

# Tartalomjegyzék

<b>Tétel 2: Program definíciója</b>	<b>1</b>
Utasítás, hatásreláció, értékadás . . . . .	1
Utasítás . . . . .	1
Hatásreláció . . . . .	1
SKIP, üres utasítás . . . . .	2
Általános értékadás . . . . .	2
Feltételes értékadás . . . . .	2
Állapotátmenetfa . . . . .	2
Végrehajtási út . . . . .	2
Elérhető állapotok halmaza . . . . .	2
Absztrakt program . . . . .	2
Feltétlenül pártatlan ütemezés . . . . .	3

## Tétel 2: Program definíciója

Célunk, hogy a korábban látott és tanult szekvenciális programok helyett lehetőségünk legyen párhuzamos programok definiálására. Összefésüléses szemantikát alkalmazunk, azaz minden párhuzamos program működése (hatása) helyettesíthető szekvenciával.

Nem definiálunk végrehajtási sorrendet, hanem az absztrakt program utasításait feltételes értékadások halmazaként adjuk meg, a sorrendet pedig a megfelelő feltételekkel tudjuk biztosítani. Amennyiben egy feltételes értékadás feltétele nem teljesül, akkor az utasítás ekvivalens a SKIP utasítással, egyébként pedig az értékadás végbemegy egy atomi utasításként. Minden értékadás az állapottér összes változójához értéket rendel, legfeljebb az előző értékkel azonos értéket. A feltételes értékadás a hiba állapotba való kerülést is megakadályozza: megfelelő feltétel kiválasztásával a hiba állapotok elkerülhetők (például az osztás feltétele a nemnulla osztó).

### Utasítás, hatásreláció, értékadás

Ezek valószínűleg nem annyira fontosak.

#### Utasítás

Definíció:  $s \subseteq A \times A^{**}$  reláció, ahol

- $D_s = A$  (mindenhol értelmezve van)
- $\forall a \in A : \forall \alpha \in s(a) : \alpha_1 = a$  (állapotátmenet-sorozat első eleme a kiindulási állapot)
- $\alpha \in A^\infty \wedge \exists i : \alpha_i = \alpha_{i+1} \implies \forall k > 0 : \alpha_i = \alpha_{i+k}$  (ha végtelen sorozatban két egymást követő állapot azonos, akkor onnantól kezdve minden állapot azonos)
- $\alpha \in A^* \implies \forall i \in [1..|\alpha|] : \alpha_i \neq \alpha_{i+1}$  (véges sorozatban szomszédos állapotok eltérőek)

#### Hatásreláció

Egy utasításhoz és egy kezdő állapothoz a véges végrehajtási sorozatok utolsó állapotait rendeli.

Definíció:  $p(s) \subseteq A \times A$  ahol

- $s \subseteq A \times A^{**}$  egy utasítás
- $D_{p(s)} = \{a \in A \mid s(a) \subseteq A^*\}$
- $p(s)(a) = \{b \in A \mid \exists \alpha \in s(a), \alpha \in A^* : b = \alpha_{|\alpha|}\}$

## SKIP, üres utasítás

Definíció:  $\forall a \in A : SKIP(a) = \{(a)\}$

## Általános értékadás

$a_1, \dots, a_n$  változókhoz az  $F_1, \dots, F_n$  függvények értékeit rendeli hozzá:  $a_1 \leftarrow F_1(a_1), \dots$ . Minden változóhoz rendel egy értéket, de a függvény lehet az identitás függvény is.

## Feltételes értékadás

Az általános értékadás kiterjesztése: ahol  $\pi_i$  teljesül, ott használjuk  $F_i$ -t. Ahol nem, ott használjuk az identitásfüggvényt.

Megtiltjuk a végtelenül nemdeterminisztikus műveleteket (pl. random szám generálás limit nélkül):  $\forall a \in A : |p(s)(a)| < \omega$  ahol  $\omega$  a természetes számok számossága.

A továbbiakban minden értékadás feltételes értékadás. Még akkor is, ha nem adunk meg egy explicit feltételt (ilyenkor a feltétel az azonosan igaz).

## Állapotátmenetfa

Egy gyökér kiindulási állapotból induló fa, amelynek egy újabb szintje egy újabb utasítás lefutását jelenti. Például a 2. szinten már 2 utasítás futott le.

Kikötés, hogy a gyökérből csak az  $s_0$ , a többi csúcsból  $s_i \in S$ , azaz  $s_1, \dots, s_n$  futhat le. Ha  $s_0$  determinisztikus, akkor a gyökérből csak egy él vezet ki, egyébként több él is lehetséges. A többi csúcsból pedig mindig legalább  $n$  él vezet ki (nemdeterminisztikusság miatt lehet több él, mint  $n$ ).

Feltétlenül pártatlan ütemezés mellett valójában helytelen az az út, ahol végtelenszer azonos utasítás kerül kiválasztásra. Ennek ellenére a fát úgy rajzoljuk fel, hogy minden (nem gyökér) csúcsból az összes utasításhoz felrajzoljuk az éleket, végtelen mélységig.

A fa helyesen címkézett, ha a csúcsok címkéi (állapotok) és az élek címkéi (utasítás indexek) összhangban vannak.

## Végrehajtási út

Összefésüléses szemantika miatt a program egy lehetséges lefutását reprezentálja.

Definíció:  $b \in UPG(S)(a)$  ekvivalenciaosztály reprezentációjának egy útja.

Végtelen sok út létezik, ezért nem tudjuk az összes utat egyesével ellenőrizni. Emiatt az utak összességét például invariáns állítások segítségével elemezzük.

## Elérhető állapotok halmaza

A végrehajtási utakon elhelyezkedő csúcsok címkéinek (azaz állapotoknak) halmaza.

## Absztrakt program

(Kezdő) állapothoz helyesen címkézett állapotátmenetfák ekvivalenciaosztályát rendeli.

Definíció:  $UPG(S) \subseteq A \times A^{***}$  ahol  $\forall a \in A : UPG(S)(a) = \text{az } (p(s_0), UP(S)) \text{ relációpár által generált helyesen címkézett állapotátmenetfák ekvivalenciaosztálya.}$

Jelölés:  $S = (s_0, \{s_1, \dots, s_n\})$  ahol  $s_i$  véges sok feltételes értékadás

Definíció értelmezése:

- $UP(S) = \bigcup_{s_i \in S}^{\text{diszjunkt unió}} p(s_i)$
- $A^{***}$ :  $A$  felett generált fák ekvivalenciaosztályainak halmaza
- Generált fa definíciója miatt csak feltételes értékadások lehetnek az utasítások és csak véges sok utasítás engedélyezett (azaz  $\{s_i \in S\}$  halmaznak végesnek kell lennie).

### **Feltétlenül pártatlan ütemezés**

Egy végrehajtási útra teljesül a feltétlenül pártatlan ütemezés, ha minden utasítás végtelen sokszor kerül végrehajtásra (minden él cimke végtelen sokszor fordul elő).