Objektumelvű programozás
Algoritmusminták

Kiss-Bartha Nimród

2023. szeptember 5.

1. Programozási tétel vs. algoritmusminta

Tételekről progalapon beszélünk, OEP-en mintákról. Szerepüket tekintve hasonlók.

Egy átlagos feladat megoldásához az alábbi 3 elemet kell vázolni:

1. Specifikáció – állapottér (A), előfeltétel (Ef), utófeltétel (Uf)
2. Visszavezetési táblázat – valamiért nagyon fontosnak tűnik
3. Struktogram

2. Algoritmusminták

Megjegyzés: a nem matematikai halmazok ($\mathbb{R}, \mathbb{N}$, stb.) jelölésére is ugyanazt a betűtípust fogom használni, a saját típusokat meg félkövérrel jelölöm. A ZH-n nem kell így.

2.1. Összegzés

$f : \mathbb{E} \to \mathbb{H}$

$\mathbb{H}$ elemein értelmezett:

$+ : \mathbb{H} \times \mathbb{H} \to \mathbb{H}$

$0 \in \mathbb{H}$ (0 balneutrális elem eleme $\mathbb{H}$-nak)

$$A \;\; = (t : enor(\mathbb{E}),\; s : \mathbb{H})$$

$$Ef = (t = t')$$

$$Uf = \left(s = \sum_{e \in t'} f(e) \right)$$

$s := 0;\ t.First()$	
$\neg t.End()$	
	$s := s + f(t.Current())$
	$t.Next()$

Feltételes összegzés esetén így módosul a fenti függvény:

$$g : \mathbb{E} \to \mathbb{H}, \quad felt : \mathbb{E} \to \mathbb{L} \qquad \left(\sum_{\substack{e \in t' \\ felt(e)}} g(e) \right)$$

$$f(e) = \begin{cases} g(e) & \text{ha } felt(e) \\ 0 & \text{különben (feltéve, hogy 0 jobb neutrális elem is)} \end{cases}$$

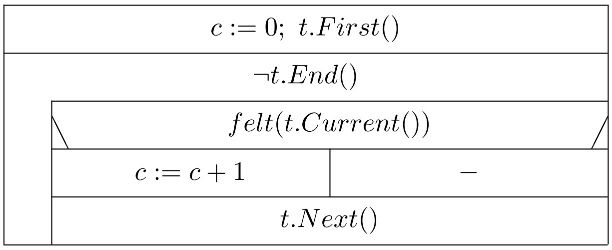
2.2. Megszámlálás

$felt : \mathbb{E} \to \mathbb{L}$

$$A \;= (t : enor(\mathbb{E}),\ c : \mathbb{H})$$

$$Ef = (t = t')$$

$$Uf = \left(c = \sum_{\substack{e \in t' \\ felt(e)}} 1 \right)$$



A számlálás egy speciális összegzés

$$\sum_{e \in t'} felt(e), \text{ azaz } f(e) = \begin{cases} 1 & \text{ha } felt(e) \\ 0 & \text{különben} \end{cases}$$

2.3. Maximumkiválasztás

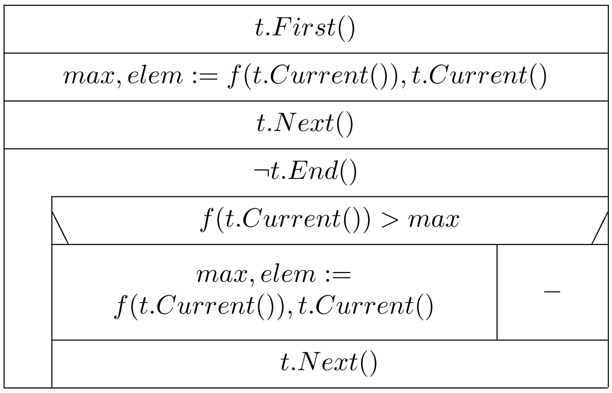
$felt : \mathbb{E} \to \mathbb{L}$

$\mathbb{H}$ elemei rendezhetőek

$$A \;= (t : enor(\mathbb{E}),\ elem : \mathbb{E},\ max : \mathbb{H})$$

$$Ef = (t = t' \ \wedge\ |t| > 0)$$

$$Uf = \left((max, elem) = \operatorname*{MAX}_{e \in t'} f(e) \right)$$



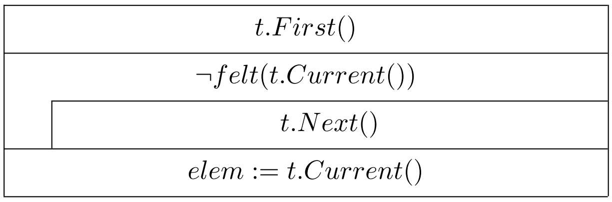
2.4. Kiválasztás (biztosan talál)

$felt : \mathbb{E} \to \mathbb{L}$

$$A \;= (t : enor(\mathbb{E}),\ elem : \mathbb{E})$$

$$Ef = (t = t' \ \wedge\ \exists e \in t : felt(e))$$

$$Uf = \left((elem, t) = \operatorname*{SELECT}_{e \in t'} felt(e) \right)$$



2.5. Lineáris keresés

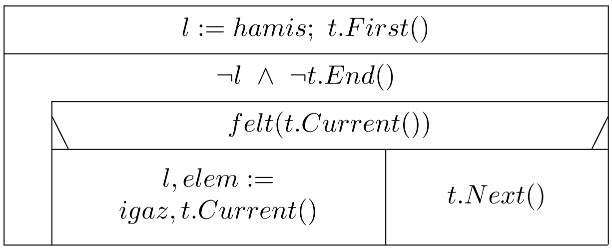
2.5.1. Pesszimista lineáris keresés ($\exists$)

$felt : \mathbb{E} \to \mathbb{L}$

$$A \;= (t : enor(\mathbb{E}),\ l : \mathbb{L})$$

$$Ef = (t = t')$$

$$Uf = \left((l, elem, t) = \operatorname*{SEARCH}_{e \in t'} felt(e) \right)$$



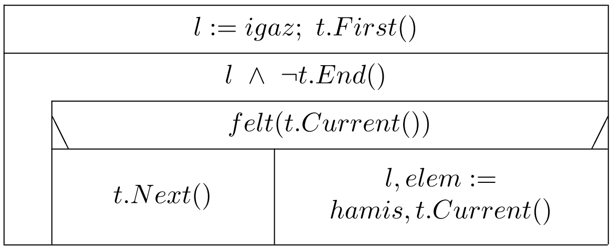
2.5.2. Optimista lineáris keresés (∀)

$felt : \mathbb{E} \to \mathbb{L}$

$A \;= (t : enor(\mathbb{E}),\ l : \mathbb{L},\ elem : \mathbb{E})$

$Ef = (t = t')$

$$Uf = \left((l, elem, t) = \forall \operatorname*{SEARCH}_{e \in t'} felt(e) \right)$$



2.6. Feltételes maximumkiválasztás

$f : \mathbb{E} \to \mathbb{H}$

$felt : \mathbb{E} \to \mathbb{L}$

$\mathbb{H}$ halmaz elemei rendezhetők

$A \;= (t : enor(\mathbb{E}),\ l : \mathbb{L},\ elem : \mathbb{E},\ max : \mathbb{H})$

$Ef = (t = t')$

$$Uf = \left((l, max, elem) = \operatorname*{MAX}_{\substack{e \in t' \\ felt(e)}} f(e) \right)$$