

PROGRAMOVANIE PARALÉLNYCH A DISTRIBUOVANÝCH SYSTÉMOV

DOMÁCA ÚLOHA 6

Autor: Marián Kravec

Úloha 1 - Global snapshot

Takéto pravidlo nezaručí správne zaznamenanie, pozrime sa na príklad:

Majme dva príkazy:

$a = a + b$

$c = a$

Uvažujme, že máme zaznamenané premenné $a.rec$ a $b.rec$, a $b > 0$, v takomto prípade by sa obe príkazy mohli vykonávať, keďže spĺňajú podmienku, že premenné na pravej strane, sú zaznamenané. Vidíme, že v každom bode programu musí platiť $c \leq a$, čiže musí platiť $c.rec \leq a.rec$ (lebo k a pripočítavame kladné b a c dokáže nadobudnúť iba hodnotu a v nejakom bode). Avšak ak tieto príkazy sa prvý príkaz niekoľkokrát vykoná, následne sa vykoná druhý príkaz a následne sa zaznamená hodnota premennej c , môže nastať situácia keď $c.rec > a.rec$ (keďže hodnota a rástla ale hodnota $a.rec$ už bola stabilná), čiže zaznamenáme nemožný stav programu.

Z toho vyplýva, že takáto podmienka nezaručí správne zaznamenanie.

Ak zmeníme podmienku na to, že ľavá strana musí byť celá zaznamenaná alebo nezaznamenaná tak stále nebude platiť, že máme zaručenú správnosť záznamu a ako príklad môžeme použiť presne ten istý príklad ktorý je uvedený vyššie, keďže ak sú zaznamenané iba $a.rec$ a $b.rec$ tak sa obe príkazy môže vykonať, keďže prvý má na ľavej strane iba zaznamenanú a druhý iba nezaznamenanú premennú.

Úloha 2 - Detekcia terminácie

Pravdupovediac som stratený, čo by sa dalo použiť ako vhodná metrika pre tieto programy. Nakoniec som si vybral, že metrikou bude hodnota premennej $claim$, kde $False = 0$ a $True = 1$.

Pre program R2:

safety: hodnota $claim$ nikdy neklesne, ak je 0 tak nemá kam klesnúť a ak je 1 tak sú všetky procesy idle takže neexistuje proces ktorý by niekoho mohol zobudiť a tým znížiť hodnotu $claim$

progress: existuje postupnosť príkazov ktoré zvýšia hodnotu $claim$ konkrétne ak sa vykonajú všetky príkazy z $\langle \Box u : u \in V :: u.idle := true \mid d := d \cup \{u\} \rangle$ tak sa dostanú všetky procesy do stavu idle a ak následne sa vykoná príkaz $claim := (d = V)$ tak sa zvýši hodnota $claim$ na 1

Pre program R3:

safety: totožná s programom R2

progress: veľmi podobná ako pri programe R2, len s tým rozdielom, že aby sme zaručili, že $b = V$ tak sa musia najprv vykonať všetky príkazy z časti $\langle \Box u : u \in V :: u.idle := true \rangle$, aby boli všetky procesy idle, následne $\langle \Box u : b, u.delta := b \cup \{u\} - u.delta, \emptyset \text{ if } u.idle \rangle$ aby sa vyprázdnili všetky množiny $u.delta$, potom ešte raz všetky príkazy $\langle \Box u : b, u.delta := b \cup \{u\} - u.delta, \emptyset \text{ if } u.idle \rangle$ aby sme zaručili, že sa všetky u dostali do množiny b a nakoniec $claim := (b = V)$ čo zvýši hodnotu $claim$.