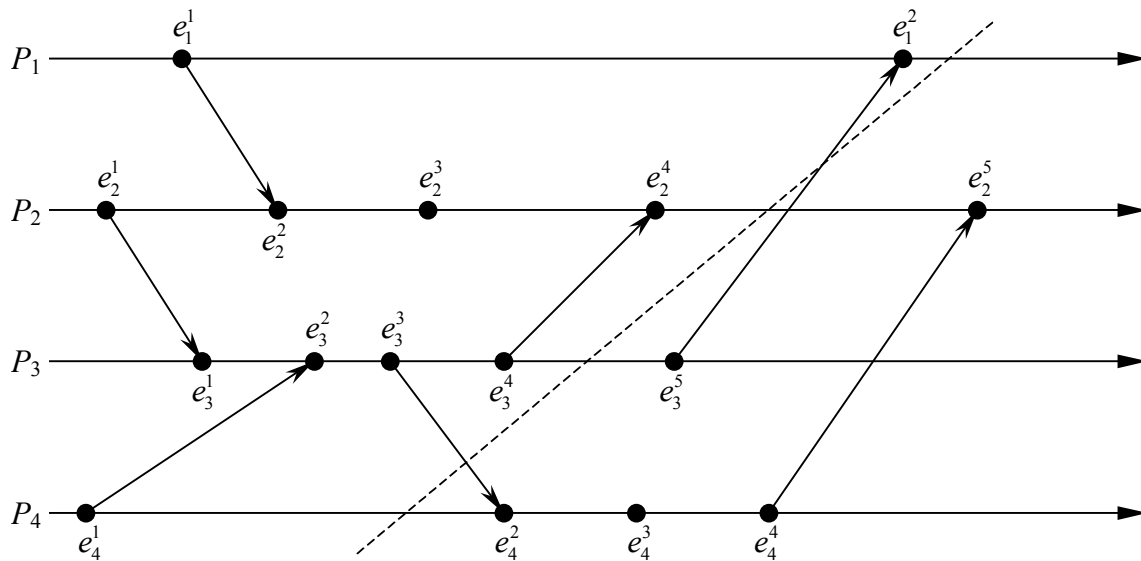
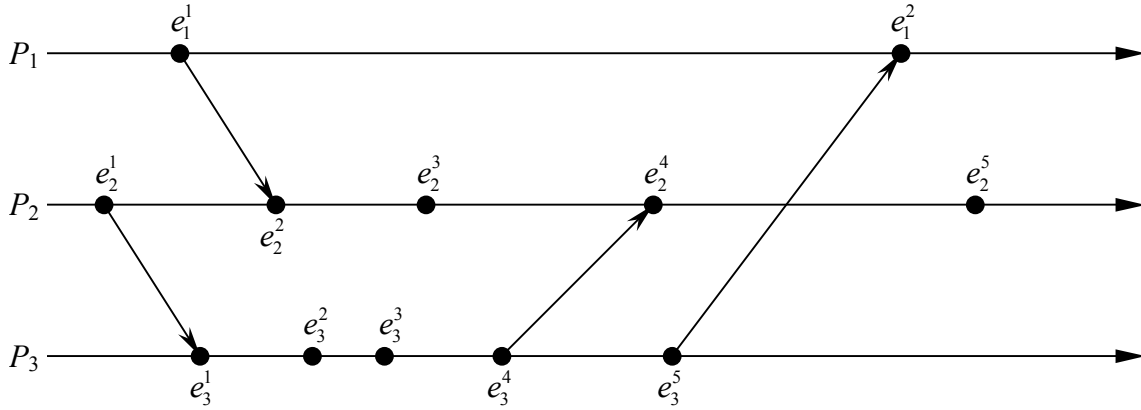


Treść zadań 1 – 8 odnosi się do poniższego diagramu przestrzenno-czasowego.

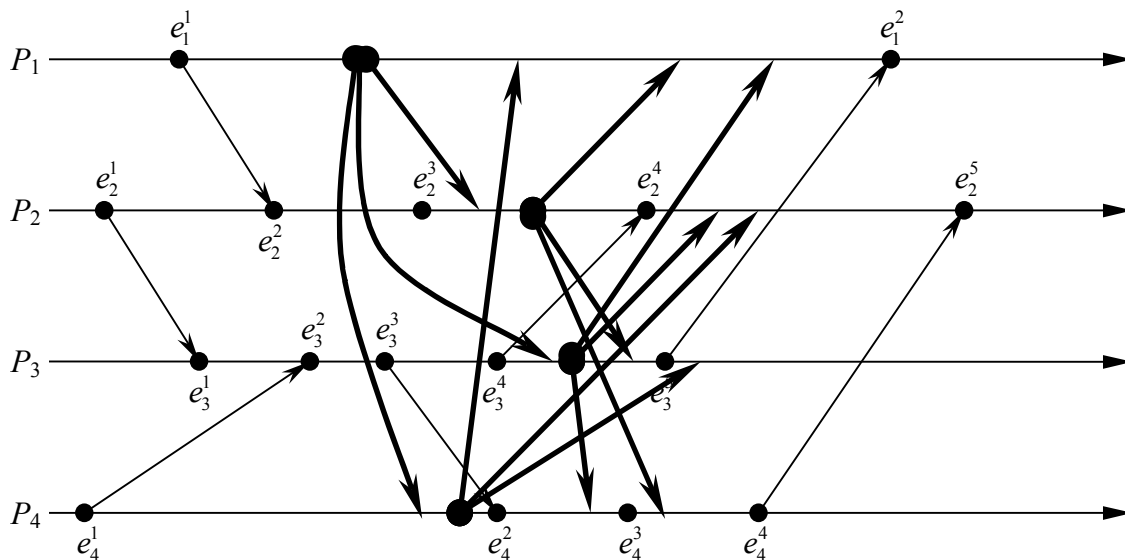


- Jaka będzie wartość zmiennej  $clock_2$  (zegara skalarnego) po zajściu zdarzenia  $e_2^5$  w procesie  $P_2$ , zakładając, że wartość początkowa (przed wystąpieniem pierwszego zdarzenia) w każdym procesie jest równa 0, a każde zdarzenie powoduje zwiększenie wartości zmiennej  $clock_i$  o 1?
- Jaka wartości znajdują się w tablicy  $vClock_2$  (zegar wektorowy) po zajściu zdarzenia  $e_2^5$  w procesie  $P_2$ , zakładając, że wartość początkowa (przed wystąpieniem pierwszego zdarzenia) w każdym procesie jest równa 0 na każdej pozycji tablicy  $vClock_i$ , a każde zdarzenie powoduje zwiększenie wartości na odpowiedniej pozycji tablicy  $vClock_i$  o 1?
- Czy w przetwarzaniu przedstawionym na powyższym diagramie przestrzenno-czasowym możliwe jest, żeby proces  $P_2$  był w stanie  $S_2^4$  i w tym samym czasie proces  $P_3$  był w stanie  $S_3^3$ . Uzasadnić odpowiedź.
- W programie każdego procesu zadeklarowana jest zmienna  $lv$  o wartości początkowej 0. Zdarzenie wewnętrzne  $e_2^3$  jest dekrementacją zmiennej  $lv$  ( $lv := lv - 1$ ), a zdarzenie wewnętrzne  $e_4^3$  jest inkrementacją zmiennej  $lv$  ( $lv := lv + 1$ ). Określić rodzaj predykatu globalnego (np. stabilny, definitely, possibly, niemożliwy do spełnienia)
  - $\mathcal{G}(\Sigma) \equiv (lv_1 + lv_2 + lv_3 + lv_4) > 0$
  - $\mathcal{G}(\Sigma) \equiv (lv_1 + lv_2 + lv_3 + lv_4) < 0$
  - $\mathcal{G}(\Sigma) \equiv (lv_1 + lv_2 + lv_3 + lv_4) = 0$
  - $\mathcal{G}(\Sigma) \equiv (lv_1 + lv_2 + lv_3 + lv_4) > 1$
- Czy obraz, w którym:
  - proces  $P_1$  jest w stanie  $S_1^2$ , proces  $P_2$  jest w stanie  $S_2^2$ , proces  $P_3$  jest w stanie  $S_3^2$  i proces  $P_4$  jest w stanie  $S_4^2$ ,
  - proces  $P_1$  jest w stanie  $S_1^1$ , proces  $P_2$  jest w stanie  $S_2^2$ , proces  $P_3$  jest w stanie  $S_3^3$  i proces  $P_4$  jest w stanie  $S_4^4$ ,
  - proces  $P_1$  jest w stanie  $S_1^1$ , proces  $P_2$  jest w stanie  $S_2^2$ , proces  $P_3$  jest w stanie  $S_3^3$  i proces  $P_4$  jest w stanie  $S_4^1$ ,
 jest spójnym obrazem stanu globalnego przedstawionego przetwarzania?
- Proszę podać trzy przykłady par zdarzeń współbieżnych oraz trzy przykłady par zdarzeń przyczynowo zależnych w przedstawionym przetwarzaniu.

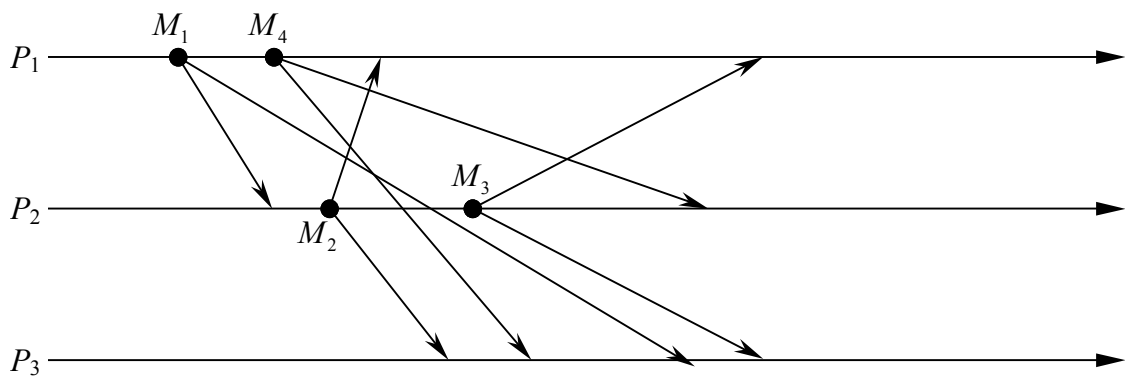
7. Proszę podać dwa przykłady konfiguracji spójnej stanów lokalnych i dwa przykłady konfiguracji, która nie jest spójna, w przedstawionym przetwarzaniu (uzasadnić odpowiedź).
8. Czy stan globalny odpowiadający konfiguracji, zaznaczonej przerywaną linią jest stanem osiągalnym przetwarzania przedstawionego na powyższym diagramie. Uzasadnić odpowiedź.
9. Narysować graf stanów osiągalnych (siatkę obliczeń rozproszonych) dla przedstawionego poniżej przetwarzania.



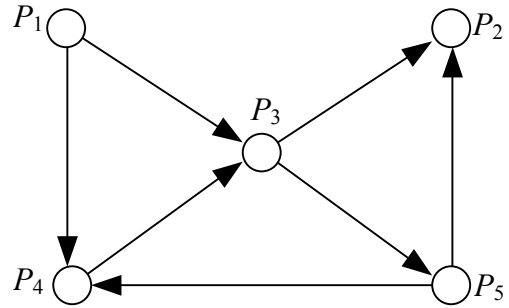
10. Jaki wniosek odnośnie kolejności wystąpienia zdarzeń  $e_i$ ,  $e_j$  i  $e_k$  można wyciągnąć z faktu, że:
- $T(e_i) = 3$ ,  $T(e_j) = 5$  i  $T(e_k) = 7$  oraz  $i = k$  i  $i \neq j$ ,
  - $T(e_i) = 4$ ,  $T(e_j) = 1$  i  $T(e_k) = 6$  oraz  $i \neq k, j \neq k$  i  $i \neq j$ ?
11. Jaki wniosek odnośnie kolejności wystąpienia zdarzeń  $e_1$ ,  $e_2$  i  $e_3$  w systemie rozproszonym, złożonym ze zbioru procesów sekwencyjnych  $\{P_1, P_2, P_3\}$ , można wyciągnąć z faktu, że:
- $T^\nu(e_1) = [3, 1, 2]$ ,  $T^\nu(e_2) = [5, 3, 4]$  i  $T^\nu(e_3) = [7, 2, 1]$ ,
  - $T^\nu(e_1) = [4, 1, 2]$ ,  $T^\nu(e_2) = [1, 3, 4]$  i  $T^\nu(e_3) = [5, 3, 3]$ ?
12. Jaki będzie obraz stanu globalnego, uzyskany z algorytmu Chandy-Lamport'a, którego realizację przedstawia poniższy diagram przestrzenno-czasowy (pogrubione linie oznaczają przesłanie znacznika)?



13. Który z komunikatów na poniższym diagramie przestrzenno-czasowym narusza
- porządek lokalny (porządek FIFO),
  - porządek przyczynowy? (uzasadnić odpowiedź)



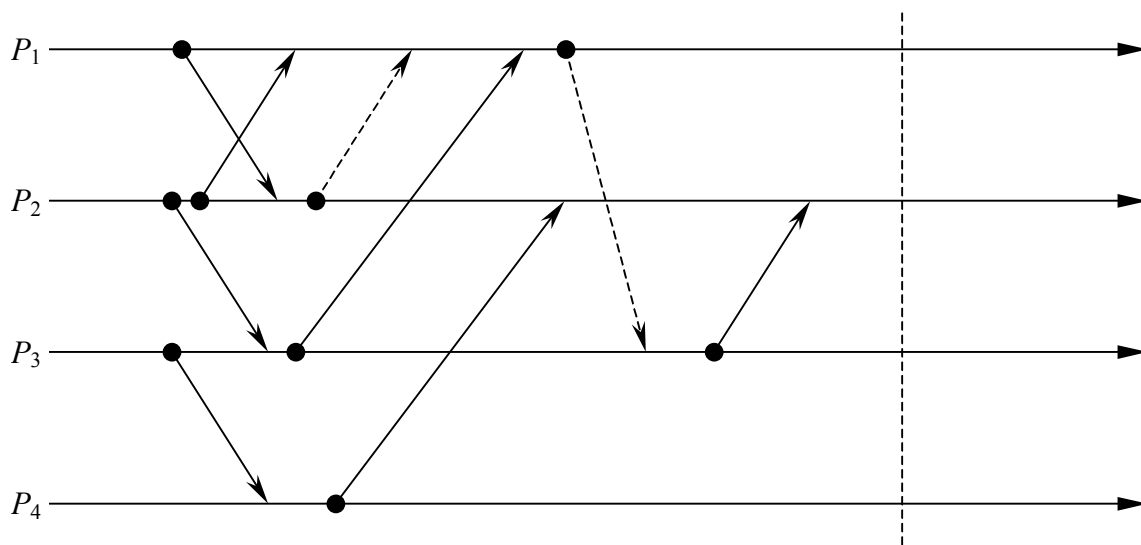
14. Na rys. obok przedstawiono graf oczekiwania dla procesów  $P_1, P_2, P_3, P_4, P_5$ .



(a) Czy wystąpiło zakleszczenie w przypadku modelu żądań OR? Jeśli tak, to które procesy znalazły się w stanie zakleszczenia?

(b) Czy wystąpiło zakleszczenie w przypadku modelu żądań AND? Jeśli tak, to które procesy znalazły się w stanie zakleszczenia?

15. Skonstruować graf oczekiwania dla procesów  $P_1, P_2, P_3, P_4$  po osiągnięciu stanu przetwarzania zaznaczonego przerywaną linią na poniższym diagramie przestrzenno-czasowym (linie ciągłe oznaczają komunikaty REQUEST, a przerywane GRANT).



16. Proces  $P_1$  wykonuje następujący ciąg operacji wysłania komunikatów za pomocą operacji typu *flush*:

$send^p(P_1, P_2, M1)$

$send^p(P_1, P_2, M2)$

$send^p(P_1, P_2, M3)$

$send^f(P_1, P_2, M4)$

$send^p(P_1, P_2, M5)$

$send^p(P_1, P_2, M6)$

Wskazać trzy różne porządki odbioru komunikatów (czyli kolejność odbioru komunikatów) przez proces  $P_2$ , które

(a) nie naruszają ograniczeń wynikających z własności kanałów typu *FC*,

(b) naruszają ograniczenia wynikające z własności kanałów typu *FC*.

17. Poniższy algorytm ma na celu zagwarantowanie kolejności FIFO doręczania komunikatów wysłanych przez proces  $P_1$  do procesu  $P_2$ . Wyjaśnić na czym polega błąd w poniższej implementacji i dokonać odpowiedniej korekty.

### TYPY KOMUNIKATÓW

**type** PACKET **extends** FRAME **is record of**

*seqNo* : INTEGER

*data* : MESSAGE

**end record**

### DEKLARACJE

*msgIn* : MESSAGE

*pcktOut* : PACKET

*delayBuf<sub>j</sub>* : **set of** PACKET :=  $\emptyset$

*seqNo<sub>j</sub>* : INTEGER := 0

*delivNo<sub>j</sub>* : INTEGER := 0

*delivered<sub>j</sub>* : BOOLEAN

### AKCJE

**when** *e\_send*( $P_1, P_2, msgOut$  : MESSAGE) **do**

*pcktOut.data* := *msgOut*

*pcktOut.seqNo* := *seqNo<sub>1</sub>*

**send**(  $Q_1, Q_2, pcktOut$  )

**end when**

**when** *e\_receive*( $Q_1, Q_2, pcktIn$  : PACKET) **do**

**if** *pcktIn.seqNo* = *delivNo<sub>2</sub>* + 1

**then**

*msgIn* := *pcktIn.data*

**deliver**(  $P_1, P_2, msgIn$  )

*delivNo<sub>2</sub>* := *delivNo<sub>2</sub>* + 1

*delivered<sub>2</sub>* := True

**else**

*delayBuf<sub>2</sub>* := *delayBuf<sub>2</sub>*  $\cup$  { *pcktIn* }

*delivered<sub>2</sub>* := False

**end if**

**while** *delivered<sub>2</sub>* **do**

*delivered<sub>2</sub>* := False

**for all** *pckt*  $\in$  *delayBuf<sub>2</sub>* **do**

**if** *pckt.seqNo* = *delivNo<sub>2</sub>* + 1

**then**

*msgIn* := *pckt.data*

**deliver**(  $P_1, P_2, msgIn$  )

*delivNo<sub>2</sub>* := *delivNo<sub>2</sub>* + 1

*delivered<sub>2</sub>* := True

*delayBuf<sub>2</sub>* := *delayBuf<sub>2</sub>*  $\setminus$  { *pckt* }

**end if**

**end for**

**end while**

**end when**