11.

Visszalépéses keresés

Legyen $n \in \mathbb{N}$, és $n > 1$. Legyenek U_i $(i \in [1..n])$ tetszőleges véges, nem üres halmazok ($0 < \sigma_i = |U_i| < \infty$). $U = U_1 \times \cdots \times U_n$.

Legyen $\varrho : U \to \mathbb{L}$, amely felbontható $\varrho_i : U \to \mathbb{L}$ $(i \in [0..n])$ tulajdonságok sorozatára az alábbi módon:

1. $\varrho_0 = \uparrow$;
2. $\forall i \in [0..n-1]) : \forall u \in U : \varrho_{i+1}(u) \to \varrho_i(u)$;
3. $\forall i \in [1..n] : \forall u, v \in U : (\forall j \in [1..i] : u_j = v_j) \to \varrho_i(u) = \varrho_i(v)$;
4. $\varrho = \varrho_n$.

A feladat annak eldöntése, hogy létezik-e olyan elem U-ban, amelyre teljesül a ϱ feltétel, és ha igen, adjunk meg egy ilyen elemet.

$$A = \underset{u}{U} \times \underset{l}{\mathbb{L}}$$
$$B = \{\mathcal{X}\}$$
$$Q : \uparrow$$
$$R : (l = \exists v \in U : \varrho(v) \wedge l \to (u \in U \wedge \varrho(u)))$$

Számozzuk meg U elemeit az alábbi módon: Minden U_i halmaz elemeit számozzuk meg nullától $\sigma_i - 1$-ig. Ezután U minden u eleméhez van egy $(i_1, \ldots, i_n)$ rendezett n-es, amelyre $u = (u_{i_1}, \ldots, u_{i_n})$, ahol $0 \le i_1 < \sigma_1, \ldots, 0 \le i_n < \sigma_n$. Ezen megszámozás egy lexikografikus rendezést definiál U-n.

Legyen $\mathcal{N} = [0..\sigma_1 - 1] \times \cdots \times [0..\sigma_n - 1]$. Ekkor a fenti megszámozás egy bijekciót létesít $\mathcal{N}$ és U között. Jelölje ezt az $\mathcal{N} \to U$ leképezést φ.

Vegyük észre, hogy az $\mathcal{N}$ halmaz elemei felfoghatók vegyesalapú számrendszerben felírt számként is. Ez alapján egy $\nu \in \mathcal{N}$ n-es számértéke:

$$\begin{aligned} f(\nu) &= \sum_{i=1}^{n} \nu_i * Q_i, \qquad \text{ahol:} \\ Q_i &= \prod_{j=i+1}^{n} \sigma_j \quad (i \in [1..n]) \end{aligned}$$

A bevezetett jelölésekkel a feladatot újra specifikáljuk, és most már megkövetelhetjük azt is, hogy ha létezik keresett tulajdonságú elem, akkor az első ilyet adjuk meg:

$$\begin{array}{l} A = \underset{\nu}{\mathcal{N}} \times \underset{l}{\mathbb{L}} \\ B = \{\mathcal{X}\} \\ Q : \uparrow \\ R : (l = \exists \mu \in \mathcal{N} : \varrho(\varphi(\mu))) \wedge \\ \qquad l \to (\varrho(\varphi(\nu)) \wedge \forall \mu \in \mathcal{N} : f(\mu) < f(\nu) \to \neg\varrho(\varphi(\mu))) \end{array}$$

Ha nem használjuk ki a ϱ speciális tulajdonságait, akkor a fenti feldat megoldható lineáris kereséssel, a $[0..|\mathcal{N}|-1]$ intervallumon. Vizsgáljuk meg, hogy hogyan használhatnánk ki ϱ specialitását!

Ha $\varrho_i(\varphi(\nu))$ igaz és $\varrho_{i+1}(\varphi(\nu))$ pedig hamis, akkor minden olyan $\nu' \in \mathcal{N}$-re, amelynek első $i+1$ komponense megegyezik ν első $i+1$ komponensével, $\varrho_{i+1}(\varphi(\nu'))$ is hamis lesz. Ezért ha az $i+1$-edik pozíció után a ν csak nullákat tartalmaz, akkor a keresésben nagyobbat léphetünk, és növelhetjük ν-t az $i+1$-edik pozíción.

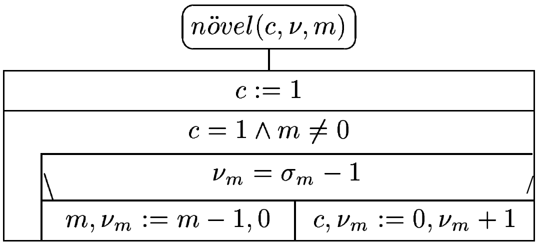
Az algoritmus még egyszerűbbé tehető, ha a ν-t kiegészítjük egy túlcsordulás bittel, amelynek 1 értéke azt jelzi, hogy ν értéke már nem növelhető.

Terjesszük ki az f függvényt az alábbi módon:

$f : \{0,1\} \times \mathcal{N} \to \mathbb{N}_0$,

$$\begin{aligned} f(c,\nu) &= c * Q_0 + \sum_{i=1}^{n} \nu_i * Q_i, \qquad \text{ahol:} \\ Q_0 &= \prod_{j=1}^{n} \sigma_j \end{aligned}$$

$$\begin{array}{l} A_{növel} = \underset{\nu}{\mathcal{N}} \times \underset{m}{\mathbb{N}_0} \times \underset{c}{\{0,1\}} \\ B_{növel} = \underset{\nu'}{\mathcal{N}} \times \underset{m'}{\mathbb{N}_0} \\ Q_{növel} : (\nu = \nu' \wedge m = m' \wedge m' \in [1..n] \wedge \forall i \in [m'+1..n] : \nu_i = 0) \\ R_{növel} : (f(c,\nu) = f(\nu') + Q_{m'} \wedge m \in [0..m'] \wedge \\ \qquad \forall i \in [m+1..n] : \nu_i = 0) \end{array}$$



$$P_{növel} : (f(\nu) + c * Q_m = f(\nu') + Q_{m'} \wedge m \in [0..m'] \wedge$$
$$\forall i \in [m+1..n] : \nu_i = 0 \wedge \forall i \in [1..m-1] : \nu_i = \nu_i')$$

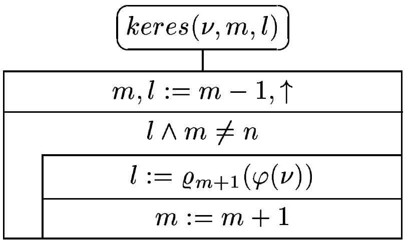
A fenti meggondolások másik következménye, hogy célszerű egy olyan művelet bevezetése is amely amellett, hogy $\varrho(\varphi(\nu))$-t eldönti, azt a legkisebb indexet is megadja, amelyre $\varrho_i(\varphi(\nu))$ hamis.

$$A_{keres} = \underset{\nu}{\mathcal{N}} \times \underset{m}{\mathbb{N}_0} \times \underset{l}{\mathbb{L}}$$

$$B_{keres} = \underset{\nu'}{\mathcal{N}} \times \underset{m}{\mathbb{N}_0}$$

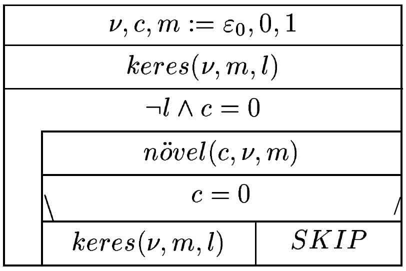
$$Q_{keres} : (\nu = \nu' \wedge m = m' \wedge m' \in [1..n] \wedge \varrho_{m'-1}(\varphi(\nu))$$
$$R_{keres} : (\nu = \nu' \wedge l = \varrho(\varphi(\nu)) \wedge$$
$$\neg l \rightarrow (m \in [m'..n] \wedge \varrho_{m-1}(\varphi(\nu)) \wedge \neg \varrho_m(\varphi(\nu)))$$

Ez a feladat visszavezethető lineáris keresés 2.8-ra.



A program levezetéséhez a már bemutatott specifikációt használjuk. Legyen az invariáns tulajdonság:

$$P : (\forall \mu \in \mathcal{N} : 0 \leq f(0, \mu) < f(c, \nu) \rightarrow \neg \varrho(\varphi(\mu)) \wedge l = \varrho(\varphi(\nu)) \wedge$$
$$\neg l \rightarrow (c = 1 \vee m \in [1..n] \wedge \neg \varrho_m(\varphi(\nu)) \wedge \varrho_{m-1}(\varphi(\nu)) \wedge$$
$$\forall i \in [m+1..n] : \nu_i = 0)$$



Megjegyezzük, hogy a visszalépéses kereséshez hasonlóan több visszalépéses technikát alkalmazó algoritmus is levezethető, például a visszalépéses számlálás: