

Kolokwium z Programowania Współbieżnego 2017-11-29

Każde zadanie prosimy oddać na osobnej kartce. Prosimy pisać jednostronnie.

Zadanie 1 (Semaforey, AJ, 10p.) Mieszkańcy pewnego miasta bardzo lubią grać w *Ucieknij z pokoju*. W mieście znajduje się P pokoi. Pokój i dla $0 \leq i < P$ ma ustaloną pojemność: $1 \leq pojemność_pokoju[i] \leq L$. Każdy z graczy należy do jednej z N grup. Każda grupa j dla $0 \leq j < N$ ma ustaloną licznosc: $1 \leq licznosc_grupy[j] \leq L$. Można założyć, że dla każdej grupy istnieje pokój o odpowiedniej pojemności oraz że $L \leq 10$.

Warunkiem rozegrania gry jest zebranie się całej grupy oraz dostępny pokój o pojemności równej licznosci tej grupy. Gracze rozpoczynają wtedy zabawę: wywołują `gra(numer_pokoju)`.

Gracze opuszczają pokoje pojedynczo. Pokój może być wykorzystany przez nową grupę graczy tylko wtedy, gdy jest pusty.

Zapisz treść procesu `Gracz(int numer_grupy)` wykorzystując do synchronizacji semaforey. Zadbaj o równomierne wykorzystanie pokoi o każdej pojemności oraz o żywotność grup graczy.

Zadanie 2 (Monitory, MM, 10p.) Firma przewozowa zatrudnia pewną liczbę pracowników oraz posiada K samochodów służących do przewozu towarów. Każdy pracownik ma jednoznaczny identyfikator i posiada prawo jazdy kategorii B lub C (wyższej).

Do realizacji każdego zlecenia jest potrzebny dokładnie jeden samochód oraz co najmniej dwóch pracowników, z których co najmniej jeden musi mieć prawo jazdy kategorii C. Dokładna liczba pracowników zależy od wielkości zlecenia.

Każdy pracownik zgłasza się do pracy i czeka na przydział zlecenia. Wśród pracowników przydzielonych do zlecenia kierownikiem zostaje ten z prawem jazdy kategorii C i najdłuższym stażem (liczba lat pracy). W przypadku gdy kilku pracowników ma jednakowy staż, kierownikiem zostaje dowolny pracownik o maksymalnym stażu. Po zakończeniu realizacji zlecenia kierownik zwalnia samochód.

Dyspozytor po otrzymaniu zlecenia przekazuje je do realizacji i czeka na informację kto został kierownikiem, którą przekazuje klientowi.

Schemat działania pracowników oraz dyspozytora jest następujący:

```
process PRACOWNIK (int id, int staż,
                    bool katC) {
    bool czyKierownik;
    Zlecenie zlecenie;
    while (true) {
        PRZEWOZY.Gotowy(id, staż, katC,
                        &czyKierownik, &zlecenie);
        realizuj(zlecenie);
        if (czyKierownik) {
            zwrotSamochodu;
            PRZEWOZY.Koniec();
        }
    }
}
```

```
process DYSPOZYTOR () {
    int iluPrac, idKier;
    Zlecenie zlecenie;
    while (true) {
        odbieramZlecenie(zlecenie, iluPrac);
        PRZEWOZY.NoweZlecenie(iluPrac,
                              zlecenie, &idKier);
        informacja(idKier);
    }
}
```

Napisz monitor `PRZEWOZY` zarządzający pracą firmy przewozowej w taki sposób, aby wszystkie podane warunki były spełnione i żaden proces nie został zablokowany. Monitor powinien udostępniać wszystkie wymienione powyżej procedury.