем
        - корак тражења  $P$  није унапред задат већ зависи од вредности кључа
        - употребљавају се две трансформације вредности кључа
          - трансформација  $h_1$  за одређивање адресе матичног бакета
            - интервал могућих вредности  $[1, B]$
          - трансформација  $h_2$  за одређивање корака  $P$ 
            - интервал могућих вредности  $[1, B - 1]$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем (Mogin, 2008)

- додела локација слоговима

- поступак

- одређивање одговарајуће адресе матичног бакета  $A_0$  за слог  $S$   
 $A_0^r = h_1(k(S))$
- одређивање адресе  $n$ -тог бакета у односу на матични бакет

$$A_n^r = \begin{cases} A_{n-1}^r + h_2(k(S)), & A_{n-1}^r + h_2(k(S)) \leq B \\ A_{n-1}^r + h_2(k(S)) - B, & A_{n-1}^r + h_2(k(S)) > B \end{cases}$$

$$n = 1, \dots, B - 1$$

- избор трансформације  $h_2$ 
  - резултат трансформације  $h_2$  је природан број релативно прост у односу на број бакета  $B$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем (Mogin, 2008)

- додела локација слоговима

- поступак

- пример избора трансформација

- трансформација  $h_1$  за случај када је  $B$  прост број

$$h_1(k(S)) = 1 + k(S) \pmod{B}$$

- трансформација  $h_2$

$$h_2(k(S)) = 1 + k(S) \pmod{(B-1)}$$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем

- пример

- датотека је бакетирана
  - фактор бакетирања  $b = 5$
- постоје укупно  $B = 3$  бакета
  - бакети  $A_1, A_2, A_3$
- постоји  $N = 13$  слогова
  - 13 обичних слогова
    - слогови  $S_1, \dots, S_{13}$
- структура слога
  - вредност кључа (цео број)
  - вредности некључних обележја
    - $n(S_i)$ ,  $i$  је ознака слога
- постоје 2 слободне локације
  - специјална ознака у пољу кључа (\*)

$$h_1(k(S)) = 1 + k(S) \pmod{3}$$
$$h_2(k(S)) = 1 + k(S) \pmod{2}$$

$A_1$

6	3	44	21	36
$n(S_5)$	$n(S_7)$	$n(S_9)$	$n(S_{11})$	$n(S_{12})$

$A_2$

13	17	15	*	*
$n(S_8)$	$n(S_{10})$	$n(S_{13})$		

$A_3$

14	8	5	11	26
$n(S_1)$	$n(S_2)$	$n(S_3)$	$n(S_4)$	$n(S_6)$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем

- пример

- постоје 3 скупа синонима

- $\{S_5, S_7, S_{11}, S_{12}, S_{13}\}$
    - $\{S_8\}$
    - $\{S_1, S_2, S_3, S_4, S_6, S_9, S_{10}\}$

- постоји 10 примарних слогова

- $S_5, S_7, S_{11}, S_{12}, S_8, S_1, S_2, S_3, S_4, S_6$

- постоје 3 слога прекорачиоца

- $S_9, S_{10}, S_{13}$

$$h_1(k(S)) = 1 + k(S) \pmod{3}$$
$$h_2(k(S)) = 1 + k(S) \pmod{2}$$

$A_1$

6	3	44	21	36
$n(S_5)$	$n(S_7)$	$n(S_9)$	$n(S_{11})$	$n(S_{12})$

$A_2$

13	17	15	*	*
$n(S_8)$	$n(S_{10})$	$n(S_{13})$		

$A_3$

14	8	5	11	26
$n(S_1)$	$n(S_2)$	$n(S_3)$	$n(S_4)$	$n(S_6)$

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке са случајним тражењем (Mogin, 2008)
  - одлике
    - зависност корака тражења од вредности кључа
      - за потребе смештања прекорачилаца, различите секвенце бакета могу бити провераване и за прекорачиоце од истог матичног бакета
        - очекивано да распоред прекорачилаца од истог матичног бакета буде што мање концентрисан
      - успостављање зависности корака тражења од вредности кључа делује против негативног утицаја нагомилавања
        - у случају када постоји већи број прекорачилаца из неколико блиских бакета

# Статичке расуте датотеке с отвореним адресирањем

- Статичке расуте датотеке с отвореним адресирањем (Mogin, 2008)
  - одлике
    - погодност за примену када фактор попуњености  $q$  до граничне вредности 0,7
    - ако се фактор попуњености  $q$  повећава преко граничне вредности 0,7, очекивано је нагло повећање броја приступа и при успешном и посебно при неуспешном тражењу
      - узроци погоршања перформанси тражења
        - **У1**: тражење се може одвијати и у бакетима који не садрже слоге из скупа синонима траженог слога
        - **У2**: неуспешно тражење бива завршено тек по наиласку на слободну локацију
        - **У3**: постојање прекорачилаца једног бакета може довести до појаве прекорачилаца неких других бакета

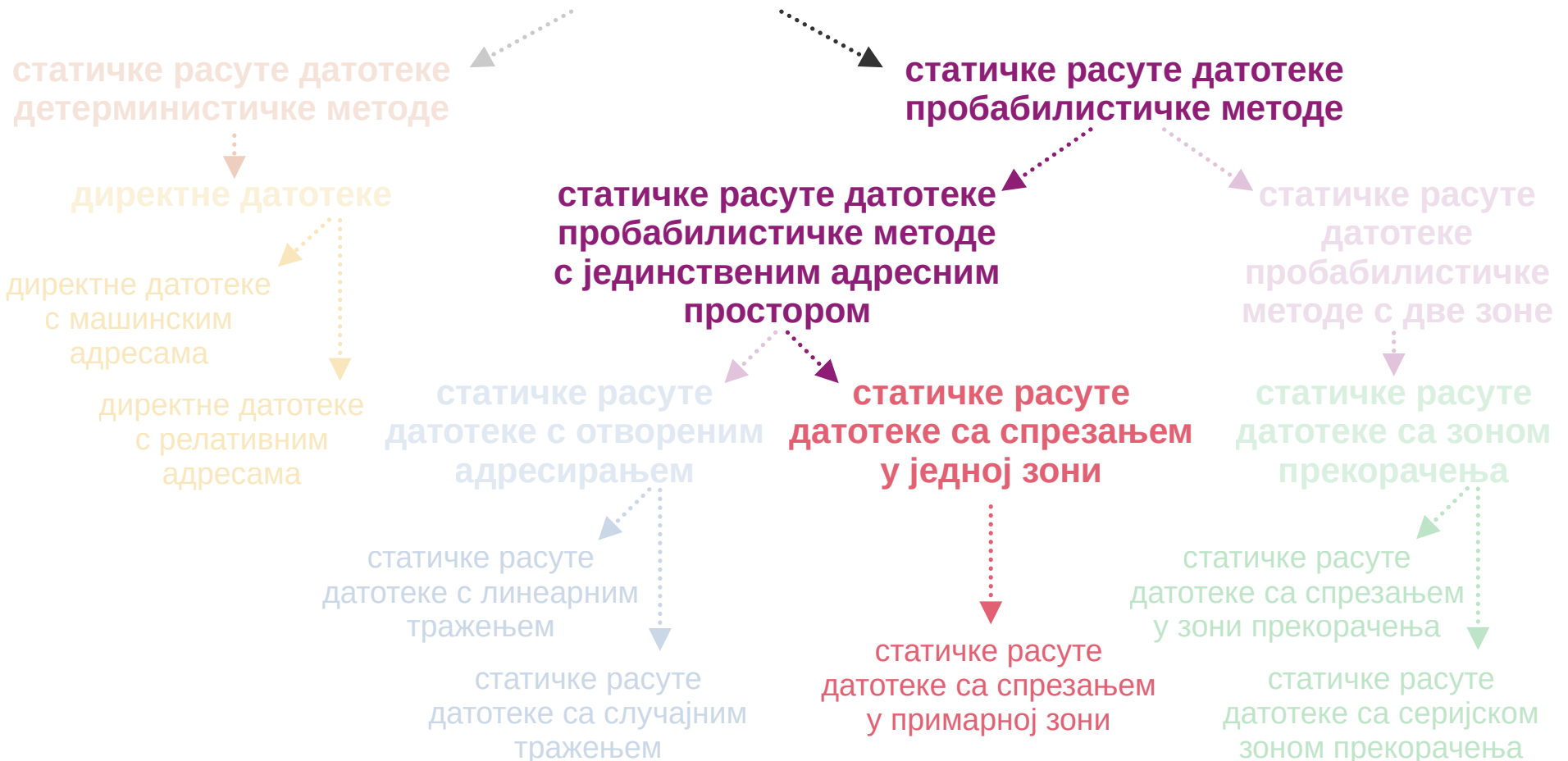
# Садржај

- Увод
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- **Статичке расуте датотеке са спрезањем у једној зони**
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- Ресурси

# Статичке расуте датотеке са спрезањем у једној зони

- Врсте статичких расутих датотека – примери (Mogin, 2008)

## статичке расуте датотеке



# Статичке расуте датотеке са спрезањем у једној зони

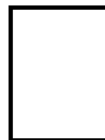
- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - сви слогови датотеке су у јединственом адресном простору
    - сваки скуп синонима је спрегнут у засебну листу
      - сваки скуп синонима је везан за један бакет
  - смештање прекорачилаца се заснива на праћењу показивача
    - тражење места за смештај прекорачиоца у листи бакета са слободним простором
      - постојање индекса слободних бакета
      - бакети са слободним простором су двоструко спрегнути у листу
    - прекорачилац бива увезан у одговарајућу листу синонима

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - структура
    - помоћна структура података
      - индекс слободних бакета  $L$ 
        - показивач на почетак двоструко спрегнуте листе бакета са слободним простором
    - $B$  бакета

пример структуре индекса  
слободних бакета

$L$



# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)

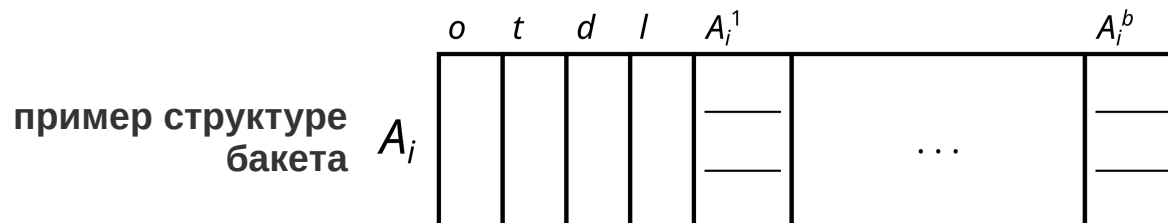
- структура – бакет

- заглавље

- поље  $o$  – показивач на почетак листе синонима везаних за бакет
- поље  $t$  – показивач на непосредно претходни бакет у листи бакета са слободним простором
- поље  $d$  – показивач на непосредно наредни бакет у листи бакета са слободним простором
- поље  $l$  – евиденција броја слободних локација у бакету

- $b$  локација за слоге

- свака локација за слог садржи и поље показивача на непосредно наредни слог из истог скупа синонима



# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - формирање
    - 1) иницијализација помоћне структуре  $L$  и  $B$  бакета
      - иницијализација показивача у структури  $L$ 
        - упис адресе првог бакета

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - формирање
    - 1) иницијализација помоћне структуре  $L$  и  $B$  бакета
      - иницијализација садржаја бакетâ
        - иницијализација празних листа синонима
          - постављање ознаке краја (нпр. ознака  $*$ ) у поља  $o$
        - двоструко спрезање бакетâ путем показивача из поља  $t$  и  $d$
        - упис вредности фактора бакетирања  $b$  у поља  $l$
        - у сваком бакету свака локација је слободна
          - специјална ознака у пољу кључа локације указује на то да је локација слободна (нпр. ознака  $*$ )
          - специјална ознака у пољу показивача локације указује на то да нема непосредно наредног слога из скупа синонима (нпр. ознака  $*$ )

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - формирање
    - 2) смештање слогова
      - ако има слободне локације у матичном бакету, слог бива смештен у матични бакет, а, ако такве локације тамо нема, слог бива смештен у први бакет из листе бакета са слободним простором
      - слог бива увезан на крај одговарајуће листе синонима
      - број слободних локација у пољу  $l$  оног бакета у који је смештен слог бива смањен за 1
        - ако број слободних локација у бакету постане 0, тај бакет бива одвезан из листе бакета са слободним простором
      - слог не може бити смештен ако нема слободног простора

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони
  - пример 1 (без слогова)

- датотека је бакетирана
  - фактор бакетирања  $b = 5$
- постоје укупно  $B = 3$  бакета
  - бакети  $A_1, A_2, A_3$
- структура слога
  - вредност кључа (цео број)
  - вредности некључних обележја
  - адреса локације наредног слога из истог скупа синонима
- постоје 3 скупа синонима
  - синоними из истог скупа спрегнути у листу
  - показивачи на почетке листâ синонима у пољима  $o$

$h(k(S)) = 1 + k(S) \pmod{3}$

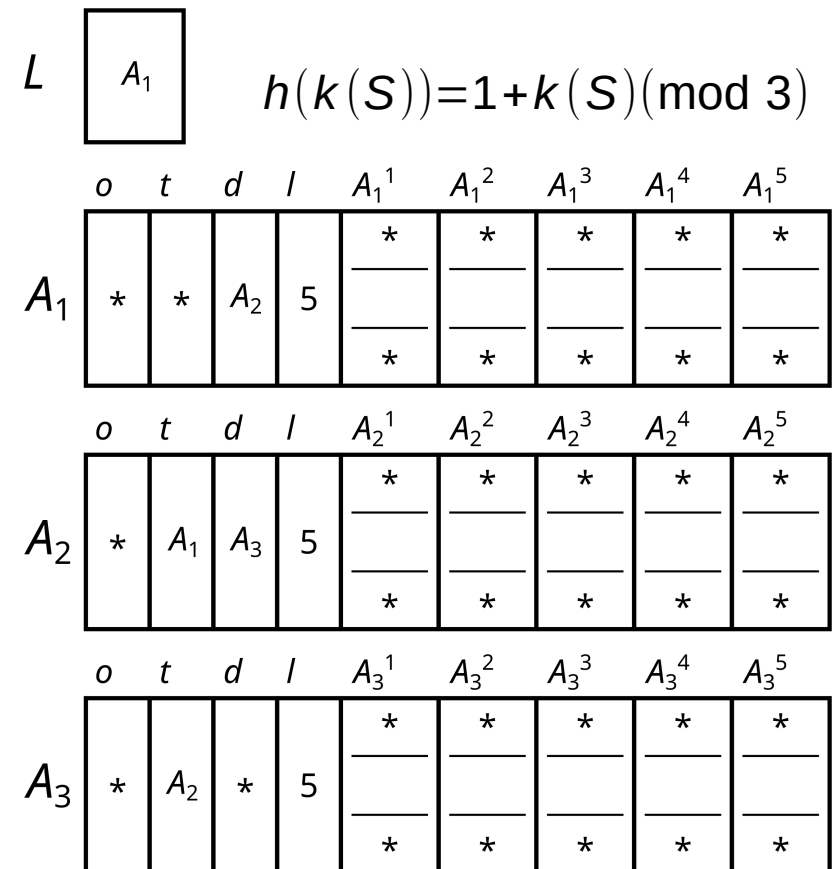
	$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$L$	$A_1$								
$A_1$									

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони

- пример 1 (без слогова)

- постоје 3 бакета са слободним простором
  - бакети са слободним простором двоструко спрегнути у листу путем показивача у пољима  $t$  и  $d$
  - показивач на почетак листе бакета са слободним простором у помоћној структури података  $L$
  - број слободних локација у бакету у пољу  $l$
- постоји 15 слободних локација
  - специјална ознака у пољу кључа (\*)



# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони

– пример 2

- постоји  $N = 12$  слогова
  - 12 обичних слогова
  - слогови  $S_1, \dots, S_{12}$
- постоји један бакет са слободним простором
  - бакет  $A_2$
  - 3 слободне локације

$b=5 \quad B=3$

$h(k(S)) = 1 + k(S) \pmod{3}$

$L$	$A_2$										
	$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$		
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$		
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$		
	$o$	$t$	$d$	$l$	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$		
$A_2$	*	*	*	3	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	*	*	*		
					*	*	*	*	*		
	$o$	$t$	$d$	$l$	$A_3^1$	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$		
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$		
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$		

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони

– пример 2

- постоје 3 скупа синонима
  - $\{S_1, S_2, S_5, S_7, S_8, S_{12}\}$
  - $\{\}$
  - $\{S_3, S_4, S_6, S_9, S_{10}, S_{11}\}$
- постоји 10 примарних слогова
  - $S_1, S_2, S_5, S_7, S_8, S_3, S_4, S_6, S_9, S_{10}$
- постоје 2 слога прекорачиоца
  - $S_{11}, S_{12}$

$b=5 \quad B=3$   
 $h(k(S))=1+k(S) \pmod{3}$

$L$		$A_2$		$h(k(S))=1+k(S) \pmod{3}$					
				$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$
$A_2$	$A_2^1$	*	*	3	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	*	*	*
					*	*	*	*	*
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - тражење
    - прво долази до приступа матичном бакету, чија адреса је добијена применом трансформације аргумента тражења
    - тражење се одвија праћењем показивача кроз листу синонима везаних за матични бакет
      - у случају наиласка на слог с траженом вредношћу кључа, тражење је успешно
      - у случају проласка кроз целокупну листу без проналажења слога с траженом вредношћу кључа, тражење је неуспешно

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони

- тражење – пример

- тражење слога с вредношћу кључа  $k = 18$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 23$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 15$ 
      - исход: ?
      - број приступа: ?
    - тражење слога с вредношћу кључа  $k = 19$ 
      - исход: ?
      - број приступа: ?

$b=5 \quad B=3$

$h(k(S)) = 1 + k(S) \pmod{3}$

	$L$	$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$A_2$						12	6	18	24	3
$A_1$	$A_1^1$	*	*	0	$\frac{n(S_1)}{A_1^2}$	$\frac{n(S_2)}{A_1^3}$	$\frac{n(S_5)}{A_1^4}$	$\frac{n(S_7)}{A_1^5}$	$\frac{n(S_8)}{A_2^2}$	
	$o$	$t$	$d$	$l$	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$	
$A_2$	*	*	*	3	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	*	*	*	
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$	
	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$					

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони

- тражење – пример

- тражење слога с вредношћу кључа  $k = 18$ 
      - исход: *успешно*
      - број приступа: 1
    - тражење слога с вредношћу кључа  $k = 23$ 
      - исход: *неуспешно*
      - број приступа: 2
    - тражење слога с вредношћу кључа  $k = 15$ 
      - исход: *успешно*
      - број приступа: 2
    - тражење слога с вредношћу кључа  $k = 19$ 
      - исход: *неуспешно*
      - број приступа: 1

$b=5 \quad B=3$   
 $h(k(S)) = 1 + k(S) \pmod{3}$

	$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$L$	$A_2$								
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$
	$o$	$t$	$d$	$l$	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$
$A_2$	*	*	*	3	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	*	*	*
					*	*	*	*	*
	$o$	$t$	$d$	$l$	$A_3^1$	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - тражење – перформансе
    - број приступа за успешно тражење  $R_u$   
 $1 \leq R_u \leq 1 + S$ 
      - $S$  – број слогова у скупу синонима ( $0 \leq S \leq N$ )
    - број приступа за неуспешно тражење  $R_n$   
 $1 \leq R_n \leq 1 + S$

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - ажурирање
    - упис
      - нови слог бива уписан након неуспешног тражења, под условом да у датотеци има слободног места
        - локација за смештај новог слога је слободна локација у матичном бакету, ако тамо има слободног места, а иначе је у првом бакету из листе бакета са слободним простором
          - могла би бити заузета прва слободна локација у бакету
        - у евиденцији броја слободних локација у бакету новог слога треба садржану вредност умањити за један
        - у случају да након смештања новог слога у бакет нема више слободних локација у том бакету, потребно је уклонити тај бакет из листе бакета са слободним простором
        - уписани нови слог бива придружен као последњи у одговарајућу листу синонима

# Статичке расуте датотеке са спрезањем у једној зони

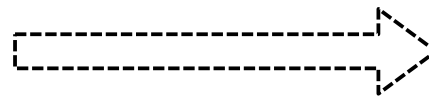
- Статичке расуте датотеке са спрезањем у примарној зони
  - ажурирање
    - упис – пример

СТАРО СТАЊЕ

$L$   $A_2$   $b=5$   $B=3$   
 $h(k(S))=1+k(S) \pmod{3}$

	<i>o</i>	<i>t</i>	<i>d</i>	<i>l</i>	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$
	<i>o</i>	<i>t</i>	<i>d</i>	<i>l</i>	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$
$A_2$	*	*	*	3	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	*	*	*
					*	*	*	*	*
	<i>o</i>	<i>t</i>	<i>d</i>	<i>l</i>	$A_3^1$	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$

упис слога с вредношћу  
кључа  $k = 29$



исход: успешно

НОВО СТАЊЕ

$L$   $A_2$   $b=5$   $B=3$   
 $h(k(S))=1+k(S) \pmod{3}$

	<i>o</i>	<i>t</i>	<i>d</i>	<i>l</i>	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$
	<i>o</i>	<i>t</i>	<i>d</i>	<i>l</i>	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$
$A_2$	*	*	*	2	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	<b>29</b> $n(S_{13})$	*	*
					$A_2^3$	*	*	*	*
	<i>o</i>	<i>t</i>	<i>d</i>	<i>l</i>	$A_3^1$	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - ажурирање
    - модификација
      - као у случају модификације за серијску или секвенцијалну организацију, али уз ослањање на тражење које се користи у статичким расутим датотекама са спрезањем у примарној зони

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - ажурирање
    - логичко брисање
      - подразумева успешно тражење и логичко уклањање слога за брисање
        - подразумева увођење поља статуса у структуру слога
          - могуће вредности статуса могу бити као у случају логичког брисања код статичких расутих датотека с линеарним тражењем
        - очекивано прилагођење поступка тражења
          - при тражењу, наилазак на локацију која садржи логички обрисан слог сам по себи не подразумева заустављање
        - статус у локацији слога за брисања бива промењен на логички обрисан слог

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - ажурирање
    - физичко брисање
      - након успешног тражења слога за брисање, тај слог бива уклоњен и бива ослобођена његова локација
        - слог за брисање бива одвезан из одговарајуће листе синонима
        - у локацију слога за брисање бива уписан садржај који означава да је локација слободна
        - у евиденцији броја слободних локација у бакету слога за брисање треба садржану вредност увећати за један
        - ако пре уклањања слога за брисање бакет тог слога није имао слободног простора, тај бакет треба увезати у листу бакета са слободним простором
          - увезивање би могло бити на почетак листе

# Статичке расуте датотеке са спрезањем у једној зони

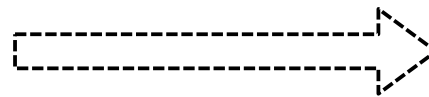
- Статичке расуте датотеке са спрезањем у примарној зони
  - ажурирање
    - физичко брисање – пример

СТАРО СТАЊЕ

$L$   $\boxed{A_2}$   $b=5$   $B=3$   
 $h(k(S))=1+k(S) \pmod{3}$

	$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$
	$o$	$t$	$d$	$l$	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$
$A_2$	*	*	*	2	$\frac{11}{n(S_{11})}$	$\frac{15}{n(S_{12})}$	$\frac{29}{n(S_{13})}$	*	*
					$A_2^3$	*	*	*	*
	$o$	$t$	$d$	$l$	$A_3^1$	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^1$

физичко брисање слога с вредношћу кључа  $k = 11$



исход: успешно

НОВО СТАЊЕ

$L$   $\boxed{A_2}$   $b=5$   $B=3$   
 $h(k(S))=1+k(S) \pmod{3}$

	$o$	$t$	$d$	$l$	$A_1^1$	$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$
$A_1$	$A_1^1$	*	*	0	$\frac{12}{n(S_1)}$	$\frac{6}{n(S_2)}$	$\frac{18}{n(S_5)}$	$\frac{24}{n(S_7)}$	$\frac{3}{n(S_8)}$
					$A_1^2$	$A_1^3$	$A_1^4$	$A_1^5$	$A_2^2$
	$o$	$t$	$d$	$l$	$A_2^1$	$A_2^2$	$A_2^3$	$A_2^4$	$A_2^5$
$A_2$	*	*	*	3	*	$\frac{15}{n(S_{12})}$	$\frac{29}{n(S_{13})}$	*	*
					*	*	*	*	*
	$o$	$t$	$d$	$l$	$A_3^1$	$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$
$A_3$	$A_3^1$	*	*	0	$\frac{5}{n(S_3)}$	$\frac{14}{n(S_4)}$	$\frac{8}{n(S_6)}$	$\frac{17}{n(S_9)}$	$\frac{2}{n(S_{10})}$
					$A_3^2$	$A_3^3$	$A_3^4$	$A_3^5$	$A_2^3$

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)
  - ажурирање – перформансе
    - физичко брисање
      - обухвата динамичко управљање слободним локацијама
        - део заузетог простора бива ослобођен

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)

- одлике

- поређење са статичким расутим датотекама с отвореним адресирањем

- повећање ефикасности тражења

- посебно за случајеве фактора попуњености  $q$  изнад 0,7 и неуспешног тражења

- елиминисање узрока **У1** и **У2** који воде до погоршања перформанси тражења код статичких расутих датотека с отвореним адресирањем

- приликом тражења могу бити проверавани само бакети у којима су слогови из одговарајућег скупа синонима

- тражење се одвија најдаље до краја одговарајуће листе синонима

# Статичке расуте датотеке са спрезањем у једној зони

- Статичке расуте датотеке са спрезањем у примарној зони (Mogin, 2008)

- одлике

- поређење са статичким расутим датотекама с отвореним адресирањем

- смањење ефикасности искоришћења меморијског простора, за исти фактор попуњености
- задржавање проблема у вези с прекорачиоцима
  - узрок **УЗ** који води до погоршања перформанси тражења код статичких расутих датотека с отвореним адресирањем није елиминисан
    - постојање неких прекорачилаца може довести до појаве нових прекорачилаца
    - негативан утицај на перформансе тражења
    - потенцијална решења
      - реорганизација
      - формирање у два пролаза

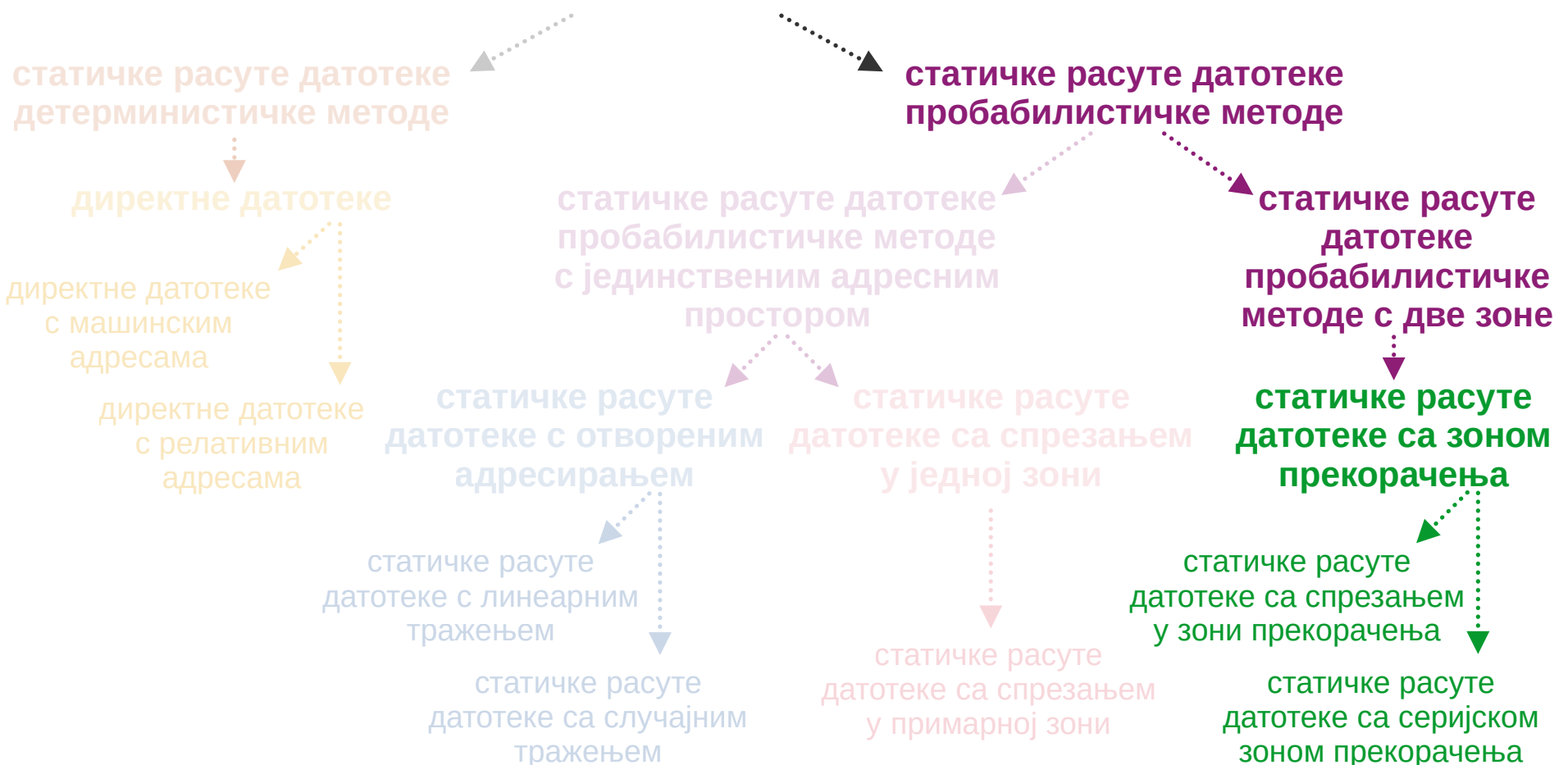
# Садржај

- Увод
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- **Статичке расуте датотеке са зоном прекорачења**
- Одлике и примена
- Ресурси

# Статичке расуте датотеке са зоном прекорачења

- Врсте статичких расутих датотека – примери (Mogin, 2008)

## статичке расуте датотеке



# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са зоном прекорачења (Mogin, 2008)
  - слогови датотеке су распоређени у две зоне
    - примарна зона садржи матичне бакете с примарним слоговима
    - зона прекорачења садржи прекорачиоце
  - смештање прекорачилаца
    - тражења места за смештај прекорачиоца у зони прекорачења

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са зоном прекорачења (Mogin, 2008)
  - врсте статичких расутих датотека са зоном прекорачења
    - статичке расуте датотеке са спрезањем у зони прекорачења
      - сви прекорачиоци су унутар зоне прекорачења и распоређени у спрегнуте листе прекорачилаца у зависности од припадности скупу синонима
    - статичке расуте датотеке са серијском зоном прекорачења
      - сви прекорачиоци су без спрезања унутар серијске зоне прекорачења

# Статичке расуте датотеке са зоном прекорачења

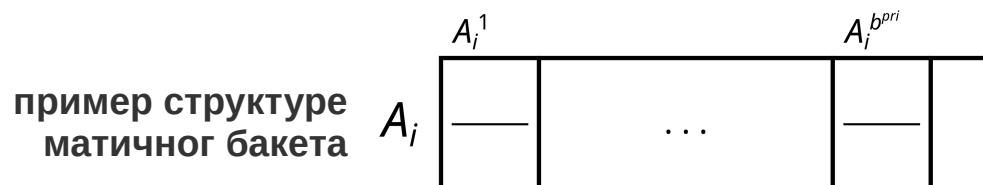
- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - слогови датотеке су распоређени у две зоне
    - примарна зона садржи матичне бакете с примарним слоговима
    - зона прекорачења садржи прекорачиоце
  - смештање прекорачилаца се заснива на праћењу показивача у зони прекорачења
    - тражења места за смештај прекорачиоца у листи сегмената зоне прекорачења са слободним простором
      - постојање индекса слободних бакета у зони прекорачења
    - прекорачилац бива увезан у одговарајућу листу прекорачилаца
      - листа прекорачилаца је везана за одговарајући матични бакет

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - структура
    - примарна зона
    - зона прекорачења

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - структура – примарна зона
    - $B^{pri}$  матичних бакета
      - структура матичног бакета
        - $b^{pri}$  локација за слоге
        - поље показивача на почетак листе прекорачилаца везаних за бакет

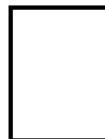


# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - структура – зона прекорачења
    - помоћна структура података
      - индекс слободних бакета  $L$ 
        - показивач на почетак листе сегмената зоне прекорачења са слободним простором
    - $B^{pre}$  бакета за прекорачиоце

пример структуре индекса  
слободних бакета

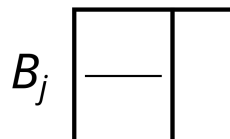
$L$



# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - структура – зона прекорачења – бакет
    - мала вероватноћа да у истом бакету буде више прекорачилаца из истог скупа синонима
    - случај  $b^{pre} = 1$ , што је често решење
      - структура бакета
        - једна локација за прекорачилац ( $b^{pre} = 1$ )
        - поље показивача
          - у случају заузете локације, показивач на наредни прекорачилац из истог скупа синонима
          - у случају слободне локације, показивач на наредну слободну локацију

пример структуре  
бакета за прекорачиоце



# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - формирање
    - 1) иницијализација садржаја зона
      - формирање примарне зоне
        - иницијализација садржаја бакетâ
          - свака локација је слободна
          - иницијализација празних листа прекорачилаца
      - формирање празне зоне прекорачења
        - иницијализација показивача у структури  $L$ 
          - упис адресе првог бакета
        - иницијализација садржаја бакетâ
          - свака локација је слободна
          - спрезање бакета

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - формирање
    - 2) смештање слогова
      - ако има слободне локације у матичном бакету, слог бива смештен у матични бакет у примарној зони
      - ако нема слободне локације у матичном бакету, слог бива смештен као прекорачилац у прву слободну локацију у зони прекорачења
        - нови слог бива увезан на почетак листе прекорачилаца одговарајућег матичног бакета
        - локација за смештај слога бива одвезана из листе сегмената зоне прекорачења са слободним простором
      - слог не може бити смештен ако нема одговарајућег слободног простора

# Статичке расуте датотеке са зоном прекорачења

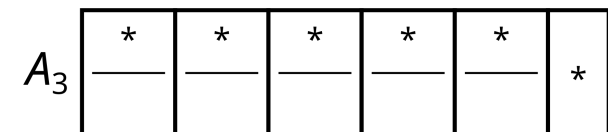
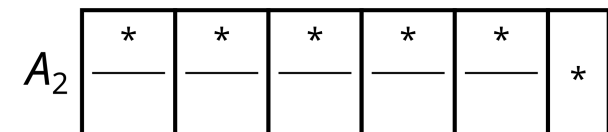
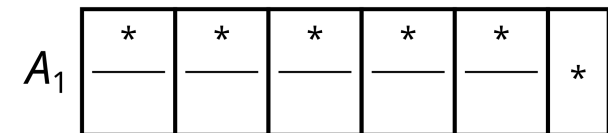
- Статичке расуте датотеке са спрезањем у зони прекорачења

## – пример 1 (без слогова)

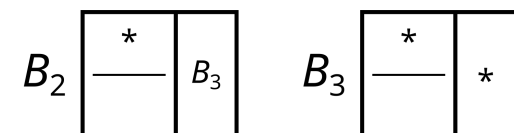
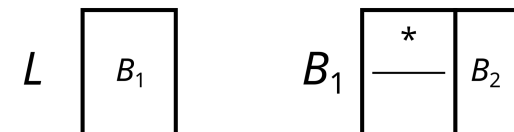
- примарна зона
  - фактор бакетирања  $b^{pri} = 5$
  - постоје укупно  $B^{pri} = 3$  бакета
    - бакети  $A_1, A_2, A_3$
    - показивачи на почетке листа прекорачилаца у посебним пољима показивача
    - постоји 15 слободних локација
      - специјална ознака у пољу кључа (\*)

$$h(k(S)) = 1 + k(S) \pmod{3}$$

ПРИМАРНА ЗОНА



ЗОНА ПРЕКОРАЧЕЊА



# Статичке расуте датотеке са зоном прекорачења

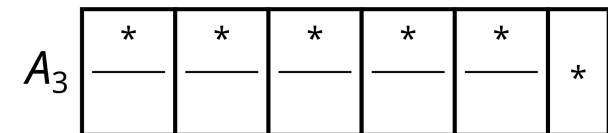
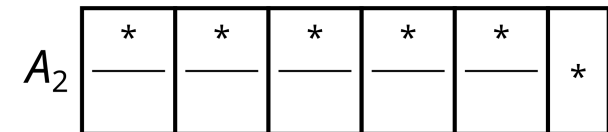
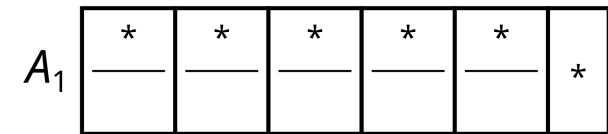
- Статичке расуте датотеке са спрезањем у зони прекорачења

## – пример 1 (без слогова)

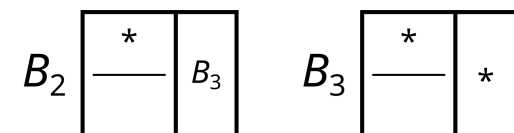
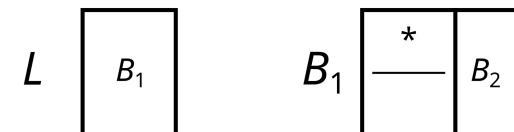
- зона прекорачења
  - фактор бакетирања  $b^{pre} = 1$
  - постоје укупно  $B^{pre} = 3$  бакета
    - бакети  $B_1, B_2, B_3$
    - постоје 3 слободне локације
      - специјална ознака у пољу кључа (\*)
  - помоћна структура података  $L$ 
    - показивач на почетак листе сегмената са слободним простором
- основна структура слога
  - вредност кључа (цео број)
  - вредности некључних обележја

$$h(k(S)) = 1 + k(S) \pmod{3}$$

ПРИМАРНА ЗОНА



ЗОНА ПРЕКОРАЧЕЊА



# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења

– пример 2

- постоји  $N = 15$  слогова
  - 15 обичних слогова
    - слогови  $S_1, \dots, S_{15}$
- постоји један бакет са слободним простором
  - бакет  $A_2$ 
    - 3 слободне локације

$$b^{pri} = 5 \quad B^{pri} = 3 \quad b^{pre} = 1 \quad B^{pre} = 3$$

$$h(k(S)) = 1 + k(S) \pmod{3}$$

ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{6}{n(S_9)}$	$\frac{27}{n(S_{11})}$	$B_2$
-------	---------------------	--------------------	---------------------	--------------------	------------------------	-------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	*	*	*	*
-------	--------------------	------------------------	---	---	---	---

$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$	$B_3$
-------	--------------------	---------------------	--------------------	---------------------	------------------------	-------

ЗОНА ПРЕКОРАЧЕЊА

$L$	*	$B_1$	$\frac{36}{n(S_{12})}$	*
-----	---	-------	------------------------	---

$B_2$	$\frac{42}{n(S_{14})}$	$B_1$	$B_3$	$\frac{14}{n(S_{15})}$	*
-------	------------------------	-------	-------	------------------------	---

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења

– пример 2

- постоје 3 скупа синонима
  - $\{S_2, S_3, S_8, S_9, S_{11}, S_{12}, S_{14}\}$
  - $\{S_5, S_{13}\}$
  - $\{S_1, S_4, S_6, S_7, S_{10}, S_{15}\}$
- постоји 12 примарних слогова
  - $S_2, S_3, S_8, S_9, S_{11}, S_5, S_{13}, S_1, S_4, S_6, S_7, S_{10}$
- постоје 3 листе прекорачилаца
  - $[S_{14}, S_{12}]$
  - $[\ ]$
  - $[S_{15}]$

$$b^{pri}=5 \quad B^{pri}=3 \quad b^{pre}=1 \quad B^{pre}=3$$

$$h(k(S))=1+k(S) \pmod{3}$$

ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{6}{n(S_9)}$	$\frac{27}{n(S_{11})}$	$B_2$
-------	---------------------	--------------------	---------------------	--------------------	------------------------	-------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	*	*	*	*
-------	--------------------	------------------------	---	---	---	---

$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$	$B_3$
-------	--------------------	---------------------	--------------------	---------------------	------------------------	-------

ЗОНА ПРЕКОРАЧЕЊА

$L$	*	$B_1$	$\frac{36}{n(S_{12})}$	*
-----	---	-------	------------------------	---

$B_2$	$\frac{42}{n(S_{14})}$	$B_1$	$B_3$	$\frac{14}{n(S_{15})}$	*
-------	------------------------	-------	-------	------------------------	---

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - тражење
    - прво долази до приступа матичном бакету, чија адреса је добијена применом трансформације аргумента тражења
    - у случају да тражени слог јесте пронађен у матичном бакету
      - тражење је успешно
    - у случају да тражени слог није пронађен у матичном бакету
      - ако за матични бакет нема везаних прекорачилаца, тражење је неуспешно
      - ако има прекорачилаца везаних за матични бакет, тражење тече праћењем показивача кроз одговарајућу листу прекорачилаца
        - у случају наиласка на тражени слог, тражење је успешно
        - у случају проласка кроз целокупну листу прекорачилаца без наиласка на тражени слог, тражење је неуспешно

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења

## – тражење – пример

- тражење слога с вредношћу кључа  $k = 11$ 
  - исход: ?
  - број приступа: ?
- тражење слога с вредношћу кључа  $k = 17$ 
  - исход: ?
  - број приступа: ?
- тражење слога с вредношћу кључа  $k = 36$ 
  - исход: ?
  - број приступа: ?

$$b^{pri} = 5 \quad B^{pri} = 3 \quad b^{pre} = 1 \quad B^{pre} = 3$$

$$h(k(S)) = 1 + k(S) \pmod{3}$$

ПРИМАРНА ЗОНА

$$A_1 \begin{array}{|c|c|c|c|c|} \hline \frac{12}{n(S_2)} & \frac{9}{n(S_3)} & \frac{18}{n(S_8)} & \frac{6}{n(S_9)} & \frac{27}{n(S_{11})} \\ \hline \end{array} B_2$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline \frac{7}{n(S_5)} & \frac{19}{n(S_{13})} & * & * & * \\ \hline \end{array} *$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline \frac{5}{n(S_1)} & \frac{20}{n(S_4)} & \frac{2}{n(S_6)} & \frac{11}{n(S_7)} & \frac{23}{n(S_{10})} \\ \hline \end{array} B_3$$

ЗОНА ПРЕКОРАЧЕЊА

$$L \begin{array}{|c|} \hline * \\ \hline \end{array} \quad B_1 \begin{array}{|c|c|} \hline \frac{36}{n(S_{12})} & * \\ \hline \end{array}$$

$$B_2 \begin{array}{|c|c|} \hline \frac{42}{n(S_{14})} & B_1 \\ \hline \end{array} \quad B_3 \begin{array}{|c|c|} \hline \frac{14}{n(S_{15})} & * \\ \hline \end{array}$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења

## – тражење – пример

- тражење слога с вредношћу кључа  $k = 11$ 
  - исход: *успешно*
  - број приступа: 1
- тражење слога с вредношћу кључа  $k = 17$ 
  - исход: *неуспешно*
  - број приступа: 2
- тражење слога с вредношћу кључа  $k = 36$ 
  - исход: *успешно*
  - број приступа: 3

$$b^{pri} = 5 \quad B^{pri} = 3 \quad b^{pre} = 1 \quad B^{pre} = 3$$

$$h(k(S)) = 1 + k(S) \pmod{3}$$

ПРИМАРНА ЗОНА

$$A_1 \left[ \begin{array}{c|c|c|c|c|} \frac{12}{n(S_2)} & \frac{9}{n(S_3)} & \frac{18}{n(S_8)} & \frac{6}{n(S_9)} & \frac{27}{n(S_{11})} & B_2 \end{array} \right]$$

$$A_2 \left[ \begin{array}{c|c|c|c|c|} \frac{7}{n(S_5)} & \frac{19}{n(S_{13})} & * & * & * & * \end{array} \right]$$

$$A_3 \left[ \begin{array}{c|c|c|c|c|} \frac{5}{n(S_1)} & \frac{20}{n(S_4)} & \frac{2}{n(S_6)} & \frac{11}{n(S_7)} & \frac{23}{n(S_{10})} & B_3 \end{array} \right]$$

ЗОНА ПРЕКОРАЧЕЊА

$$L \left[ \begin{array}{c|} * \end{array} \right] \quad B_1 \left[ \begin{array}{c|c} \frac{36}{n(S_{12})} & * \end{array} \right]$$

$$B_2 \left[ \begin{array}{c|c} \frac{42}{n(S_{14})} & B_1 \end{array} \right] \quad B_3 \left[ \begin{array}{c|c} \frac{14}{n(S_{15})} & * \end{array} \right]$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>

– тражење – перформансе

- број приступа за успешно тражење  $R_u$

$$1 \leq R_u \leq 1+r$$

$$r = \begin{cases} 0, & S \leq b^{prim} \\ S - b^{prim}, & S > b^{prim} \end{cases}$$

- $S$  – број слогова у скупу синонима

- број приступа за неуспешно тражење  $R_n$

$$R_n = \begin{cases} 1, & 0 \leq S \leq b^{prim} \\ 1 + S - b^{prim}, & b^{prim} < S \leq N \end{cases}$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - ажурирање
    - упис
      - нови слог бива уписан након неуспешног тражења, под условом да има одговарајућег слободног места
        - локација за смештај новог слога је прва слободна локација у матичном бакету, ако тамо има слободног места, а иначе је прва слободна локација у зони прекорачења, под условом да у зони прекорачења има слободног места
        - случај смештања слога у зону прекорачења
          - локација за смештај новог слога бива одвезана из листе сегмената зоне прекорачења са слободним простором
          - нови слог бива увезан на почетак листе прекорачилаца везане за матични бакет

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења
  - ажурирање
    - упис – пример

## СТАРО СТАЊЕ

$$b^{pri}=5 \quad B^{pri}=3 \quad b^{pre}=1 \quad B^{pre}=3$$

$$h(k(S))=1+k(S) \pmod{3}$$

### ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{6}{n(S_9)}$	$\frac{27}{n(S_{11})}$	$B_2$
-------	---------------------	--------------------	---------------------	--------------------	------------------------	-------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	*	*	*	*
-------	--------------------	------------------------	---	---	---	---

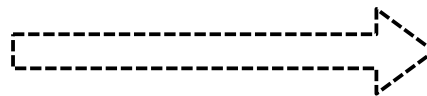
$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$	$B_3$
-------	--------------------	---------------------	--------------------	---------------------	------------------------	-------

### ЗОНА ПРЕКОРАЧЕЊА

$L$	*	$B_1$	$\frac{36}{n(S_{12})}$	*
-----	---	-------	------------------------	---

$B_2$	$\frac{42}{n(S_{14})}$	$B_1$	$B_3$	$\frac{14}{n(S_{15})}$	*
-------	------------------------	-------	-------	------------------------	---

упис слога с вредношћу  
кључа  $k = 3$



исход: неуспешно

## НОВО СТАЊЕ

$$b^{pri}=5 \quad B^{pri}=3 \quad b^{pre}=1 \quad B^{pre}=3$$

$$h(k(S))=1+k(S) \pmod{3}$$

### ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{6}{n(S_9)}$	$\frac{27}{n(S_{11})}$	$B_2$
-------	---------------------	--------------------	---------------------	--------------------	------------------------	-------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	*	*	*	*
-------	--------------------	------------------------	---	---	---	---

$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$	$B_3$
-------	--------------------	---------------------	--------------------	---------------------	------------------------	-------

### ЗОНА ПРЕКОРАЧЕЊА

$L$	*	$B_1$	$\frac{36}{n(S_{12})}$	*
-----	---	-------	------------------------	---

$B_2$	$\frac{42}{n(S_{14})}$	$B_1$	$B_3$	$\frac{14}{n(S_{15})}$	*
-------	------------------------	-------	-------	------------------------	---

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења (Mogin, 2008)
  - ажурирање
    - модификација
      - као у случају модификације за серијску или секвенцијалну организацију, али уз ослањање на тражење које се користи у статичким расутим датотекама са спрезањем у зони прекорачења

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења (Mogin, 2008)
  - ажурирање
    - логичко брисање
      - по узору на логичко брисање код статичких расутих датотека са спрезањем у примарној зони, али уз ослањање на тражење које се користи код статичких расутих датотека са спрезањем у зони прекорачења
      - додатна могућност је да локације логички обрисаних слогова могу бити поново искоришћене
        - локација логички обрисаног слога може бити употребљена за смештај новог слога приликом уписа, али под условом да су оба слога из истог скупа синонима

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења (Mogin, 2008)
  - ажурирање
    - физичко брисање
      - након успешног тражења слога за брисање, тај слог бива уклоњен и бива ослобођена једна локација
        - случај када је слог за брисање у матичном бакету, део 1
          - долази до померања садржаја унутар матичног бакета, под условом да слог за брисање није у последњој локацији
            - садржаји који су у опсегу од прве локације иза локације слога за брисање до последње локације у бакету бивају померени за једну локацију налево
            - локација која је пре почетка померања била слободна остаје слободна

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - ажурирање
    - физичко брисање
      - након успешног тражења слога за брисање, тај слог бива уклоњен и бива ослобођена једна локација
        - случај када је слог за брисање у матичном бакету, део 2
          - ако нема прекорачилаца везаних за матични бакет
            - последња локација у матичном бакету је слободна
          - ако има прекорачилаца везаних за матични бакет
            - први прекорачилац бива премештен у последњу локацију матичног бакета и одвезан из листе прекорачилаца
            - првобитна локација премештеног прекорачиоца постаје слободна и бива увезана на почетак листе сегмената зоне прекорачења са слободним простором

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - ажурирање
    - физичко брисање
      - након успешног тражења слога за брисање, тај слог бива уклоњен и бива ослобођена једна локација
        - случај када је слог за брисање у листи прекорачилаца
          - слог за брисање бива одвезан из листе прекорачилаца
          - локација слога за брисање постаје слободна и бива увезана на почетак листе сегмената зоне прекорачења са слободним простором

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења
  - ажурирање
    - физичко брисање – пример

## СТАРО СТАЊЕ

$$b^{pri}=5 \quad B^{pri}=3 \quad b^{pre}=1 \quad B^{pre}=3$$

$$h(k(S))=1+k(S) \pmod{3}$$

### ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{6}{n(S_9)}$	$\frac{27}{n(S_{11})}$	$B_2$
-------	---------------------	--------------------	---------------------	--------------------	------------------------	-------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	*	*	*	*
-------	--------------------	------------------------	---	---	---	---

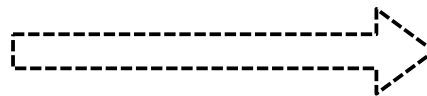
$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$	$B_3$
-------	--------------------	---------------------	--------------------	---------------------	------------------------	-------

### ЗОНА ПРЕКОРАЧЕЊА

$L$	*	$B_1$	$\frac{36}{n(S_{12})}$	*
-----	---	-------	------------------------	---

$B_2$	$\frac{42}{n(S_{14})}$	$B_1$	$B_3$	$\frac{14}{n(S_{15})}$	*
-------	------------------------	-------	-------	------------------------	---

физичко брисање слога с вредношћу кључа  $k = 6$



исход: успешно

## НОВО СТАЊЕ

$$b^{pri}=5 \quad B^{pri}=3 \quad b^{pre}=1 \quad B^{pre}=3$$

$$h(k(S))=1+k(S) \pmod{3}$$

### ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{27}{n(S_{11})}$	$\frac{42}{n(S_{14})}$	$B_1$
-------	---------------------	--------------------	---------------------	------------------------	------------------------	-------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	*	*	*	*
-------	--------------------	------------------------	---	---	---	---

$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$	$B_3$
-------	--------------------	---------------------	--------------------	---------------------	------------------------	-------

### ЗОНА ПРЕКОРАЧЕЊА

$L$	$B_2$	$B_1$	$\frac{36}{n(S_{12})}$	*
-----	-------	-------	------------------------	---

$B_2$	*	*	$B_3$	$\frac{14}{n(S_{15})}$	*
-------	---	---	-------	------------------------	---

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>

- ажурирање – перформансе

- упис

- број приступа за успешан упис  $R_i$

$$R_i = \begin{cases} 2, & 0 \leq S < b^{prim} \\ S - b^{prim} + 4, & b^{prim} \leq S \leq N \end{cases}$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>

- ажурирање – перформансе

- логичко брисање

- број приступа за успешно логичко брисање  $R_d'$

$$R_d' = R_u + 1$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - ажурирање – перформансе
    - физичко брисање
      - број приступа за успешно физичко брисање  $R_d^f$ 
        - ако нема прекорачилаца везаних за матични бакет
$$R_d^f = 2$$
        - ако има прекорачилаца везаних за матични бакет
          - случај брисања примарног слога или првог прекорачиоца
$$R_d^f = 4$$
          - случај брисања прекорачиоца који није први
$$R_d^f = R_u + 2$$
$$1 < R_u \leq 1 + S - b^{prim}$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - одлике
    - поређење са статичким расутим датотекама с отвореним адресирањем
      - елиминисање узрока **У1**, **У2** и **У3** који воде до погоршања перформанси тражења код статичких расутих датотека с отвореним адресирањем
        - приликом тражења могу бити проверавани само матични бакет и одговарајући ланац прекорачилаца
        - тражење се одвија најдаље до краја листе прекорачилаца из истог скупа синонима
        - сви матични бакети су у примарној зони а сви прекорачиоци су у зони прекорачења, те није могуће да прекорачилац буде у нематичном бакету примарне зоне

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са спрезањем у зони прекорачења <sup>(Mogin, 2008)</sup>
  - одлике
    - погодности у односу на друге врсте статичких расутих датотека
      - очекиване боље перформансе тражења
        - мањи очекивани број приступа за неуспешно тражење, нарочито када фактор попуњености близу 1
        - релативно повољан очекивани број приступа за успешно тражење
      - узроци побољшања перформанси тражења
        - увођење зоне прекорачења за смештај прекорачилаца и повезивање прекорачилаца с матичним бакетом
          - додатни меморијски простор потребан за зону прекорачења и показиваче

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са серијском зоном прекорачења (Mogin, 2008)
  - слогови датотеке су распоређени у две зоне
    - примарна зона садржи матичне бакете с примарним слоговима
    - зона прекорачења садржи прекорачиоце
  - смештање прекорачилаца се заснива на линеарном тражењу локације у зони прекорачења
    - прекорачиоци бивају смештани у узастопне локације зоне прекорачења
    - прекорачиоци нису међусобно спрегнути
    - прекорачиоци нису спрегнути с матичним бакетом

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са серијском зоном прекорачења

- пример

- примарна зона
  - фактор бакетирања  $b^{pri} = 5$
  - постоје укупно  $B^{pri} = 3$  бакета
    - бакети  $A_1, A_2, A_3$
- зона прекорачења
  - фактор бакетирања  $b^{pre} = 3$
  - постоје укупно  $B^{pre} = 2$  бакета
    - бакети  $B_1, B_2$
- основна структура слога
  - вредност кључа (цео број)
  - вредности некључних обележја
- постоји  $N = 15$  слогова
  - 15 обичних слогова
    - слогови  $S_1, \dots, S_{15}$

$$h(k(S)) = 1 + k(S) \pmod{3}$$

## ПРИМАРНА ЗОНА

$A_1$	$\frac{12}{n(S_2)}$	$\frac{9}{n(S_3)}$	$\frac{18}{n(S_8)}$	$\frac{6}{n(S_9)}$	$\frac{27}{n(S_{11})}$
-------	---------------------	--------------------	---------------------	--------------------	------------------------

$A_2$	$\frac{7}{n(S_5)}$	$\frac{19}{n(S_{13})}$	$\frac{*}{n(S_?)}$	$\frac{*}{n(S_?)}$	$\frac{*}{n(S_?)}$
-------	--------------------	------------------------	--------------------	--------------------	--------------------

$A_3$	$\frac{5}{n(S_1)}$	$\frac{20}{n(S_4)}$	$\frac{2}{n(S_6)}$	$\frac{11}{n(S_7)}$	$\frac{23}{n(S_{10})}$
-------	--------------------	---------------------	--------------------	---------------------	------------------------

## ЗОНА ПРЕКОРАЧЕЊА

$B_1$	$\frac{36}{n(S_{12})}$	$\frac{42}{n(S_{14})}$	$\frac{14}{n(S_{15})}$
-------	------------------------	------------------------	------------------------

$B_2$	$\frac{*}{n(S_?)}$	$\frac{*}{n(S_?)}$	$\frac{*}{n(S_?)}$
-------	--------------------	--------------------	--------------------

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са серијском зоном прекорачења

- пример

- постоје 3 скупа синонима
  - $\{S_2, S_3, S_8, S_9, S_{11}, S_{12}, S_{14}\}$
  - $\{S_5, S_{13}\}$
  - $\{S_1, S_4, S_6, S_7, S_{10}, S_{15}\}$
- постоји 12 примарних слогова
  - $S_2, S_3, S_8, S_9, S_{11}, S_5, S_{13}, S_1, S_4, S_6, S_7, S_{10}$
- постоје 3 слога прекорачиоца
  - $S_{12}, S_{14}, S_{15}$

$$h(k(S)) = 1 + k(S) \pmod{3}$$

ПРИМАРНА ЗОНА

$$A_1 \begin{array}{|c|c|c|c|c|} \hline 12 & 9 & 18 & 6 & 27 \\ \hline n(S_2) & n(S_3) & n(S_8) & n(S_9) & n(S_{11}) \\ \hline \end{array}$$

$$A_2 \begin{array}{|c|c|c|c|c|} \hline 7 & 19 & * & * & * \\ \hline n(S_5) & n(S_{13}) & & & \\ \hline \end{array}$$

$$A_3 \begin{array}{|c|c|c|c|c|} \hline 5 & 20 & 2 & 11 & 23 \\ \hline n(S_1) & n(S_4) & n(S_6) & n(S_7) & n(S_{10}) \\ \hline \end{array}$$

ЗОНА ПРЕКОРАЧЕЊА

$$B_1 \begin{array}{|c|c|c|} \hline 36 & 42 & 14 \\ \hline n(S_{12}) & n(S_{14}) & n(S_{15}) \\ \hline \end{array}$$

$$B_2 \begin{array}{|c|c|c|} \hline * & * & * \\ \hline & & \\ \hline \end{array}$$

# Статичке расуте датотеке са зоном прекорачења

- Статичке расуте датотеке са серијском зоном прекорачења (Mogin, 2008)
  - одлике и примена
    - структурна једноставност
    - погодност за случајеве малог броја прекорачилаца а ретког ажурирања

# Садржај

- Увод
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- **Одлике и примена**
- Ресурси

# Одлике и примена

- Одлике и примена (Mogin, 2008)
  - предности
    - мали очекивани број приступа за тражење случајно одабраног слога
    - очекивана погодност за улогу обрађиване датотеке у режиму директне обраде

# Одлике и примена

- Одлике и примена (Mogin, 2008)
  - мане
    - величина датотеке треба да буде унапред одређена
      - погодност само за датотеке које ретко бивају ажуриране
    - случај примене пробабилистичке трансформације вредности кључа у адреса
      - треба одабрати трансформацију која постиже равномерну расподелу синонима по бакетима при формирању и ажурирању
        - лош одабир може довести до повећаног броја прекорачилаца
    - број приступа при тражењу може понекад бити веома велик

# Одлике и примена

- Одлике и примена (Mogin, 2008)
  - примена
    - примена физичких структура типа статичких расутих датотека
      - интерактивна даљинска обрада
        - када корисници траже податке произвољним редоследом
      - режим пакетне обраде
        - када подаци тражени случајним редоследом
    - примена физичких структура података заснованих на принципима изградње статичких расутих датотека
      - системи за управљање мрежним базама података
      - поједини системи за управљање релационим базама података

# Садржај

- Увод
- Статичке расуте датотеке детерминистичке методе
- Методе пробабилистичке трансформације
- Статичке расуте датотеке пробабилистичке методе
- Статичке расуте датотеке с отвореним адресирањем
- Статичке расуте датотеке са спрезањем у једној зони
- Статичке расуте датотеке са зоном прекорачења
- Одлике и примена
- **Ресурси**

# Ресурси

- Извори и литература
  - Pavle Mogin. Strukture podataka i organizacija datoteka. 3. izdanje. Računarski fakultet (Beograd, Srbija), CET (Beograd, Srbija). 2008.
    - Glava 11. Statička rasuta organizacija datoteke