

30. Összefuttató felsorolók

- Adott két rendezett sorozat, elő kell állítani a kimenetet

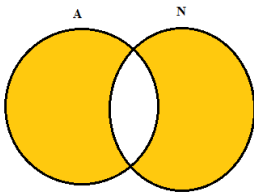
Az alábbi minta feladatokon keresztül példákat láthatunk a metszet, a szimmetrikus differenciál és a különbség halmazműveletekre vonatkozó feladatokra egy-egy egyszerűbb példát, amilyen a tavalyi zárthelyi dolgozatban is találkozhattunk, továbbá egy nehezebb feladatot (record-os) az unió műveletre.

30.1 Szimmetrikus differencia

Bemenet: két rendezett sorozat

- első file tartalmazza az angolul tudó hallgatók neveit
- második file tartalmazza a németül tudó hallgatók neveit

Kik azok a hallgatók, akik pontosan egy nyelvet beszélnek?
(Szimmetrikus differencia)



Specifikáció:

$$A = (a: \text{infile}(\mathbb{K}^*), n: \text{infile}(\mathbb{K}^*), z: \text{outfile}(\mathbb{K}^*))$$

$$Ef = (a = a' \wedge n = n' \wedge a \uparrow \wedge n \uparrow)$$

$$Uf = (z = \bigoplus_{e \in a \cup n} f(e))$$

$$f(e) = \begin{cases} \langle e \rangle & e \in a \wedge e \notin n \\ \langle e \rangle & e \notin a \wedge e \in n \\ \langle \rangle & e \in a \wedge e \in n \end{cases}$$

Megjegyzés: $f(e)$

- *összefuttató felsoroló (a mi esetünkben most a szimmetrikus differenciára)*

$z := \langle \rangle \quad sa, da, a: read \quad sn, dn, n: read$		
$sa = norm \vee sn = norm$		
$(sn = abnorm) \vee$ $(sa = norm \wedge da < dn)$	$sa = norm \wedge sn = norm \wedge$ $da = dn$	$(sa = abnorm) \vee$ $(sn = norm \wedge da > dn)$
$z: write(da)$ $sa, da, a: read$	$sa, da, a: read$ $sn, dn, n: read$	$z: write(dn)$ $sn, dn, n: read$

Magyarázatok a pirossal kiemelt részekről:

\vee : Általában vagyot használunk, mert mind a két fileunkat teljesen fel szeretnénk dolgozni. Abban az esetben, ha metszetet vagy különbséget kérdeznek a feladatban, akkor jobb az \wedge -t használni, mivel ezekben az esetekben ügyesebb és hatékonyabb nem feldolgozni a két fileunkat.

$da < dn$:

$(sn = abnorm) \vee$

$(sa = norm \wedge da < dn)$

Abban az esetben, ha elfogyott a németesek nevét tartalmazó fileunk vagy van még az angol neveket tartalmazó file-ban valami adatunk és névsor szerint előrébb található az adatban foglalt név, a németeseket tartalmazó file-unckhoz képest, akkor lépünk be ebbe az ágba. Itt da -t dolgozzuk fel és csak az a fileból olvasunk.

$da = dn$:

$sa = norm \wedge sn = norm \wedge$

$da = dn$

Mind két fileból feldolgozunk, ezért szükséges, hogy mind 2 file tartalmazzon adatot és ha olyan állapotba érünk, hogy az angolos és németes fileban is megtaláljuk ugyan annak a tanulóknak a nevét, akkor lépünk be ebbe az elágazásba.

Ekkor a feladat miatt (szimmetrikus differencia) a metszet nem számít, így nem iratjuk ki a hallgatót, csak mind két fileból beolvassuk a következő nevet.

$da > dn$:

$(sa = abnorm) \vee$

$(sn = norm \wedge da > dn)$

Abban az esetben, ha elfogyott az angolosok nevét tartalmazó fileunk vagy van még a német neveket tartalmazó file-ban valami adatunk és névsor szerint előrébb található az adatban foglalt név, az angolosok nevét tartalmazó file-unckhoz képest, akkor lépünk be ebbe az ágba. Itt dn -t dolgozzuk fel és csak az n fileból olvasunk.

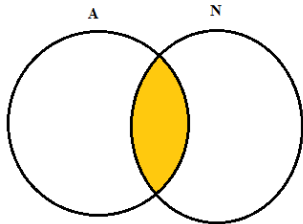
30.2 Metszet

Bemenet: két rendezett sorozat

- első file tartalmazza az angolul tudó hallgatók neveit
- második file tartalmazza a németül tudó hallgatók neveit

Kik azok, akik két nyelven beszélnek?

(Metszet)



Specifikáció:

$$A = (a: \text{infile}(\mathbb{K}^*), n: \text{infile}(\mathbb{K}^*), z: \text{outfile}(\mathbb{K}^*))$$

$$Ef = (a = a' \wedge n = n' \wedge a \uparrow \wedge n \uparrow)$$

$$Uf = (z = \bigoplus_{e \in a \cup n} f(e))$$

$$f(e) = \begin{cases} \langle \rangle & e \in a \wedge e \notin n \\ \langle \rangle & e \notin a \wedge e \in n \\ \langle e \rangle & e \in a \wedge e \in n \end{cases}$$

Algoritmus:

$z := \langle \rangle \quad sa, da, a: \text{read} \quad sn, dn, n: \text{read}$		
$sa = \text{norm} \wedge sn = \text{norm}$		
$da < dn$	$da = dn$	$da > dn$
$sa, da, a: \text{read}$	$z: \text{write}(da)$ $sa, da, a: \text{read}$ $sn, dn, n: \text{read}$	$sn, dn, n: \text{read}$

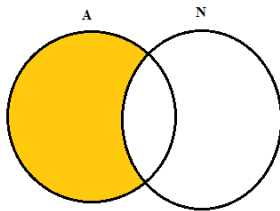
30.3 Különbség

Bemenet: két rendezett sorozat

- első file tartalmazza az angolul tudó hallgatók neveit
- második file tartalmazza a németül tudó hallgatók neveit

Kik azok, akik csak angolul beszélnek?

(Különbség)



Specifikáció:

$$A = (a: \text{infile}(\mathbb{K}^*), n: \text{infile}(\mathbb{K}^*), z: \text{outfile}(\mathbb{K}^*))$$

$$Ef = (a = a' \wedge n = n' \wedge a \uparrow \wedge n \uparrow)$$

$$Uf = (z = \bigoplus_{e \in a \cup n} f(e))$$

$$f(e) = \begin{cases} \langle e \rangle & e \in a \wedge e \notin n \\ \langle \rangle & e \notin a \wedge e \in n \\ \langle \rangle & e \in a \wedge e \in n \end{cases}$$

Algoritmus:

$z := \langle \rangle \quad sa, da, a: \text{read} \quad sn, dn, n: \text{read}$		
$sa = \text{norm}$		
$(sn = \text{abnorm}) \vee$ $(da < dn)$	$sn = \text{norm} \wedge$ $da = dn$	$(sn = \text{norm} \wedge da > dn)$
$z: \text{write}(da)$ $sa, da, a: \text{read}$	$sa, da, a: \text{read}$ $sn, dn, n: \text{read}$	$sn, dn, n: \text{read}$

30.4 Összetettebb uniós

Adott egy zh-kat és egy pót zh-kat tartalmazó file.

Az adott fileban: - EHA kód és pont szerepel

EHA kód szerint szigorúan monoton növekvő sorrendben

Mi lett az adott diák eredménye, ha a jobbik pontszám számít?

(Unió)

Specifikáció:

$tanuló = rec(Eha: \mathbb{K}^*, pont: \mathbb{N})$

$A = (zh: infile(tanuló), pót: infile(tanuló), eredmény: outfile(tanuló))$

$Ef = (zh = zh' \wedge pót = pót' \wedge zh \uparrow Eha \wedge pót \uparrow Eha)$

$Uf = (eredmény = \bigoplus_{e.Eha \in Eha(zh') \cup Eha(pót')} f(e))$

$$f(e) = \begin{cases} \langle e \rangle & e.Eha \in Eha(zh') \wedge e.Eha \notin Eha(pót') \\ \langle e \rangle & e.Eha \notin Eha(zh') \wedge e.Eha \in Eha(pót') \\ g(e) & e.Eha \in Eha(zh') \wedge e.Eha \in Eha(pót') \end{cases}$$

$g(e)$ – t két féle képpen lehet felírni. Szabadon választható az egyik.

1.:

$g(e) = e(zh').pont \leftarrow \max(e(zh').pont, e(pót').pont)$

A felírás: $e(zh')$ - vegyük a hallgatót a zh' -ből

$pont \leftarrow \max(e(zh').pont, e(pót').pont)$ Módosítsuk a pont rekordot, a zh' és $pót'$ közül a maximálisra.

2.:

$g(e) = \langle e(zh').Eha; \max(e(zh').pont, e(pót').pont) \rangle$

Algoritmus:

$eredmény := \langle \rangle \quad szh, dzh, zh: read \quad spot, dpot, pot: read$		
$szh = norm \vee spot = norm$		
$(spot = abnorm) \vee$ $(szh = norm \wedge$ $dzh.eha < dpot.eha)$	$szh = norm \wedge$ $spot = norm \wedge$ $dzh.eha = dpot.eha$	$(szh = abnorm) \vee$ $(spot = norm \wedge$ $dzh.eha > dpot.eha)$
$eredmény: write(dzh)$ $szh, dzh, zh: read$	$dzh.pont :=$ $\max(dzh.pont, dpot.pont)$ $eredmény: write(dzh)$ $szh, dzh, zh: read$ $spot, dpot, pot: read$	$eredmény: write(dpot)$ $spot, dpot, pot: read$

Készítette: Nagy Krisztián (HTANBW)

[RÉSZLET A KÖNYVEMBŐL]