

PAPIROS PRÓBA ZH MEGOLDÁSA

1. feladat:

Egy kórházban fekvő betegnél egy mérőműszer naponta többször feljegyzi a testhőmérsékletet és a vérnyomás értékeket (alsó, felső). A mért adatok egy szekvenciális input fájlban kerülnek tárolásra:

- időpont (string)
- testhőmérséklet (valós)
- vérnyomás alsó és felső érték (egész)

Az adatok időpont szerint rendezettek.

Lázasnak tekintjük a beteget 38, vagy annál magasabb mérés esetén. Tudjuk, hogy az első mérésnél lázas volt.

Mennyi volt az első lázas szakaszban (amíg folyamatosan lázat mértek), a vérnyomás felső érték maximuma? Az ezt követő láztalan szakaszban volt-e ennél magasabb a vérnyomás felső érték, ha igen, adjuk meg az elsőnek az időpontját! (Nem tehető fel, hogy volt láztalan szakasz.)

SPECIFIKÁCIÓ:

Meres = rec(ido: \mathbb{S} , h: \mathbb{R} , va,vf: \mathbb{N})

A = (f:infile(Meres), max: \mathbb{N} , l: \mathbb{L} , time: \mathbb{S})

Ef = (f = f' \wedge |f|>0 \wedge f₁.h \geq 38 \wedge f \uparrow ido)

Uf = (max, (st'',e'',f'') = $\text{MAX}_{e \in f'}^{e.h \geq 38}$ e.vf \wedge l,elem (st,e,f) = $\text{SEARCH}_{e \in (e'',f'')}^{e.h < 38}$ e.vf > max \wedge l \rightarrow time=elem.ido)

VISSZAVEZETÉS:

Maximum kiválasztás, feltétel fennállásáig

t:enor(E) ~ f:infile(Meres) (st,e,f:read)
amíg: e.h \geq 38

f(e) ~ e.vf

H,> \mathbb{N} , >

folytatás: lineáris keresés

t:enor(E) ~ f:infile(Meres) (st,e,f:read)
first() nélkül
amíg: e.h<38

felt(e) ~ e.vf>max

ALGORITMUS

st,e,f:read	
max:=e.vf	
st,e,f:read	
st=norm \wedge e.h \geq 38	
┌ e.vf > max ─┐	
max:=e.vf	—
st,e,f:read	
l:=hamis	
\neg l \wedge st = norm \wedge e.h < 38	
┌ l, elem:= e.vf > max, e ─┐	
st,e,f:read	
└──────────────────┘	
time:=elem.ido	—

2. feladat

Egy sportoló a napi edzéseiről az alábbi nyilvántartást vezette:

- hónap (egész)
- nap (egész)
- edzés időtartama (egész, perc)
- lefutott km (valós) (nem minden edzésen fut)

A fájl hónap, azon belül nap szerint rendezett.

Igaz-e, hogy minden hónapban összesen legalább 100 km-t futott? Azon hónapok közül, amelyben volt legalább egy 20 km-es futó edzés melyik hónapban edzett a legtöbbet?

Edzes = rec(ho: \mathbb{N} , nap: \mathbb{N} , ido: \mathbb{N} , futas: \mathbb{R})

A = (f:infile(Edzes), mindenho: \mathbb{L} , l: \mathbb{L} , maxido: \mathbb{N} , month: \mathbb{N})

új állapottér:

az egyedi felsoroló minden hónapra megadja a következőket:

- hónap (ho),
- összesen mennyit futott (osszFut),
- volt-e legalább egyszer 20 km-es futás (volt20)
- edzési idők összege (osszIdo)

honapOssz = rec(ho: \mathbb{N} , osszFut: \mathbb{N} , osszIdo: \mathbb{N} , volt20: \mathbb{L})

Egyedi felsoroló: t:enor(honapOssz)

Az egyedi felsorolón a feladat megoldása:

SPECIFIKÁCIÓ:

A = (t : enor(honapOssz), mindenho: \mathbb{L} , l: \mathbb{L} , maxido: \mathbb{N} , month: \mathbb{N})

Ef = (t = t')

Uf = (mindenho = $\forall \text{SEARCH}_{e \in t'} e. \text{osszFut} \geq 100 \wedge$
l,maxido,elem = $\text{MAX}_{e \in t'} e. \text{osszIdo} \wedge l \rightarrow \text{month} = \text{elem.ho}$)
e.volt20

VISSZAVEZETÉS

Összegzés (összeérelés) és feltételes maximum keresés ugyanazon a felsorolón

t:enor(E) ~ t:enor(honapOssz)

összegzés:

f(e) ~ e.osszFut \geq 100

s ~ mindenho

H,+ ,0 \mathbb{L}, \wedge , igaz

feltételes maximum keresés

f(e) ~ e.osszIdo

felt(e) ~ e.volt20

max ~ maxido

H,> ~ $\mathbb{N}, >$

ALGORITMUS

l, mindenHo := hamis, igaz t.first()		
¬ t.end()		
e:=t.current()		
¬ e.volt20	e.volt20 ∧ l	e.volt20 ∧ ¬ l
–	e.osszldo>maxido	l,month,maxido:= igaz, e.ho, e.osszldo
	month,maxido:= e.ho,e.osszldo	
mindenho := mindenho ∧ e.osszFut≥100		
t.next()		

Megjegyzések:

- e:honapOssz lokális segéd változó – célszerű a t.current() értéket egyszer kiolvasni egy lokális változóba, majd azzal dolgozni a tétel ciklusában.
- A feltételes maximum keresés „elem” változója helyett közvetlenül a „month” eredmény változót használjuk, így a tétel után nincs szükség az elágazásra.

Az egyedi felsoroló terve:

t:enor(honapOssz) honapOssz = rec(ho: ℕ, osszFut: ℕ, osszldo: ℕ, volt20: ℒ)

honapOssz*	first()	next()	current() : honapOssz	end() : ℒ
f : infile(Edzes) df : Edzes st : Status akt : honapOssz vége : ℒ	st,df,f:read next()	lásd külön	return akt	return vége

Edzes = rec(ho: ℕ, nap: ℕ, ido: ℕ, futas: ℝ)

next() művelet:

SPECIFIKÁCIÓ:

$A = (f:infile(Edzes), df:Edzes, st:Status, akt:honapOssz, vége:ℒ)$

$Ef = (f = f' \wedge f \nearrow (ho,nap) \wedge df = df' \wedge st = st')$

$Uf = (vége = (st'=abnorm) \wedge (\neg vége \rightarrow akt.ho=df'.ho \wedge$

$df.ho=akt.ho$
 $\wedge (akt.osszFut, (st,df,f)) = \sum_{df \in (df',f')} df.futas \wedge$

$df.ho=akt.ho$
 $\wedge (akt.osszldo, (st,df,f)) = \sum_{dx \in (df',f')} df.ido \wedge$

$df.ho=akt.ho$
 $\wedge (akt.volt20, (st,df,f)) = \bigvee_{df \in (df',f')} df.futas \geq 20))$

VISSZAVEZETÉS

Három összegzés (egy vagyolás és két összeadás) közös felsorolóval

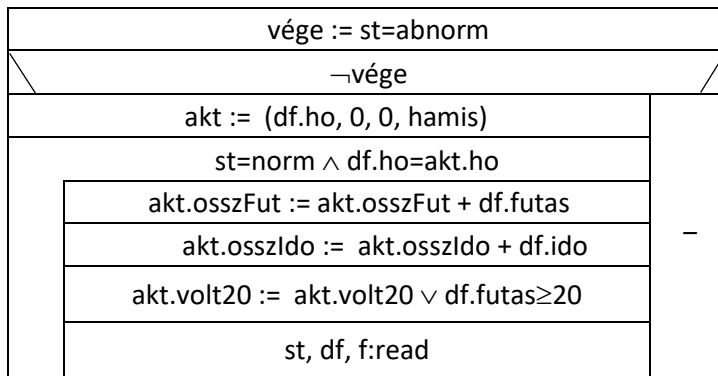
t:enor(E) ~ f:infile(Edzes) st,df,f:read,
first() nélkül,
amíg: df.ho=akt.ho

f(e) ~ (df.futas,
df.ido,
df.futas \geq 20)

s ~ akt.osszFutas, akt.osszldo, akt.volt20

H, +, 0 ~ ($\mathbb{N} \times \mathbb{N} \times \mathbb{L}$), (+,+, \vee), (0,0,hamis)

ALGORITMUS



Pontozás:

- 1. feladat: 30 pont**
specifikáció: 15 (4+3+8)
visszavezetés: 5
algoritmus: 10
- 2. feladat: 30 pont**
új állapottér, azon a specifikáció 7 (2+1+4)
visszavezetés: 2
algoritmus: 5
felsoroló:
állapottér: 2
first(): 1
current(): 1
end(): 1
next specifikáció (ef:1,uf:4): 5
visszavezetés: 2
algoritmus: 4

(A ZH pontozása egységes lesz, így ettől kicsit majd eltérő lehet.)