SZÓTÁR (üresre) init(), insert(k), search(k), remove(k)

$U :=$ kulcsok univerzuma — EGYEDI KULCSOK!

$U = [0..m) \land m \geq n$ $\quad T(m >> n)$ $\quad$ Pl $m \leq 1.2n$

Direkt címzés $T : D^*[m]$

D
$+\ k : U$ // k is the key
$+\ \ldots$ // satellite data

Direct-address tables
Direct addressing

$T \in \Theta(1)$

$\Theta(m)$ idő

init(T:D*[m])

$i := 0$ to $m-1$	
	$T[i] := \oslash$

search(T:D*[] ; k:U):D*

return $T[k]$

insert(T:D*[] ; p:D*):$\mathbb{B}$

$T[p \to k] = \oslash$	
$T[p \to k] := p$	return *false*
return *true*	

remove(T:D*[] ; k:U):D*

$p := T[k]$
$T[k] := \oslash$
return p

Probléma:

$|U| >> n$ sokkal nagyobb: $>>$

hash fv.

$h : U \to [0..m)$

nem injektív általában

$k_1 \neq k_2$ $(k_1, k_2 \in U)$

$h(k_1) = h(k_2)$

key collision!

HASH TABLES (HASÍTÓ TÁBLÁK)

Notations:

m: the size of the hash table

$T[0..(m-1)]$: the hash table

rések

$T[0], T[1], \ldots, T[m-1]$: the slots of the hash table

$\oslash$: empty slot in the hash table (when we use direct-address tables or key collisions are resolved by chaining) LÁNCOLÁSSAL OLDJUK FEL A KULCSÜTKÖZÉST. $\Big\}$ $\alpha > 1$ lehetséges

E: the key of empty slots in case of open addressing

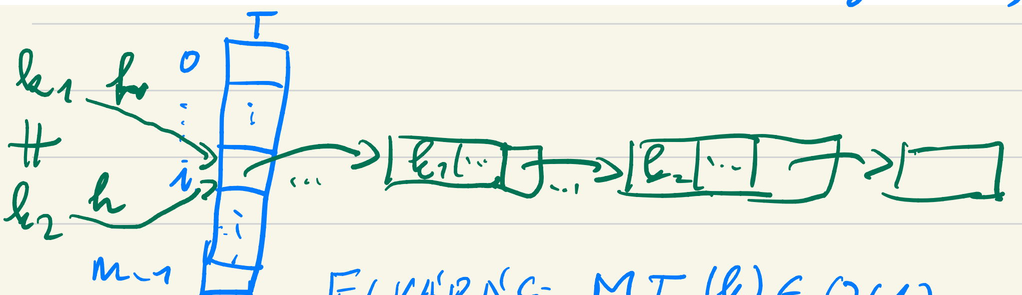
D: the key of deleted slots in case of open addressing $\Big\}$ NYÍLT CÍMZÉSSEL $\Big\}$ $\alpha \leq 1$

n: the number of records stored in the hash table

$\alpha = n/m$: *load factor* (KITÖLTÖTTSÉGI HÁNYADOS)

U : the universe of keys; $k, k', k_i \in U$

$h : U \to 0..(m-1)$: hash function (ált. $\neg$ injektív) ELVÁRÁS: $\Big\{$ EGYSZERŰ EGYENLETES HASÍTÁS (simple uniform hashing)



$\Big\}$ KULCSÜTKÖZÉS FELOLDÁSA LÁNCOLÁSSAL

ELVÁRÁS: $MT_h(k) \in \Theta(1)$

k kulcs $\approx \frac{1}{m}$ valószínűséggel: $h(k) = i$

Collision resolution by chaining

Biz: CLRS könyv

E1
$+key : U$ ... // satellite data may come here $+next$: **E1***
+**E1**() { $next := \emptyset$ }

$T_{init}(m) \in \Theta(m)$

$mT_{\substack{insert\\search\\remove}} \in \Theta(1)$

init(T:E1*[m])

$i := 0$ to $m-1$	
	$T[i] := \emptyset$

search(T:E1*[] ; k:U):E1*

return searchS1L($T[h(k)]$,k)

insert(T:E1*[] ; p:E1*):$\mathbb{B}$

$k := p \to key$; $s := h(k)$	
searchS1L($T[s], k$) $= \emptyset$	
$p \to next := T[s]$	return $false$
$T[s] := p$	
return $true$	

searchS1L(q:E1* ; k:U):E1*

$q \neq \emptyset \wedge q \to key \neq k$	
	$q := q \to next$
return q	

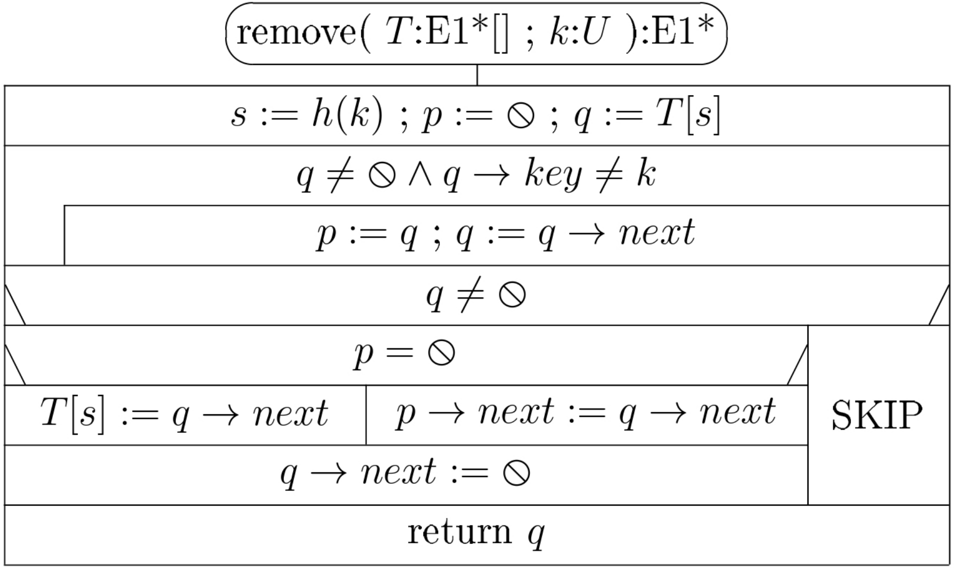
$AT_{\substack{i\\s\\r}}(n,m) \in \Theta(1+\frac{n}{m})$

$MT_{\substack{i\\s\\r}}(n) \in \Theta(n)$

$\exists k \in \mathbb{P}$ konstans:

$\alpha = \frac{n}{m} < k$

$\Rightarrow AT_{\substack{i\\s\\r}} \in \Theta(1)$



JÓ HASÍTÓ FV-EK

① OSZTÓ MÓDSZER (division method)

$U \subseteq \mathbb{Z}$

$$h(k) = k \bmod m$$

m prímszám, ami nem esik közel 2-hatványhoz

h: valószínűleg egysz. egy. hasítás.

Pl. $\left.\begin{array}{l} n \approx 2000 \\ \alpha \approx 3 \end{array}\right\}$ adott:

$m = 701$ jó választás várhatólag

↑ prím ✓

$512 < 701 < 1024$

$\alpha = \frac{n}{m} \approx 3$

HA A KULCS VALÓS SZÁM:

(2) EGYSZERŰ SZORZÓ MÓDSZER: $k \in [0;1)$

A KULCSOK EGYENLETESEN OSZOLJANAK EL A $[0;1)$-en.

$$h(k) = \lfloor k * m \rfloor \in [0..m)$$

(simple multiplication method)

(2b) $k \in [a;b)$

$$h(k) = \left\lfloor \frac{k-a}{b-a} * m \right\rfloor \in [0..m)$$

③ SZORZÓ MÓDSZER ($U \subseteq \mathbb{R}$)

(MULTIPLICATION METHOD)

$A \in (0;1)$ konstans

$$h(k) = \lfloor \{k * A\} * m \rfloor$$

A legtöbb kulcshalmazra egyszerű, egyenletes hasítást eredményez

$$A = \frac{\sqrt{5}-1}{2} \approx 0.618$$

(KNUTH)

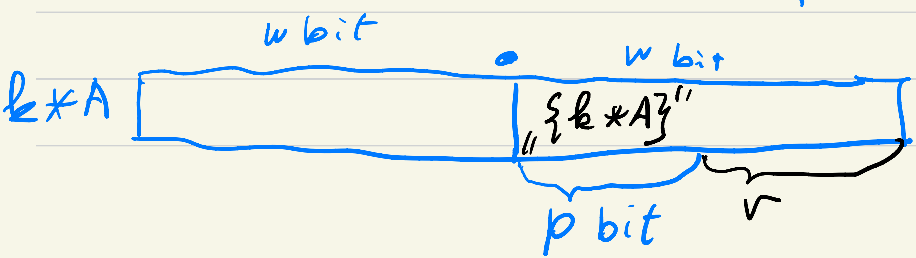
Egész számokra:

$$k \in 0..2^w-1 \Rightarrow m := 2^p \quad (p \in (1..w))$$

$$q := 2^w - 1 \qquad r := w - p$$

$$A := \left\lfloor \frac{\sqrt{5}-1}{2} \cdot 2^w \right\rfloor$$

$$h(k) = \underline{(k * A)\,\&\,q} >> r$$



NYÍLT CÍMZÉS (OPEN ADDRESSING)

R
$+\ k : U \cup \{E, D\}$ // k is a key or it is Empty or Deleted
$+ \ldots$ // satellite data // $E \neq D \wedge E, D \notin U$

Pl. $U \subseteq \mathbb{N} \Rightarrow$ lehet pl. $E = -2;\ D = -1$

init(T:R[m])

- $i := 0$ to $m-1$
  - $T[i].k := E$

Notations for open addressing:

$h : U \times 0..(m-1) \to 0..(m-1)$: probing function — próba fv.

$\langle h(k,0), h(k,1), \ldots, h(k, m-1) \rangle$: potential probing sequence — potenciális próbasorozat

$\underbrace{\langle h(k,0), \ldots, h(k, i-1) \rangle}$

actual pr.seq. (aktuális pr.sor.) i a hossza. ($i \in 1..m$)

Elvárás: perm.-ja legyen $\langle 0, 1, \ldots, m-1 \rangle$ sorozatnak.

1.) Feltesszük h. nincs törlés

Beszúrás(x) ($i \geq 1 \wedge i \leq m$)

L: $h(k, i-1)$: üres $\Rightarrow$ $T[h(k, i-1)] := x$; ✓

$h(k, i-1)$ foglalt $\Rightarrow$

a.) $T(h(k, i-1)).k = x.k$ $\Rightarrow$ fail

b.) kül. $i{+}{+}$; folyt. $i = m+1 \Rightarrow$ fail

kül goto L

keres(k): a $h(k,0)$, $h(k,1)$, stb. réseket próbálja.

C: Ha $h(k,i-1)$ üres => fail

kül ha $T[h(k,i-1)].k = k$ => return $h(k,i-1)$

kül ha $i = m$ => fail

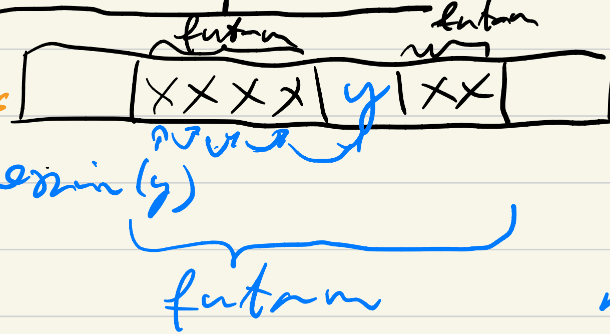
kül $i{+}{+}$; goto C

Próbasor generálása: $h_1 : U \to [0..m)$ egyenl. egy. has.

I. lineáris próba: (LINEAR PROBING) $h(k,i) = (h_1(k)+i) \bmod m$

$h(k,i) = (h(k,i-1)+1) \bmod m$

csak, ha a kulcsütközés val.-e minimális.



Elsődleges (primary csomósodás clustering)

m db próbasor van.

Másodlagos csomósodás: (secondary clustering)

$k_1 \neq k_2$, $h_1(k_1) = h_1(k_2)$ => az egész pr. sor. =

II. $h(k,i) = (h_1(k) + g(i)) \underline{\bmod} \ m$ g egyszerű másodfokú (négyzetes próba)

<u>problémák</u>: a) Másodlagos csomósodás

b) Nehéz biztosítani, h. $\langle h(k,0), \dots, h(k, m-1) \rangle$ lefedje $T[0..m)$-et.

Jó, ha kicsi a kulcsütközés vsz.-e.

(<u>DE</u> <u>Pl.</u> ha $g(i) = \frac{i+i^2}{2} \wedge m = 2^p$

$\Rightarrow$ lefedi. $\left[h(k,i) = (h(k, i-1) + i) \ \underline{\bmod} \ m \right]$)

III. <u>kettős hasítás</u> $h_1 : U \to [0..m)$ egyenl. egy. has.

$h_2 : U \to [1..m)$ — // —

$h(k,i) = (h_1(k) + i \, h_2(k)) \ \underline{\bmod} \ m$

$d := h_2(k)$; $h(k,0) = h_1(k)$

$m > i > 0 \Rightarrow h(k,i) = (h(k,i-1) + d) \underline{\bmod} \; m$

Pot. pr. sor lefedi a táblát $\Leftrightarrow$ $\mathrm{lnko}(m, h_2(k)) = 1$

<u>Elvárás:</u> $\forall k \in U$: $\mathrm{lnko}(m, h_2(k)) = 1$

<u>Pl.</u> A) m prím (távol a kettő-hatványoktól)

$h_1(k) = k \; \underline{\bmod} \; m$

$h_2(k) = 1 + (k \; \underline{\bmod} \; m')$

v. $\begin{cases} m' = m-1 : & m(m-1) \text{ pr. sor. van.} \\ m' = m-2 : & m(m-2) \text{ pr. sor. van.} \end{cases}$

(m', m) ikerprímek

<u>pl.</u> $(1607, 1609)$

$1607 * 1609 = 2.585.663$

Mindkét csomósodástól mentes a tapasztalatok szerint.

<u>B)</u> $m = 2^p$

$h_1(k) = \lfloor \{k * A\} * m \rfloor$

$h_2(k)$ ptlan.

$\Big\}$ $\dfrac{m^2}{2}$ pr. sor. van. max.

Bsp $m = 11$; $h_1(k) = k \underline{\bmod} 11$; $h_2(k) = 1 + (k \underline{\bmod} 10)$

op	key	h_2	probes	s	0	1	2	3	4	5	6	7	8	9	10
init				+											
ins	32		10	+											32
ins	40		7	+								40			32
ins	37		4	+					37			40			32
ins	15	6	4; 10; 5	+					37	15		40			32
ins	70	1	4; 5; 6	+					37	15	70	40			32
src	15	6	4; 10; 5	+					37	15	70	40			32
src	104	5	5; 10; 4; 9	–					37	15	70	40			32
del	15	6	4; 10; 5	+					37	D	70	40			32
src	70	1	4; 5; 6	+					37	D	70	40			32
ins	70	1	4; **<u>5</u>**; 6	–					37	D	70	40			32
del	37		4	+					D	D	70	40			32
ins	104	5	**<u>5</u>**; 10; 4; 9	+					D	104	70	40			32
src	15	6	4; 10; 5; 0	–					D	104	70	40			32

2.) Ha van törlés: $j :=$ sikeres keresés + $T[j].k := D$

Sikertelen keresés $\Rightarrow$ a törlés is sikertelen

keresés: ugyanúgy, mint eddig; a törölt réseken átlép

beszúrás(x) teljes keresés, de az elsőnek talált törölt rést megjegyzi, ha talál.

– sikeres keresés $\Rightarrow$ sikertelen beszúrás

– sikertelen keresés, ami talált törölt rést $\Rightarrow$ a x-et a megjegyzett törölt résbe teszi

– sikertelen keresés, ami nem talált törölt rést $\Rightarrow$ ha a keresés üres résen állt meg, $\Rightarrow$ oda teszi x-et.

különben kimenítette a pot.pr. sort: a beszúrás is sikertelen, mert a keresés nem talált sem üres, sem törölt rést.

insert(T:R[m] ; x:R):$\mathbb{Z}$

- $k := x.k$; $j := h_1(k)$
- $i := 0$; $d := h_2(k)$
- while $i < m \wedge T[j].k \notin \{E, D\}$
  - if $T[j].k = k$
    - then: **return** $-(j+1)$
    - else: $i{+}{+}$; $j := (j + d) \bmod m$
- if $i < m$
  - then: $ii := j$
  - else: **return** $-(m+1)$
- while $i < m \wedge T[j].k \neq \mathbf{E}$
  - if $T[j].k = k$
    - then: **return** $-(j+1)$
    - else: $i{+}{+}$; $j := (j + d) \bmod m$
- $T[ii] := x$
- **return** ii

search(T:R[m] ; k:U):$\mathbb{Z}$

- $i := 0$; $j := h_1(k)$
- $b := true$; $d := h_2(k)$
- while b
  - if $T[j].k = k$
    - then: **return** j
    - else: SKIP
  - $i{+}{+}$
  - $b := (T[j].k \neq E \wedge\, i < m)$
  - $j := (j + d) \bmod m$
- **return** -1

delete(T:R[] ; k:U):$\mathbb{Z}$

- $j :=$ search(T, k)
- if $j \geq 0$
  - then: $T[j].k := D$
  - else: SKIP
- **return** j

NYÍLT CÍMZÉS

T.f. nincs törlés $\langle h(k,0), \dots, h(k,m-1)\rangle$ perm $\langle 0, \dots, m-1\rangle$

(uniform hashing)

D. $h : U \times [0..m) \to \{0..m-1\}$ egyenletes hasítás, ha tetsz. ismeretlen k kulcsra tetsz. próbasorozat $\approx \frac{1}{m!}$ valószínűséggel áll elő

$\left(\alpha = \frac{n}{m}\right)$

T. h egyenletes hasítás $\wedge$ $\alpha \in (0;1)$ $\wedge$ nincs törölt rés a táblán.

$\Rightarrow$ sikertelen keresés, sikeres beszúrás: várható hossza* $\le \frac{1}{1-\alpha}$

sikeres keresés, sikertelen beszúrás: — " — $\le \frac{1}{\alpha} \cdot \ln \frac{1}{1-\alpha}$

Pl. $\alpha = 0.8 \Rightarrow \frac{1}{1-\alpha} = 5 \wedge \frac{1}{\alpha} \cdot \ln \frac{1}{1-\alpha} \approx 2$

*Az aktuális próbasorozat várható hossza

Tapasztalat: A kettős hasításnál a keresések/beszúrások akt. próbasorozatainak átlagos hossza közel van az egyenletes hasításra kiszámolt felső becslésekhez.