

5. Algoritmus minták felsorolása II.

1. Egy horgászversenyen a horgászok eredményét egy szekvenciális inputfájlban rögzítették. A fájl egy eleme egy horgász nevét és a halfogásainak sorozatát tartalmazza. Egy fogás egy időpontból, a kifogott hal fajtájának nevéből, a hal hosszából (m) és súlyából (kg) áll. Keressünk olyan horgászt, aki az 50 cm-esnél hosszabb pontyokból legalább 10 kilogramnyt fogott.

Specifikáció:

$A = (x:infile(Horgász), l:\mathbb{L}, név:\mathbb{S})$

$Horgász = rec(név:\mathbb{S}, zsákmány:Fogás^*)$

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

$Ef = (x=x_0)$

$Uf = (l, elem) = SEARCH_{horg \text{ in } x_0} \text{összsúly}(horg.zsákmány) \geq 10.0 \wedge l \rightarrow (név = elem.név)$

ahol $\text{összsúly}(horg.zsákmány) = \sum_{hal \text{ in } horg.zsákmány} hal.súly \text{ (összsúly : } Fogás^* \rightarrow \mathbb{R} \text{)}$
 $hal.fajta = \text{"ponty"} \wedge hal.hossz > 0.5$

Lineáris keresés

$felt(e) \sim \text{összsúly}(horg.zsákmány) \geq 10.0$

$t:enor(E) \sim x:infile(Horgász) \text{ (st,horg,x:read)}$

Algoritmus: st:Status, horg:Horgász

| | | | | | |
|---|--------------------|----------------------------------|--|--------------------------|--------------------|
| l := hamis | | | | | |
| st, horg, x : read | | | | | |
| $\neg l \wedge st = \text{norm}$ | | | | | |
| <table border="1"> <tr> <td colspan="2">összsúly(horg.fogás) ≥ 10.0</td></tr> <tr> <td>l, név := igaz, horg.név</td><td>st, horg, x : read</td></tr> </table> | | összsúly(horg.fogás) ≥ 10.0 | | l, név := igaz, horg.név | st, horg, x : read |
| összsúly(horg.fogás) ≥ 10.0 | | | | | |
| l, név := igaz, horg.név | st, horg, x : read | | | | |

Részfeladat: sum := pontysúly(x)

$A = (f:Fogás^*, sum:\mathbb{R}) \quad Fogás = rec(idő:\mathbb{S}, fajta:\mathbb{S}, hossz:\mathbb{R}, súly:\mathbb{R})$

$Ef = (f=f_0)$

$Uf = (f=f_0 \wedge sum = \sum_{hal \text{ in } f_0} hal.súly)$
 $hal.fajta = \text{"ponty"} \wedge hal.hossz \geq 0.5$

Összegzés (feltételes összegzés)

$H, +, 0 \sim \mathbb{R}, +, 0$

$f(e) \sim hal.súly \text{ ha } hal.fajta = \text{"ponty"} \wedge hal.hossz \geq 0.5$

$t:enor(E) \sim hal \text{ in } f$

Algoritmus:

| | | | | | |
|---|---|---|--|-----------------------|---|
| sum := 0.0 | | | | | |
| hal in f | | | | | |
| <table border="1"> <tr> <td colspan="2">hal.fajta = "ponty" \wedge hal.hossz ≥ 0.5</td></tr> <tr> <td>sum := sum + hal.súly</td><td>—</td></tr> </table> | | hal.fajta = "ponty" \wedge hal.hossz ≥ 0.5 | | sum := sum + hal.súly | — |
| hal.fajta = "ponty" \wedge hal.hossz ≥ 0.5 | | | | | |
| sum := sum + hal.súly | — | | | | |

hal:Fogás

Megjegyzés:

1. A fogások fenti összegzése lehetne a horgász (Horgász típusú objektum) egy metódusa is. Ekkor a horgász fogásainak (kezdetben üres) listájához egy másik metódussal lehetne hozzáadni új fogást a fájlból történő olvasás során.
2. A fogások összegzésének eredménye lehetne egy adattagja a horgász objektumnak, amelyet akkor módosítunk, valahányszor egy újabb fogást teszünk hozzá horgász fogásaihoz. Ekkor a fogások listájára nincs szükség.

2. Számoljuk ki egy számítástechnikai szaküzlet napi bevételét az aznapi forgalom alapján. A forgalmat a kiadott számlák mutatják, amelyeket egy szöveges állományban (szekvenciális inputfájl) rögzítettek. Az állomány minden sora egy-egy számla adatait tartalmazza: a vásárló nevét és az általa vásárolt termékek (cikkszám és ár párok) sorozatát.

Specifikáció:

$A = (f:\text{infile}(\text{Szám}la), \text{bevét}:\mathbb{N})$ Számla=rec(név:\$, lista:Áru*) Áru = rec(cikkszám:\$, ár:\$)

$Ef = (f=f_0)$

$Uf = (\text{bevét} = \sum_{sz \text{ in } f_0} \text{össz}(sz.\text{lista}))$ ahol $\text{össz}(sz.\text{lista}) = \sum_{e \text{ in } sz.\text{lista}} e.\text{ár}$

Összegzés

$H, +, 0 \sim \mathbb{N}, +, 0$

$f(e) \sim \text{össz}(sz.\text{lista})$

ahol $\text{össz} : \text{Áru}^* \rightarrow \mathbb{N}$

$s \sim \text{bevét}$

$t:\text{enor}(E) \sim f:\text{infile}(\text{Szám}la) (st,sz,f:\text{read})$

Algoritmus:

| | |
|---------------------------------|------------------------|
| bevét := 0 | st:Status sz:Számla |
| st, sz, f : read | |
| st = norm | |
| bevét := bevét + össz(sz.lista) | |
| st, sz, f : read | |

Részfeladat:

sum := össz(x)

$A = (x : \text{Áru}^*, \text{sum} : \mathbb{N})$

$Ef = (x=x_0)$

$Uf = (\text{sum} = \sum_{e \text{ in } x_0} e.\text{ár})$

Összegzés

$t:\text{enor}(E) \sim e \text{ in } x$

$f(e) \sim e.\text{ár}$

$H, +, 0 \sim \mathbb{N}, +, 0$

Algoritmus:

| | |
|-------------------|---------|
| sum := 0 | e : Áru |
| e in x | |
| sum := sum + e.ár | |

Megjegyzés:

1. A tételek fenti összegzése lehet a számla (Számla típusú objektum) egy metódusa. A számla tételeinek (kezdetben üres) listájához egy másik metódussal lehetne felvenni egy új tételt a fájlból történő olvasás során.
2. A tételek összegzésének eredménye lehet a számla (Számla típusú objektum) része (adattagja), amelyet akkor módosítunk, amikor a fájlból történő olvasás során a számla tételeinek (kezdetben üres) listájához egy újabb tételt adunk hozzá. Sőt a tételek listája sem kell: elég az összegzés eredményét adattagként felvenni.