2. előadás

Struktúra nélküli adatszerkezetek

Reprezentáció és implementáció, a szabad helyek kezelése, a halmaz és a multihalmaz

Adatszerkezetek és algoritmusok előadás 2011. február 16.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz A multihalmaz

Kósa Márk és Pánovics János Debreceni Egyetem Informatikai Kar

Reprezentáció és implementáció

Absztrakt adatszerkezetek grafikus (képi) megjelenítésénél használt jelölések:

adatelem

kapcsolat két adatelem között

Amikor egy absztrakt adatszerkezethez megadjuk a tárolási módját és a leképezését, akkor megadjuk az absztrakt adatszerkezet reprezentációját.

absztrakt adatszerkezet $\xrightarrow{\text{lek\'epez\'es}}$ ábrázolás

Ha a reprezentáció mellé megadjuk a műveletek megvalósítását is, akkor megadjuk az absztrakt adatszerkezet implementációját.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Algoritmusok megadásának módjai

adatszerkezetek Kósa Márk Pánovics János

Struktúra nélküli



Reprezentáció és implementáció

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek

A halmaz

A multihalmaz

A műveletek algoritmusok. Algoritmusokat a következő módokon lehet megadni:

- természetes (beszélt) emberi nyelven
- mondatszerű leírással
- folyamatábrával
- pszeudonyelvvel (lásd gyakorlaton)
- valamilyen programozási nyelven

Példa elefánttal és zsiráffal

Hogyan tegyünk be egy elefántot a hűtőszekrénybe?

- 1 Nyissuk ki a hűtőszekrény ajtaját.
- 2 Tegyük be az elefántot a hűtőszekrénybe.
- 3 Zárjuk be a hűtőszekrény ajtaját.

Hogyan tegyünk be egy zsiráfot a hűtőszekrénybe?

- 1 Nyissuk ki a hűtőszekrény ajtaját.
- 2 Vegyük ki az elefántot a hűtőszekrényből.
- 3 Tegyük be a zsiráfot a hűtőszekrénybe.
- 4 Zárjuk be a hűtőszekrény ajtaját.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

algoritmusok medadásának módiai

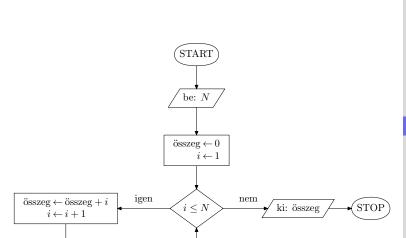
Gazdálkodás a

Gazdalkodas a memóriával

A multihalmaz

Struktúra nélküli adatszerkezetek A halmaz

Példa folyamatábrával



Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

megadásának módiai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz A multihalmaz

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Struktúra nélküli adatszerkezetek A halmaz

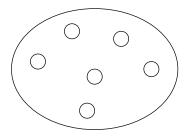
A multihalmaz

Az adatszerkezetek tárolásához memória kell, ami véges. A szabad helyekkel gazdálkodni kell! Lehetséges módszerek

- folytonos ábrázolás esetén:
 - szabad tárhelyek összegyűjtése a lefoglalt tárterület végén (időigényes)
 - szemétgyűjtögetés (garbage collection) elemmozgatással
 - minden tárhelyhez hozzárendelünk egy bitet, ami a foglaltságot jelzi (nincs elemmozgatás)
- szétszórt ábrázolás esetén:
 - szabad helyek láncolt listája (probléma: különböző méretű tárhelyek)
 - szemétgyűjtögetés (garbage collection) a szabad helyek láncolásával
 - a szabad helyek nyilvántartása bitvektor segítségével

Halmaz és multihalmaz

A halmaz és a multihalmaz struktúra nélküli, homogén és dinamikus adatszerkezetek.



A halmaz minden eleme különböző. A multihalmazban előfordulhatnak azonos elemek is.

Mindkét adatszerkezetre igaz, hogy az adatszerkezetben lévő elemek között nincs kapcsolat (ezért struktúra nélküli adatszerkezetek).



Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli

A halmaz A multihalmaz

A halmaz adatszerkezet

A halmaz adatszerkezet a matematikai halmaz fogalom megjelenése az adatszerkezetek szintjén. Mindig véges – ennyiben nem felel meg teljesen a matematikai halmaz fogalmának.

A halmaz alapműveletei

- eleme, ∈: megmondja, hogy egy adatelem benne van-e a halmazban vagy sem
- unió, ∪: két halmaz unióját adja
- metszet, ∩: két halmaz metszetét adja
- különbség, \: két halmaz különbségét adja

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek

A halmaz

A halmaz adatszerkezet

Struktúra nélküli adatszerkezetek Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli. adatszerkezetek

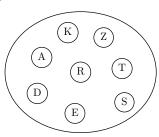
A halmaz

A multihalmaz

Az adatszerkezetekkel végezhető hagyományos műveletek megvalósítása halmazok esetén:

- Létrehozás kétféleképpen:
 - explicit módon, a halmaz elemeinek felsorolásával (esetleg) üresen)
 - egy predikátum segítségével
- Bővítés unióképzéssel
- Törlés csak fizikai, különbségképzéssel
- Csere nincs
- Rendezés, keresés, elérés, bejárás nem értelmezettek
- Feldolgozás a halmaz alapműveleteinek a segítségével

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

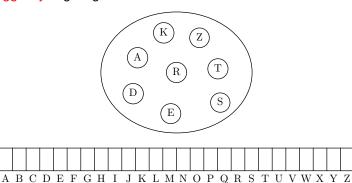
Algoritmusok megadásának módjai

Gazdálkodás a memóriával Struktúra nélküli

Struktúra nélküli adatszerkezetek

A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A halmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy egy bit méretű tárterületet.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

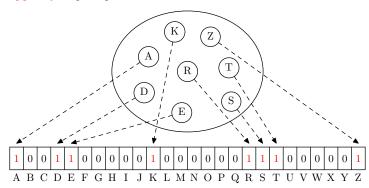
Algoritmusok megadásának módjai

Gazdálkodás a memóriával Struktúra nélküli

adatszerkezetek

A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A halmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy egy bit méretű tárterületet. Az adott értékű adatelemhez tartozó bit fogja jelezni, hogy az adatelem benne van-e a halmazban (1) vagy sem (0).

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek

A halmaz

A halmaz adatszerkezet implementációja

Folytonos reprezentáció esetén a halmaz alapműveleteinek megvalósítása visszavezethető egyszerű bitműveletekre:

Unióképzés

$$x \in A \cup B \Leftrightarrow x \in A \lor x \in B$$

Metszetképzés

$$x \in A \cap B \Leftrightarrow x \in A \land x \in B$$

Különbségképzés

$$x \in A \setminus B \Leftrightarrow x \in A \land x \notin B$$

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek

A halmaz A multihalmaz

A halmaz adatszerkezet megjelenése adattípusként a programozási nyelvekben

Java

Set interfész, melynek műveletei (többek között):

- isEmpty()
- contains(Object o), containsAll(Collection<?> c)
- add(E e), addAll(Collection<? extends E> c)
- remove(Object o), removeAll(Collection<?> c)
- size()

C++

set generikus osztály, melynek műveletei (többek között):

- empty()
- size()
- insert (...) többféle specifikációval
- erase(...) többféle specifikációval
- find(const key_type& x)

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek

A halmaz

A halmaz adatszerkezet megjelenése adattípusként a programozási nyelvekben

C#

HashSet<T> generikus kollekció osztály, melynek műveletei (többek között):

- Add(T item)
- Remove(T item)
- Clear()
- UnionWith(IEnumerable<T> other)
- IntersectWith(IEnumerable<T> other)
- ExceptWith(IEnumerable<T> other)
- Contains (T item)
- IsSubsetOf(IEnumerable<T> other)
- IsSupersetOf(IEnumerable<T> other)
- Overlaps(IEnumerable<T> other)
- SetEquals(IEnumerable<T> other)

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek

A halmaz

A multihalmaz adatszerkezet

A multihalmaz abban különbözik a halmaztól, hogy megengedi az adatelemek ismétlődését, benne több azonos értékű elem is előfordulhat.

A multihalmaz alapműveletei

- eleme, ∈: megmondja, hogy egy adatelem benne van-e a multihalmazban vagy sem
- unió, ∪: két multihalmaz unióját adja
- metszet, ∩: két multihalmaz metszetét adja
- különbség, \: két multihalmaz különbségét adja

Multihalmazoknál az adatszerkezetekkel végezhető hagyományos műveletek megvalósítása hasonló a halmazokéhoz (lásd ott). A multihalmaz feldolgozása a multihalmaz alapműveleteinek a segítségével történik.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



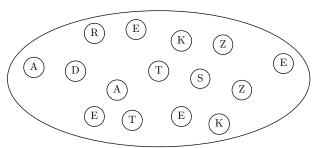
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



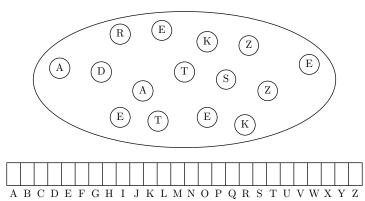
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot).

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



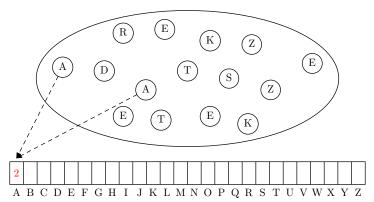
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



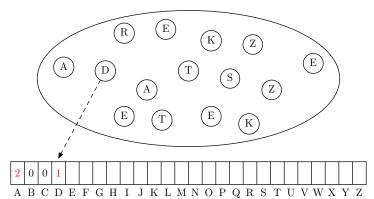
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



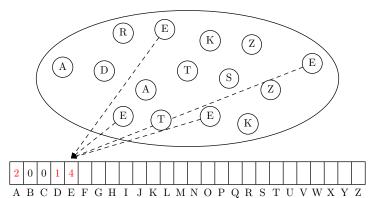
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



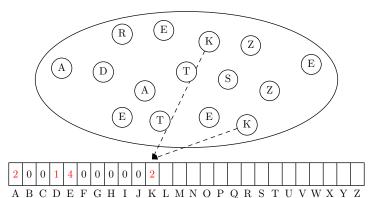
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



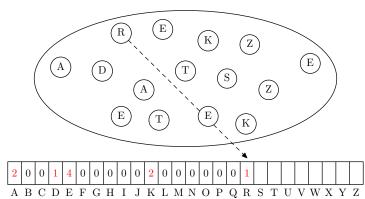
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



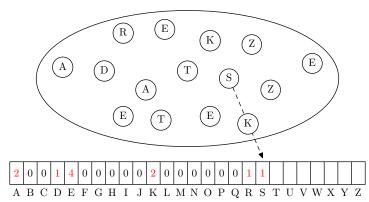
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



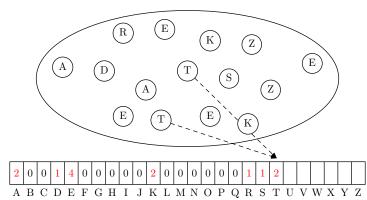
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



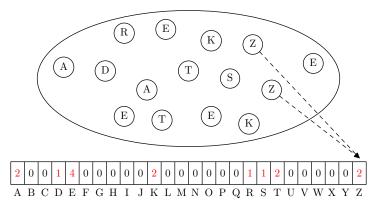
Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Klasszikus reprezentációja folytonosan, karakterisztikus függvény segítségével történik.



A multihalmaz lehetséges elemeit sorba rendezzük, s mindegyikhez hozzárendelünk egy tárterületet (általában 1 bájtot). A tárhelyeken az azonos értékű elemek előfordulásainak számát tároljuk.

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz

Folytonos reprezentáció esetén a multihalmaz alapműveleteinek megvalósítása visszavezethető egyszerű aritmetikai (számtani) műveletekre:

Unióképzés

$$x A \cup B$$
-ben = $x A$ -ban + $x B$ -ben

Metszetképzés

$$x A \cap B$$
-ben = min{ $x A$ -ban, $x B$ -ben}

Különbségképzés

$$x A \setminus B$$
-ben = max $\{0, x A$ -ban - $x B$ -ben $\}$

Struktúra nélküli adatszerkezetek

Kósa Márk Pánovics János



Reprezentáció és implementáció

Algoritmusok megadásának módjai

Gazdálkodás a memóriával

Struktúra nélküli adatszerkezetek A halmaz