

## Úlohy na precvičenie – PDS 2019 – séria H

Úlohy riešte samostatne a podrobne. Celý postup zaznamenajte a komentujte. Odpovede zdôvodňujte celými vetami (väčšinou je otázok v úlohe viac!). Cieľom cvičenia nie je vyhľadanie riešenia, ale získanie skúseností so samostatným riešením úloh. Vo výnimočných situáciách použité citácie uveďte v riešení. Je slušné tiež uviesť mená osôb, s ktorými ste riešenie konzultovali (čo samozrejme je povolené, pokiaľ riešenie spíšete potom samostatne). Uprednostnite ručný zápis – umožní vám jednoducho pridávať ilustračné obrázky a nákresy.

Za každé správne a úplné riešenie (samozrejme aj s postupom) možno získať v tejto sérii **dva body** (ak nie je uvedené inak). Zlomky bodov možno získať aj za čiastočné riešenia. Riešenia tejto série je nutné doručiť do **16. 5. 2019, 14:25** (do začiatku prednášky). Pred týmto termínom je možné odovzdať riešenia na sekretariáte Ústavu informatiky (do môjho priechodka). Neskôr dodané a opisované riešenia nebudú opravované ani hodnotené. Problémy môžete konzultovať po prednáške, elektronickou poštou alebo prostredníctvom diskusnej skupiny [pds2019@googlegroups.com](mailto:pds2019@googlegroups.com).

1. (1 bod) Počítačové hodiny nadbehnú za deň jednu minútu. Ako často treba zisťovať presný čas, aby som dosiahol synchronizáciu s inými počítačmi s presnosťou na 0,1 sekundy? Ako zrealizovať posun času, aby nedošlo k porušeniu časovej súslednosti časových pečiatok?

2. Distribuované algoritmy pre ošetrenie vzájomného vylúčenia v úplne prepojenej sieti sú citlivé na chyby, ktoré sa môžu v sieti stať. Skúste upraviť algoritmus Ricart-Agawala tak, aby čo najbezpečnejšie reagoval na výpadok najviac jedného z procesov. Zapište algoritmus po úpravách – príslušné parametre (timeout a pod.) zvolte ľubovoľne, ale v komentári obmedzenia presne špecifikujte. Opíšte aj situácie, ktoré vaša úprava nevyrieši. Ako sa prejaví na korektnosti riešenia nedodržanie predpokladu FIFO poradia prenosu správ? Za akých podmienok by bolo možné zabezpečiť aj odolnosť voči prerušeniu niektorého zo vzájomných prepojení procesov?

3. Procesy A, B a C sú prepojené do jednosmerného kruhu a každý z nich sa môže nachádzať v dvoch stavoch (0 a 1). Procesy si medzi sebou v tomto kruhu preposielajú dookola jeden token, pričom pri prijatí tokenu prepnú svoj vnútorný stav (z 0 na 1 a naopak). Predpokladajme, že všetky procesy sú v stave 0, proces A odošle token a inicializuje vytvorenie snímky systému (zaznamená svoj aktuálny stav a pošle marker procesu B). Znázornite časový diagram priebehu udalostí (podobne ako na slajde 16) a v ňom všetky možné konzistentné rezy, ktoré môžeme dosiahnuť Chandy-Lamportovym algoritmom. Výsledok zdôvodnite a okomentujte. Ukážte tiež, že v prípade silne súvislej siete  $n$  procesov po kompletizácii snímky systému Chandy-Lamportovym algoritmom bude aspoň  $n - 1$  kanálov prázdnych

\*. V súvislej distribuovanej sieti susedí každý proces s najviac  $d$  ďalšími procesmi. Chceme každému procesu priradiť farbu (číslo) z rozsahu  $\{0, 1, \dots, d\}$  tak, aby žiadne dva susediace procesy nemali rovnakú farbu (vždy to tak ide urobiť – ak nie, vezmem jeden proces z dvojice, ktorá má rovnakú farbu a prefarbím ho tak, aby mal farbu inú ako susedia – počet jednofarebných dvojíc sa zmenší). Navrhňte a zapište synchronný distribuovaný algoritmus, ktorý v čase  $O(d * \log n)$  takéto ofarbenie zrealizuje. Predpokladáme, že každý proces pozná hodnotu  $d$ , počet procesov v sieti  $n$ , svoj jednoznačný identifikátor  $MyId$  od 1 po  $n$  a zoznam susedov, s ktorými môže obojsmerne komunikovať. Algoritmus bude vykonávaný synchronne naraz vo všetkých uzloch siete. Ukážte, že sa výpočet ukončí správnym výsledkom a stanovte komunikačnú zložitosť. (hint: skúste najskôr riešiť úlohu, keď začínate s ofarbením v rozsahu  $\{0, 1, \dots, 2d, 2d+1\}$  a toto riešenie použijete pre prípad, že začíname s „ofarbením“  $\{1, \dots, n\}$ ).