

# 1 Struktury danych

- $id$  - numer id procesu
- $n$  - liczba menadżerów
- $s$  - liczba miejsc w salonie
- $l$  - liczba lekarzy
- $m_i$  - liczba modelek procesu  $i$ -tego
- $shift_l$  - przesunięcie numeru lekarza względem początku  $kol_{lek}$
- $clock_{lek}$  - oznacza chwilę w której dany menadżer ubiega się o lekarza
- $kol_{lek}$  - kolejka procesów ubiegających się o dostęp do lekarza posortowana z użyciem pary  $(clock_{rec}, id_{proc})$
- $clock_{sal}$  - oznacza chwilę w które dany menadżer ubiega się o dostęp do salonu
- $kol_{sal}$  - kolejka procesów ubiegających się o dostęp do salonu posortowana z użyciem  $(clock_{rec}, id_{proc})$
- $l_{lek}$  - lista  $id_{proc}$  oznaczająca otrzymaną akceptację dostępu do lekarza od procesu
- $l_{sal}$  - lista  $id_{proc}$  oznaczająca otrzymaną akceptację dostępu do salonu od procesu
- $clock$  - wektor zegarów logicznych procesów
- $count_s$  - liczba dostępnych miejsc w salonie

# 2 Rodzaje komunikatów

Każdy komunikat zawiera  $rec_{id}$  - numer procesu oraz zawiera zmienną  $clock_{rec}$  będącą zegarem logicznym

- $ACK_{lek}$
- $REQ_{lek}$
- $ACK_{sal}$  - zawiera  $m_{rec}$  - liczba modelek zwolniona przez  $id_{proc}$ -ty proces
- $REQ_{sal}$  - zawiera  $m_{rec}$  - liczba modelek dla procesu  $j$  – tego

# 3 Algorytm

## 3.1 Odbiór wiadomości

$clock_i = \max(clock_i, clock_{rec}) + 1$ ,  $clock_{rec_{id}} = clock_{rec}$ , jeżeli wiadomość typu:

- $ACK_{lek}$  - Jeżeli para z  $kol_{lek}$  dla której  $id_{proc} = rec_{id}$  poprzedza parę  $(clock_{lek}, id)$  to  $shift_l + 1$ . Usuń tę parę z  $kol_{lek}$ , dodaj  $rec_{id}$  do  $l_{lek}$
- $REQ_{lek}$  - Dodaj parę  $(clock_{rec}, rec_{id})$  do  $kol_{lek}$ .
- $ACK_{sal}$  -  $count_s$  zwiększ o  $m_{rec}$
- $REQ_{sal}$  - Jeżeli nie istnieje taka para  $(clock_{sal}, id) \in kol_{sal}$ . Wyślij wiadomość typu  $ACK_{sal}$  do procesu  $rec_{id}$ , gdzie  $m_{rec} = 0$ . W przeciwnym wypadku jeżeli para z  $kol_{sal}$  dla której  $id_{proc} = rec_{id}$  poprzedza parę  $(clock_{sal}, id)$  to  $count_s$  zmniejsz o  $m_{rec}$ , w przeciwnym wypadku dodaj  $id_{proc}$  do  $l_{sal}$ .

### 3.2 Pseudokod

1.  $count_s = s$ ,  $clock_i = 0$ , losuj  $m_i$   $count_{acklek} = 0$ ,  $count_{acksal} = 0$
2. Odbiór wiadomości. Inkrementuj  $clock_i$ . Wyślij wiadomość typu  $REQ_{lek}$  z  $clock_{rec} = clock_i$ , dodaj parę  $(clock_i, id_{proc})$  do  $kol_{lek}$  oraz  $clock_{lek} = clock_i$ .
3. Obierz wiadomości. Jeżeli  $kol_{lek}.size() + l_{lek}.size() = n$  oraz  $l > nr$  oraz  $shift_l = l$  oraz gdzie  $nr$  jest numerem pozycji procesu w  $kol_{lek}$ . Skorzystaj z lekarza o numerze równym  $nr$  gdzie  $nr$  jest pozycją procesu w  $kol_{lek}$ ,  $shift_l := 0$ . W przeciwnym wypadku GOTO 3.
4. Wyślij wiadomość  $ACK_{lek}$  do każdego procesu.
5. Odbierz wiadomości. Inkrementuj  $clock_i$ . Wyślij wiadomość typu  $REQ_{sal}$  gdzie  $m_{rec} = m_i$ . Dodaj parę  $(clock_i, id)$  do  $kol_{sal}$ .
6. Odbierz wiadomości. Jeżeli dla każdego  $id_{proc} \notin kol_{sal}$ ,  $id_{proc} \notin l_{sal}$   $clock_{id_{proc}} > clock_{id}$  oraz  $count_s \geq m_i$  GOTO 7. W przeciwnym wypadku powtórz krok.
7. Zaprowadź modelki do salonu. Po powrocie modelek wyślij wiadomość typu  $ACK_{sal}$  z  $m_{rec} = m_i$  do każdego procesu.
8. Odbierz wiadomości. Jeżeli  $l_{sal}.size() = n$  Odbywa się konkurs i GOTO 1. W przeciwnym wypadku powtórz krok.