

Slovenská technická univerzita

Fakulta informatiky a informačných technológií
Ilkovičova 3, 812 19 Bratislava

Paralelné programovanie

Synchronizačná úloha: Čašníci

Peter Berta

Cvičiaci: Ing. Matej Kloska
Študijný odbor: Informatika
Ročník: 3. Bc
Akademický rok: 2016/2017

1. Zadanie

Simulujte nasledujucu situáciu. Piaty casnici obsluhuju v restaurácii. Styria kuchari varia v kuchyni. Casnikovi trva nejaky cas, kym donesie objednavku od stola do kuchyne (v simulácii 1s). Potom caka na kuchara, ktoremu chvilu trva, kym jedlo uvari (v simulácii 2s). Nasledne casnik donesie jedlo z kuchyne zakaznikovi (v simulácii 1s). Cela simulacia nech trva nejaky cas (30s).

1. Doplnite do programu pocitadlo celkového počtu uvarených jedál a tiež nech si každý casnik pocita, kolko objednavok vybavil. Na konci simulacie vypiste hodnoty pocitadiel. [2b]
2. Ked casnik vybaví 2 objednavky, pocaka na dalsich dvoch a spravia si prestavku (v simulácii 2s). [5b]
3. Osetrite v programe spravne ukoncenie simulacie hned po uplynuti stanoveneho casu (nezacne sa dalsia cinnost). [3b]

Poznámky:

- na synchronizáciu použijete iba mutexy + podmienené premenné
- nespoliehajte sa na uvedené časy, simulácia by mala fungovať aj s inými časmi

2. Typy vlákien

Okrem hlavného vlákna (mainu) sú v tomto projekte použité dva typy vlákien. Čašníci sa starajú o prinesenie objednávky od stola do kuchyne, kde zobudia čakajúce vlákna kuchárov. To je druhý typ vlákien. Keď kuchár dovarí, tak zobudí čakajúcich čašníkov, ktorí donesú jedlo ku stolu.

3. Podrobnejší popis implementácie

Pri riešení tejto synchronizačnej úlohy som použil mutexy a podmienené premenné.

```
pthread_mutex_t uvar_mutex = PTHREAD_MUTEX_INITIALIZER;  
pthread_mutex_t cond_mutex = PTHREAD_MUTEX_INITIALIZER;
```

```
pthread_cond_t pauza_cond = PTHREAD_COND_INITIALIZER;  
pthread_cond_t varenie_cond = PTHREAD_COND_INITIALIZER;  
pthread_cond_t uvarene_cond = PTHREAD_COND_INITIALIZER;
```

Jedna podmienená premenná sa stará o čakanie kuchárov na prinesenie objednávky čašníkom. Ďalšia podmienená premenná sa stará o správne čakanie čašníka na uvarenie jedla. Posledná podmienená premenná sa stará o čakanie čašníkov na to, aby mohli ísť na pauzu, teda čakanie na ďalších čašníkov.

V programe používam globálne premenné pre počítanie uvarených jedál alebo prinesených objednavok:

```
int uvarene = 0;  
int cakaju = 0;  
int pauzuju = 0;  
int objednane = 0;  
int uvarene_akt = 0;
```

Prvá premenná je len jednoduché počítadlo celkového počtu uvarených jedál. Druhá premenná je použitá pri počítaní už čakajúcich čašníkov. Premenná s názvom „pauzuju“ je použitá pri správnom uspaní vlákien na pauze a zároveň zabezpečuje korektné počítanie čakajúcich. Premenná

„objednane“ je počítadlom objednávok prinesených do kuchyne a „uvarene_akt“ je počítadlom už uvarených jedál, ktoré čakajú na prinesenie ku stolu.

4. Ukončenie

Správne ukončenie simulácie je riešené pomocou premennej `stop`, ktorá je hlavným vláknom nastavená na hodnotu 1, keď má byť simulácia ukončená. Každé vlákno má v sebe podmienku pre ukončenie po každom `wait` alebo `sleep`. Ukončenie prebieha tak, že vlákno najprv odomkne zamknuté mutexy a potom zavolá všetky podmienené premenné, na ktoré by sa mohlo ešte čakať. Až potom sa vlákno ukončí.