

# Adatszerkezetek

## 10. Hasító (hash) táblák

Vekov Géza

2023. május 3.



# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések  
Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete  
Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Feladat

Tároljuk el a barátaink telefonszámát úgy, hogy minnél könnyebben hozzá tudjunk férni.

# Hasító (hash) tábla

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések  
Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Feladat

Tároljuk el a barátaink telefonszámát úgy, hogy minnél könnyebben hozzá tudjunk férni.

1	0755548975
2	0745698102
3	0740123497
4	0771236458
5	0756585820
6	0744147852
...	...

**Ötlet:** tároljuk a telefonszámokat egy tömbben

# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések  
Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete  
Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Feladat

Tároljuk el a barátaink telefonszámát úgy, hogy minnél könnyebben hozzá tudjunk férni.

1	0755548975
2	0745698102
3	0740123497
4	0771236458
5	0756585820
6	0744147852
...	...

**Ötlet:** tároljuk a telefonszámokat egy tömbben

**De:** Célszerű lenne név-telefonszám párokat eltárolni

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

# Hasító (hash) tábla

# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Adatok

A hasító (hash) tábla a tömb általánosítása:

- Az "index" bármilyen érték lehet.
- **Kulcs-érték** párokat tárolunk el (pl. név-telefonszám), ahol
  - a **kulcs** az index,
  - az **érték** a táblában tárolt adat
- **Fontos:** több lehetséges kulcs van, mint hely a hasító (hash) táblában.

# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Adatok

A hasító (hash) tábla a tömb általánosítása:

- Az “index” bármilyen érték lehet.
- **Kulcs-érték** párokat tárolunk el (pl. név-telefonszám), ahol
  - a **kulcs** az index,
  - az **érték** a táblában tárolt adat
- **Fontos:** több lehetséges kulcs van, mint hely a hasító (hash) táblában.

## Cél

- Eltárolni egy (folyamatosan változó) adathalmazt

# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Műveletek

- Beszúrás (\*)
- Törlés (\*)
- Keresés (\*)
- \* adott kulcs alapján

## Megjegyzés

$O(1)$

- ha megfelelően van implementálva
- nem "rossz" bemenetre



# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Műveletek

- Beszúrás (\*) -  $O(1)$
- Törlés (\*) -  $O(1)$
- Keresés (\*) -  $O(1)$
- \* adott kulcs alapján

## Megjegyzés

$O(1)$

- ha megfelelően van implementálva
- nem "rossz" bemenetre

# Hasító (hash) tábla

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Adott

$U$  - univerzum, a lehetséges kulcsok halmaza

- általában nagyon nagy méret

# Hasító (hash) tábla

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Adott

$U$  - univerzum, a lehetséges kulcsok halmaza

- általában nagyon nagy méret

## Cél

Eltárolni egy  $S \subseteq U$  adathalmazt

- általában reális méret

# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Adott

$U$  - univerzum, a lehetséges kulcsok halmaza

- általában nagyon nagy méret

## Cél

Eltárolni egy  $S \subseteq U$  adathalmazt

- általában reális méret

## Megoldás

- $n$  - cellák száma
- hasító (hash) függvény:  $h : U \rightarrow 0, 1, \dots, n - 1$
- A hasító (hash) tábla egy  $n$  **elemű tömb**:
  - $x$ -et a  $h(x)$ -dik helyen tároljuk

# Hasító (hash) tábla

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

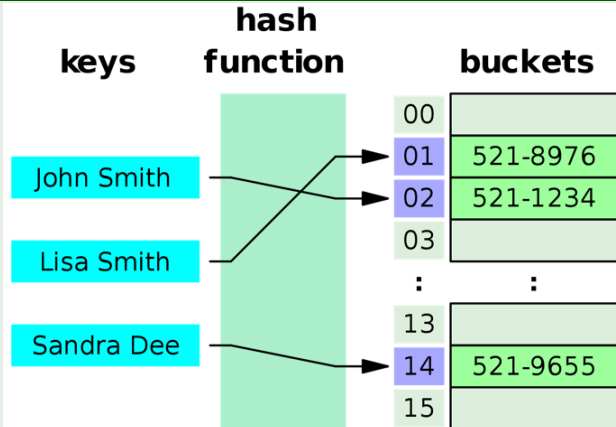
Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa



# Felmerülő kérdések

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

- Hogyan kezeljük az ütközéseket?
- Mekkora legyen a hasító (hash) tábla mérete?
- Mi történik, ha megtelik a hasító (hash) tábla?
- Milyen hasító (hash) függvényt használjunk?

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

### Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete  
Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

# Ütközések

## Ütközés

különböző  $x, y \in U : h(x) = h(y)$

- Az ütközések előfordulása nagyon gyakori, szinte minden esetben fellép

## Kezelési stratégiák

- Láncolás, keresőfa
  - a hasító (hash) tábla egyes indexein egy-egy láncolt listát tárolunk
  - a hasító (hash) tábla egyes indexein egy-egy PF fát tárolunk
- Nyílt címzés
  - egy indexen kizárólag egy értéket tárolunk
  - több megoldás létezik (lineáris próba, kvadratikus próba, kettős hashelés, stb.)



# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

- A hasító (hash) táblában minden elem valójában egy láncolt lista
- A műveletek kiegészülnek azzal, hogy az aktuális kulcs által meghatározott pozíción levő listát be kell járni.

## Futási idő

Keresés, törlés:

- Ideális esetben:  $O(1)$
- Legrosszabb esetben:  $O(n)$

## Megjegyzés

- Elvileg szükség lehet **rendezett listára**, hogy gyorsabb legyen a keresés, DE
- Ha jó a hasító (hash) függvény, akkor a listák rövidek lesznek, és nincs szükség a rendezésre, mert a keresés így is nagyon gyors lesz.

# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	
1	
2	
3	
4	
5	
6	

# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

■ Beszúr(50)

0	
1	50
2	
3	
4	
5	
6	

# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	
3	
4	
5	
6	

■ Beszúr(700)

# Láncolás

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	
3	
4	
5	
6	76

■ Beszúr(76)

# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50 → 85
2	
3	
4	
5	
6	76

■ Beszúr(85)

# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50 → 85 → 92
2	
3	
4	
5	
6	76

- Beszúr(92)



# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50 → 85 → 92
2	
3	73
4	
5	
6	76

■ Beszúr(73)

# Láncolás

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50 → 85 → 92
2	
3	73 → 101
4	
5	
6	76

■ Beszúr(101)

# Nyílt címzés

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Nyílt címzés

- minden elemet ténylegesen a hasító (hash) táblában tárolunk
- minden cellában egyetlen elem található
- ütközés esetén alternatív indexet keresünk a beszúrandó elemnek
  - pl. Hozzáadunk 1-et az indexhez

# Nyílt címzés - Műveletek

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Beszúrás( $k$ )

- addig “próbálkozunk”, amíg találunk egy üres cellát ahol elhelyezhetjük  $k$ -t

## Pédául

Beszúrandó elem:  $x$

Tábla mérete:  $n$

- ha a  $[h(x)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h(x) + 1)\%n]$  cellában
- ha a  $[(h(x) + 1)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h(x) + 2)\%n]$  cellában
- ha a  $[(h(x) + 2)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h(x) + 3)\%n]$  cellában
- ...

# Nyílt címzés - Műveletek

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete  
Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Keresés( $k$ )

- addig “próbálkozunk”, amíg megtaláljuk a  $k$ -t, vagy egy üres cellához érünk.

## Törlés( $k$ )

- ha csak egyszerűen törölünk egy kulcsot, akkor a törlés után *a keresés nem biztos, hogy sikerrel jár.*
- tényleges törlés helyett, csupán megjelöljük a cellát “*törölt*”-ként

## Megjegyzés

- beszúráskor helyezhetünk el elemet “törölt” cellába
- kereséskor ha “törölt” cellát találunk, akkor nem állítjuk le a keresést, hanem tovább haladunk amíg vagy megtaláljuk a keresett elemet, vagy egy üres cellát találunk.

# Nyílt címzés - lineáris próba

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Lineáris próba

- Lineárisan “próbálkozunk” a következő cellával
- A leggyakoribb “rés” az elemek között 1

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	
1	
2	
3	
4	
5	
6	

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

■ Beszúr(50)

0	
1	50
2	
3	
4	
5	
6	



# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	
3	
4	
5	
6	

■ Beszúr(700)

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény:  $\text{kulcs} \bmod 7$  (tábla mérete: 7)

0	700
1	50
2	
3	
4	
5	
6	76

■ Beszúr(76)

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény:  $\text{kulcs} \bmod 7$  (tábla mérete: 7)

0	700
1	50
2	85
3	
4	
5	
6	76

- Beszúr(85) - ütközés az 1-es indexen

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	92
4	
5	
6	76

- Beszúr(92) - ütközés az 1-es indexen

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	92
4	73
5	
6	76

- Beszúr(73) - ütközés az 3-as indexen

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	92
4	73
5	101
6	76

- Beszúr(101) - ütközés az 3-as indexen

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény:  $\text{kulcs} \bmod 7$  (tábla mérete: 7)

0	700
1	50
2	85
3	92
4	73
5	101
6	76

Töröljük a 85-öt!

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	92
4	73
5	101
6	76

Töröljük a 85-öt!

■  $h(85) = 1$



# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	92
4	73
5	101
6	76

Töröljük a 85-öt!

- $h(85) = 1$
- $A[1] = 85$  ?

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

Töröljük a 85-öt!

- $h(85) = 1$
- $A[1] = 85$  ?
- $A[2] = 85$  ?  $\rightarrow A[2] = \text{törölt}$

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény:  $\text{kulcs} \bmod 7$  (tábla mérete: 7)

Keressük a 92-t!

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

Keressük a 92-t!

- $h(92) = 1$

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

Keressük a 92-t!

- $h(92) = 1$
- $A[1] = 92$  ?

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

Keressük a 92-t!

- $h(92) = 1$
- $A[1] = 92$  ?
- $A[2] = 92$  ?

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

Keressük a 92-t!

- $h(92) = 1$
- $A[1] = 92$  ?
- $A[2] = 92$  ? nem, de törölt, ezért folytatjuk

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	törölt
3	92
4	73
5	101
6	76

Keressük a 92-t!

- $h(92) = 1$
- $A[1] = 92$  ?
- $A[2] = 92$  ? nem, de törölt, ezért folytatjuk
- $A[3] = 92$  ? → megtaláltuk



# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény:  $\text{kulcs} \bmod 7$  (tábla mérete: 7)

Szúrjuk be a 70-et!

0	700
1	50
2	85
3	törölt
4	73
5	101
6	76

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	törölt
4	73
5	101
6	76

Szúrjuk be a 70-et!

- $h(70) = 0$

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	törölt
4	73
5	101
6	76

Szúrjuk be a 70-et!

- $h(70) = 0$
- $A[0]$  *foglalt*

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	törölt
4	73
5	101
6	76

Szúrjuk be a 70-et!

- $h(70) = 0$
- $A[0]$  *foglalt*
- $A[1]$  *foglalt*

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	törölt
4	73
5	101
6	76

Szúrjuk be a 70-et!

- $h(70) = 0$
- $A[0]$  foglalt
- $A[1]$  foglalt
- $A[2]$  foglalt

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	törölt
4	73
5	101
6	76

Szűrjük be a 70-et!

- $h(70) = 0$
- $A[0]$  *foglalt*
- $A[1]$  *foglalt*
- $A[2]$  *foglalt*
- $A[3]$  törölt

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Példa

- kulcsok: 50, 700, 76, 85, 92, 73, 101
- hasító (hash) függvény: kulcs mod 7 (tábla mérete: 7)

0	700
1	50
2	85
3	70
4	73
5	101
6	76

Szűrjük be a 70-et!

- $h(70) = 0$
- $A[0]$  *foglalt*
- $A[1]$  *foglalt*
- $A[2]$  *foglalt*
- $A[3]$  törölt  $\rightarrow A[3] = 70$

# Nyílt címzés - lineáris próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Hátrányok

### Csoportosulás (clustering)

- Sok egymás utáni elem csoportosulhat, és egyre nehezebb lesz üres cellát találni a táblában
- A keresési idő megnő a csoportosulások kialakulásával



# Nyílt címzés - kvadratikus próba

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Kvadratikus próba

- Minden  $i$ . iterációban az  $i^2$ . cellával “próbálkozunk”

## Például

Beszúrandó elem:  $x$

Tábla mérete:  $n$

- ha a  $[h(x)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h(x) + 1 \cdot 1)\%n]$  cellában
- ha a  $[(h(x) + 1 \cdot 1)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h(x) + 2 \cdot 2)\%n]$  cellában
- ha a  $[(h(x) + 2 \cdot 2)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h(x) + 3 \cdot 3)\%n]$  cellában
- ...

# Nyílt címzés - kvadratikus próba

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Megjegyzés

- Nem jön létre csoportosulás.
- Csak akkor garantált a működés, ha
  - a tábla csak félig telik meg
  - a tábla mérete prímszám

# Nyílt címzés - dupla hasítás

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hasítás -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Kettős hashelés

- Két hasítófüggvényt használunk:
  - $h_1$  határozza meg a kezdeti indexet
  - $h_2$  határozza meg a lépésközt, amivel egy következő szabad pozíciót keresünk

## Például

Beszúrando elem:  $x$

Tábla mérete:  $n$

- ha a  $[h(x)\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h_1(x) + 1 \cdot h_2(x))\%n]$  cellában
- ha a  $[(h_1(x) + 1 \cdot h_2(x))\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h_1(x) + 2 \cdot h_2(x))\%n]$  cellában
- ha a  $[(h_1(x) + 2 \cdot h_2(x))\%n]$  cella foglalt, akkor megpróbáljuk elhelyezni  $x$ -et a  $[(h_1(x) + 3 \cdot h_2(x))\%n]$  cellában
- ...

# Nyílt címzés - dupla hasítás

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Megjegyzés

- Csak akkor garantált a működés, ha
  - $h_2$  soha nem lehet nulla, mert végtelen ciklust kapnánk
  - a tábla mérete prímszám vagy kettő hatványa
  - Ha a tábla mérete kettő hatványa, akkor a lépésköz páratlan legyen.

# Láncolás vs. nyílt címzés

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hasítás -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Összehasonlítás

Láncolás	Nyílt címzés
[+] Egyszerűbb implementálni	[-] Bonyolultabb implementálni
[+] A hasító (hash) tábla soha nem telik meg	[-] A hasító (hash) tábla betelhet
[+] Kevésbé érzékeny a hasító (hash) függvény "minőségére"	[-] A hasító (hash) függvényt úgy kell megtervezni, hogy elkerüljük a csoportosulást
[+] Többnyire akkor használjuk, amikor nem tudjuk előre, hogy milyen mennyiségű és típusú kulcsunk van	[+] Többnyire akkor használjuk, amikor ismerjük előre a kulcsok mennyiségét, és azok típusát.
[-] Extra tárhelyigény a láncolás miatt	[+] Nincsen láncolás
[-] Helypazarlás: bizonyos cellákat soha nem használunk	[+] Az ütközések miatt, egyébként üresen maradó cellákat is felhasználunk
[-] Összességében gyengébb teljesítmény, a láncolt lista használata miatt	[+] Jobb teljesítmény, mivel minden adat egy táblában van

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

# Hasító (hash) tábla mérete

## Teljesítmény

Ha sok az ütközés, és telített a hash táblánk, akkor a teljesítmény romlik:

- Minél több elemet tárolunk el, annál gyakoribbak lesznek az ütközések.
- **Láncolás:** hosszabb listák → hosszabb keresési idő
- **Nyílt címzés:** telített tábla → nehezebb üres helyet találni

Mindkét esetben: időigényesebb műveletek

## Teljesítmény

Ha sok az ütközés, és telített a hash táblánk, akkor a teljesítmény romlik:

- Minél több elemet tárolunk el, annál gyakoribbak lesznek az ütközések.
- **Láncolás:** hosszabb listák → hosszabb keresési idő
- **Nyílt címzés:** telített tábla → nehezebb üres helyet találni

Mindkét esetben: időigényesebb műveletek

## Cél

$O(1)$  idejű műveletek



## Telítettség

A hasítótábla telítettsége - telítettségi tényező:

$$\alpha = \text{eltárolt objektumok száma} / \text{cellák száma}$$

- A teljesítmény az  $\alpha$  telítettségi tényező növekedésével arányosan csökken.
- A hatékony működéshez elengedhetetlen, hogy a tábla telítettségét kontroll alatt tartsuk:

→ újra-hashelés

## Telítettség

Hash táblák esetén, melyik ütközés kezelési stratégia használható akkor is, ha a hash táblánk megtelt (azaz a telítettségi arány,  $\alpha \geq 1$ ).

- nyílt címzés
- láncolás
- egyik sem
- mindkettő

→ [www.menti.com](http://www.menti.com) - 2019 1811

# Újra-hashelés - *rehashing*

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás  
Nyílt címzés

Hasító (hash)  
tábla mérete

**Újra-hashelés -  
rehashing**

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Újra-hashelés

- A tábla újraméretezését jelenti

# Újra-hashelés - *rehashing*

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Újra-hashelés

- A tábla újraméretezését jelenti
- Minden beszúráskor **ellenőrizzük a telítettséget**
- Amennyiben a telítettség meghalad egy előre meghatározott korlátot (alapértelmezetten 0.75), akkor szükséges az **újra-hashelés**:

# Újra-hashelés - *rehashing*

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
*rehashing*

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Újra-hashelés

- A tábla újraméretezését jelenti
- Minden beszúráskor **ellenőrizzük a telítettséget**
- Amennyiben a telítettség meghalad egy előre meghatározott korlátot (alapértelmezetten 0.75), akkor szükséges az **újra-hashelés**:
  - Létrehozunk egy új táblát, mely kétszer akkora, mint az eredeti tábla
  - Bejárjuk az eredeti táblát, és minden elemet beszúrunk az új táblába (a hasítófüggvényben csak a tábla méretét változtatjuk meg)
  - Töröljük a régi táblát

Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

## Hasító (hash) függvény - mellékelve

## Adatszerkezetek

Vekov Géza

Hasító (hash)  
tábla

Ütközések

Láncolás

Nyílt címzés

Hasító (hash)  
tábla mérete

Újra-hashelés -  
rehashing

Hasító (hash)  
függvény -  
mellékelve

Alkalmazások -  
mellékelve

# Alkalmazások - mellékelve