

6. táblás gyakorlat feladatai

1. Adott egy egész számokat tartalmazó gyűjtemény.

a) Hány páros szám előzi meg az első negatívát?

Specifikáció:

$A = (x: \text{enor}(\mathbb{Z}), db: \mathbb{N})$

$Ef = (x = x_0)$

$Uf = (db = \sum_{\substack{e \in x_0 \\ e \text{ páros}}}^{\geq 0} 1)$

Amikor a számlálás leáll, az e tartalmazza az utoljára felsorolt elemet (ez az első negatív szám, ha van ilyen); az x pedig a még nem felsorolt elemek sorozatára (az aktuális felsorolóra) hivatkozik.

Számlálás, feltétel fennállásáig

$t: \text{enor}(E) \sim x: \text{enor}(\mathbb{Z})$ amíg: $e \geq 0$

$\text{felt}(e) \sim e \text{ páros}$

$c \sim db$

Algoritmus:

db := 0		e:ℤ
e in x ∧ e≥0		
e páros		
db := db+1	—	

db := 0		e:ℤ
x.First()		
¬x.End() ∧ x.Current() ≥ 0		
x.Current() páros		
db := db+1	—	
x.Next()		

b) Hány páros szám követi az első negatív számot?

Specifikáció:

$A = (x: \text{enor}(\mathbb{Z}), db: \mathbb{N})$

$Ef = (x = x_0)$

$Uf = ((\dots, x') = \text{SEARCH}_{e \in x_0}(e < 0) \wedge db = \sum_{\substack{e \in x' \\ e \text{ páros}}} 1)$

A linker szokásos két outputja: egy logikai érték és az e változó értéke termináláskor) itt nem kell; csak a harmadik, ún. másodlagos outputra van szükség: x' jelöli az x felsorolás aktuális állapotát a keresés leállásakor: ezt a felsorolást folytatja majd a számlálás.

$Uf = ((\dots, x') = \text{SELECT}_{e \in x_0}(e < 0 \vee |x| = 0) \wedge db = \sum_{\substack{e \in x' \\ e \text{ páros}}} 1)$

A kiválasztástól is csak a másodlagos output kell: x' jelöli az x felsorolás aktuális állapotát a kiválasztás leállásakor.

A kiválasztás feltétele akkor teljesül biztosan, ha még vaglyagosan tartalmazza az $|x| = 0$ -t (azaz „találunk negatív elemet vagy a felsorolás végére érünk” lesz a keresés feltétele).

Algoritmus:

l := false		e:ℤ
e in x ∧ ¬l		
e < 0		
l := igaz	—	
db := 0		
e in x		
e páros		
db := db+1	—	

e in x ∧ e ≥ 0		e:ℤ
—		
db := 0		
e in x		
e páros		
db := db+1	—	

c) Hány páros szám van az első negatív számot megelőzően, és hány azt követően?

Specifikáció:

$A = (x: \text{enor}(\mathbb{Z}), \text{dbe}, \text{dbu}: \mathbb{N})$

$Ef = (x = x_0)$ 1.kvíz [Uf]

$Uf = ((\text{dbe}, \cdot, x') = \sum_{\substack{e \in x_0 \\ e \text{ páros}}}^{e \geq 0} 1 \wedge \text{dbu} = \sum_{\substack{e \in x' \\ e \text{ páros}}} 1)$

A feltételig tartó számlálásnak három outputja van: a dbe számláló; utoljára felsorolt elem (az e aktuális értéke, amire itt nincs szükség); és az x felsoroló aktuális állapota: a felsorolás hátralevő elemeinek x'-vel jelölt sorozata. A második számlálás az x' felsorolót használja, azaz folytatja a felsorolást.

Algoritmus:

dbe := 0		e:ℤ
e in x ∧ e ≥ 0		
e páros		
dbe := dbe + 1	—	
dbu := 0		
e in x		
e páros		
dbu := dbu + 1	—	

d) Hány páros szám van az első negatív számot megelőzően, és hány azt követően azzal együtt?

Specifikáció:

$A = (x: \text{infile}(\mathbb{Z}), \text{dbe}, \text{dbu}: \mathbb{N})$

$Ef = (x = x_0)$

$Uf = ((\text{dbe}, e', x') = \sum_{\substack{e \in x_0 \\ e \text{ páros}}}^{e \geq 0} 1$

$\wedge \text{dbu} = \sum_{\substack{e \in \langle e', x' \rangle \\ e \text{ páros}}} 1)$

Az e' az e változó értéke, x' az x felsoroló állapota az első számlálás után. A második számlálás úgy folytatja felsorolást, hogy figyelembe veszi az e' elemet is.

Algoritmus:

dbe := 0		e:ℤ
e in x ∧ e ≥ 0		
e páros		
dbe := dbe + 1	—	
dbu := 0		
e in <e, x>		
e páros		
dbu := dbu + 1	—	

Szekvenciális inputfájltra:

dbe := 0		e:ℤ
st, e, x : read		st : Status
st = norm ∧ e ≥ 0		
e páros		
dbe := dbe + 1	—	
st, e, x : read		
dbu := 0		
st = norm		
e páros		
dbu := dbu + 1	—	
st, e, x : read		

Tömbre:

dbe := 0		e:ℤ
i := 1		i:ℕ
i ≤ n ∧ x[i] ≥ 0		
x[i] páros		
dbe := dbe + 1	—	
i := i + 1		
dbu := 0		
i ≤ n		
x[i] páros		
dbu := dbu + 1	—	
i := i + 1		

2. Egymás utáni napok átlaghőmérsékleteit egy szekvenciális inputfájl tartalmazza. Mennyi az első fagypont alatti értéket megelőző napok (ilyenek biztosan vannak) hőmérsékleteinek átlaga, továbbá az első fagypont alatti értéktől kezdődően (az első fagypont alatti napot is beleértve) vajon minden nap fagypont alatt maradt-e a hőmérséklet, és mi volt a legalacsonyabb hőmérséklet?

Specifikáció:

$$A = (x:infile(\mathbb{R}), a:\mathbb{R}, l:\mathbb{L}, kicsi:\mathbb{R}) \quad 2.kvíz [Ef szövegesen]$$

$$Ef = (x=x_0 \wedge |x| \geq 2 \wedge x[1] \geq 0 \wedge \exists i \in [2..|x|]: x[i] < 0)$$

$$Uf = ((s, e', x') = \sum_{e \in x_0}^{e \geq 0} (e) \wedge (db, e', x') = \sum_{e \in x_0}^{e \geq 0} 1 \wedge a = s/db \wedge$$

$$\wedge l = \forall SEARCH_{e \in <e', x'>} (e < 0) \wedge kicsi = \min_{e \in (e', x')} e) =$$

Algoritmus:

$e, s:\mathbb{R}, db:\mathbb{N}$ st:Status

feltétel fennállásáig tartó két összegzés
közös ciklusban

átlagszámítás

op. lin. ker. és min. kiv. közös ciklusban
az inicializálásnál figyelembe vesszük,
hogy az első megvizsgálandó elemet
már korábban beolvastuk,
ami biztosan negatív szám

s, db := 0.0, 0	
st, e, x : read	
st=norm \wedge e \geq 0	
s, db := s+e, db +1	
st, e, x : read	
a := s / db	
l, kicsi := igaz, e	
st, e, x : read	
st=norm	
l, kicsi := l \wedge e<0, min(kicsi,e)	
st, e, x : read	

3. Egy horgászversenyen a horgászok eredményét egy szekvenciális inputfájlban rögzítették. A fájl egy eleme egy horgász nevét és a halfogásainak sorozatát tartalmazza (lásd előző gyakorlat). Gyűjtjük ki azon horgászok neveit, akik legalább négy legalább 1 méteres (100 cm) harcsát fogtak az első legalább 1 kilós ponty-fogásukat követően.

Specifikáció:

$$A = (f:infile(\text{Horgász}), lista:\mathbb{S}^*) \quad \text{Horgász} = \text{rec}(\text{név}:\mathbb{S}, \text{eredm}:\text{Fogás}^*)$$

$$\text{Fogás} = \text{rec}(\text{idő}:\mathbb{S}, \text{hal}:\mathbb{S}, \text{súly}:\mathbb{R}, \text{hossz}:\mathbb{R})$$

$$Ef = (f=f_0)$$

$$Uf = (lista = \bigoplus_{e \in f_0} <e.név> \mid \text{jó}(e.eredm))$$

Összegzés (kiválogatás)

$$t:enor(E) \sim f:infile(\text{Horgász}) \quad (st, e, f:read)$$

$$f(e) \sim <e.név> \text{ ha } \text{jó}(e.eredm)$$

$$s \sim lista$$

$$H, +, 0 \sim \mathbb{S}^*, \bigoplus, <>$$

Algoritmus:

lista := < >		st:Status e:Horgász
st, e, f : read		
st = norm		
jó(e.eredm)		
lista := lista \oplus <e.név>	—	
st, e, f : read		

Részfeladat: $ok := jó(e.fogás)$ ahol $jó : Fogás^* \rightarrow \mathbb{L}$

$A = (x:Fogás^*, ok:\mathbb{L}) \quad Fogás = rec(idő:\mathbb{S}, hal:\mathbb{S}, súly:\mathbb{R}, hossz:\mathbb{R})$

$Ef = (x=x_0)$

$Uf = (x=x_0 \wedge (e', x') = \text{SELECT}_{e \in x_0} ((e.hal="ponty" \wedge e.súly \geq 1.0) \vee \neg(x \neq \emptyset)) \wedge$
 $\wedge db = \sum_{e \in \langle e', x' \rangle} 1 \wedge ok = db \geq 4)$
 $e.hal="harcsa" \wedge e.hossz \geq 1.0$

Algoritmus: $ok := jó(x)$ ahol $jó : Fogás^* \rightarrow \mathbb{L}$

Kiválasztás

$t:enor(E) \sim x:Fogás^*$

$felt(e) \sim (e.hal="ponty" \wedge e.súly \geq 1.0)$

Számlálás

$t:enor(E) \sim x:Fogás^*$

$felt(e) \sim e.hal="harcsa" \wedge e.hossz \geq 1.0$

$c \sim db$

e in x $\wedge \neg(e.hal="ponty" \wedge e.súly \geq 1.0)$		db:N
—		
db := 0		
e in x		
<div>e.hal="harcsa" \wedge e.hossz ≥ 1.0</div>		
db := db+1	—	
ok := db ≥ 4		