

## Demonstracja działania algorytmu Maekawa w sekcji krytycznej

### 1. Algorytm Maekawa

#### a. Wymagania:

- Proces chcący wejść do sekcji krytycznej, potrzebuje pozwolenia od wszystkich ze swojego zbioru
- Proces może naraz wysłać tylko jedną odpowiedź
- FIFO
- Bezawaryjność węzłów i łączy
- Przecięcia zbiorów są niepuste
- Każdy węzeł należy do własnego zbioru
- Zbiory mają taki sam rozmiar
- Dowolny węzeł należy do takiej samej liczby różnych zbiorów arbitrażu

#### b. Rodzaje zapytań

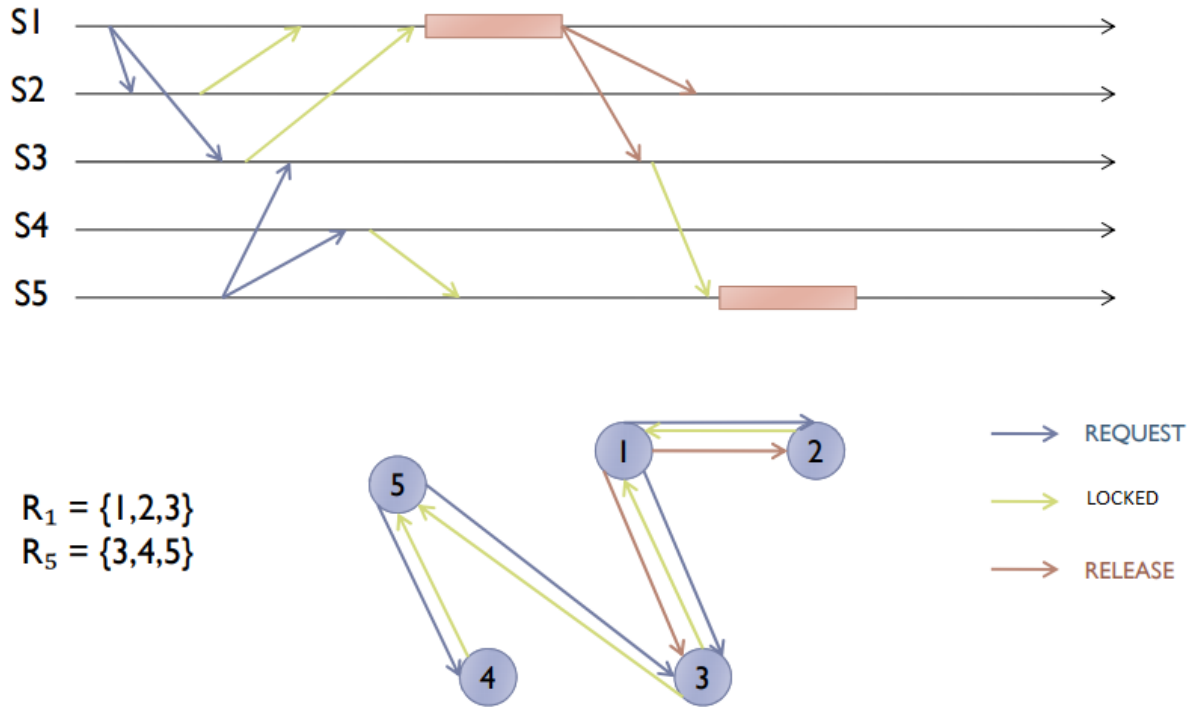
- REQUEST - proszę o zgodę na CS
- LOCKED - zgoda na CS
- RELEASE - wyszedłem z CS
- FAILED - poczekaj, jest ktoś ważniejszy
- INQUIRE - zgłosił się do mnie ktoś ważniejszy
- RELINQUISH - zrzekam się twojej zgody na CS

#### c. Opis działania

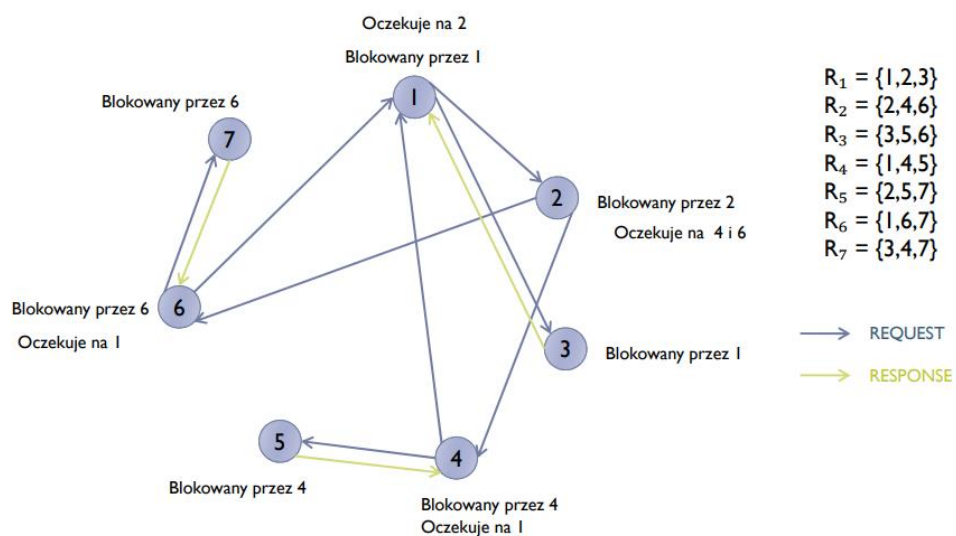
- Węzeł chcący wejść do sekcji krytycznej wysyła wiadomość REQUEST do każdego węzła w swojej grupie arbitrażu. Dostęp do CS jest możliwy po otrzymaniu kompletu odpowiedzi LOCKED.
- Węzeł otrzymujący REQUEST odpowiada LOCKED jeśli nie jest zablokowany. Odpowiadając, zmienia swój stan na zablokowany.
- Jeśli zablokowany węzeł otrzyma REQUEST, umieszcza go w swojej kolejce priorytetowej. Następnie:
  - Jeśli REQUEST ma wyższy priorytet od blokującej wiadomości oraz od wszystkich w kolejce, wysyła INQUIRE do blokującego procesu.
  - W przeciwnym wypadku wysyła FAILED do procesu który nadał REQUEST.
- Aby wejść do CS, węzeł czeka na komplet LOCKED. Po wyjściu z CS, wysyła RELEASE.
- Węzeł otrzymujący RELEASE wysyła LOCKED do węzła którego REQUEST ma najwyższy priorytet według jego kolejki. Jeśli kolejka jest pusta, węzeł staje się niezablokowany.
- Węzeł otrzymujący INQUIRE odpowiada RELINQUISH (zrzeka się przyznanego mu LOCKED) jeśli otrzymał już chociaż jeden komunikat FAILED. Jeśli nie, węzeł zapamiętuje INQUIRE na wypadek gdyby FAILED przybył później.

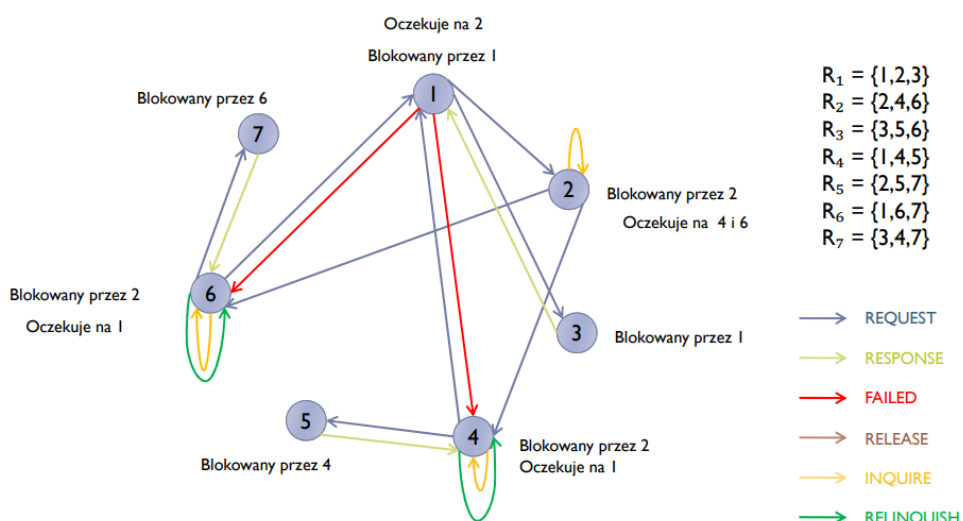
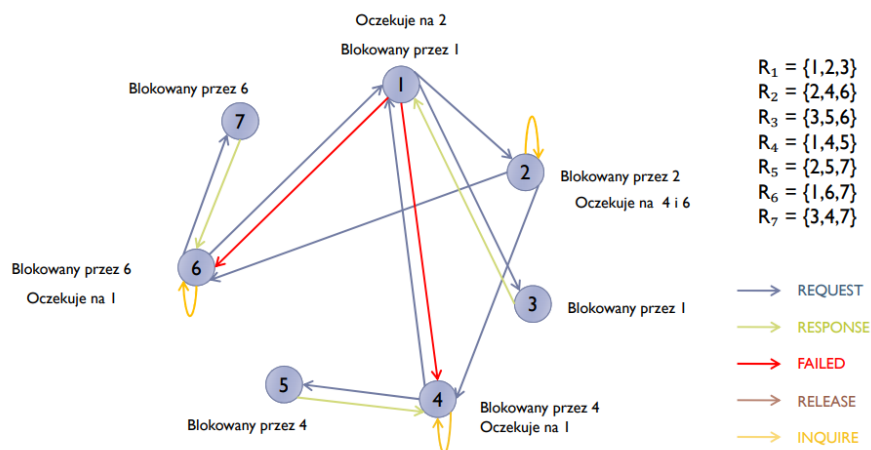
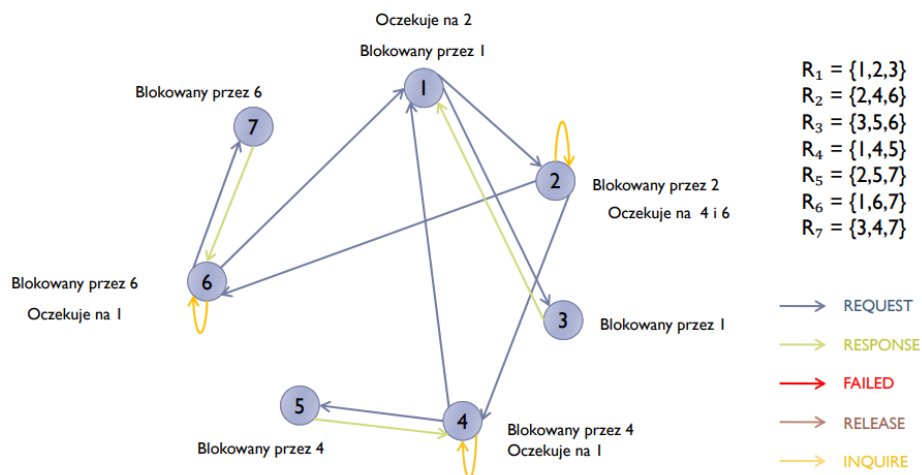
## 2. Maszyna stanów

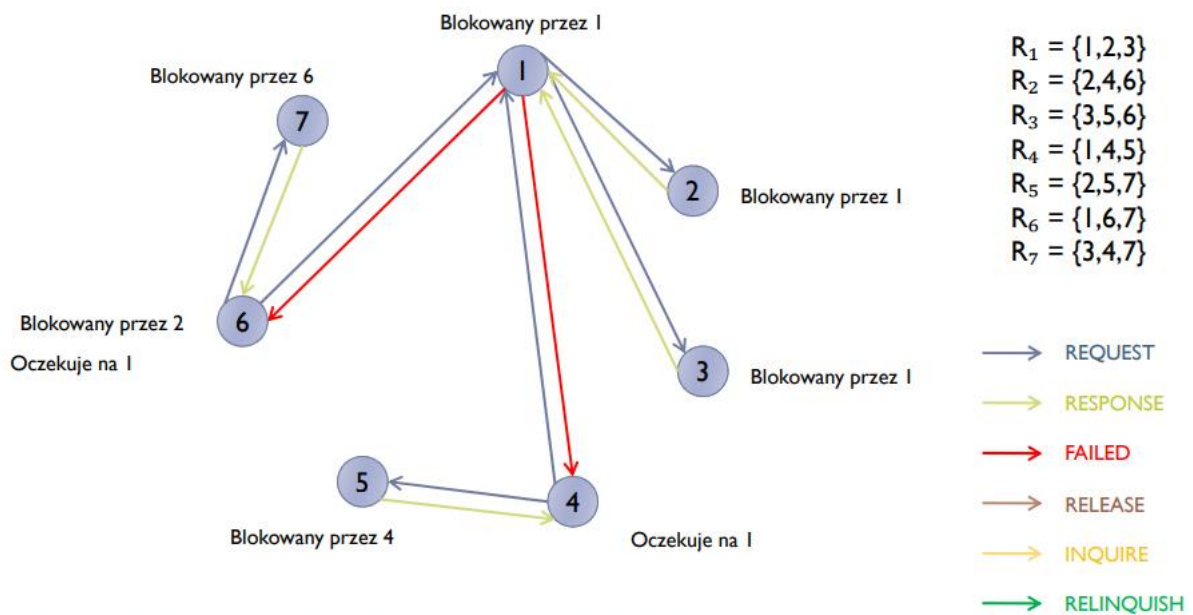
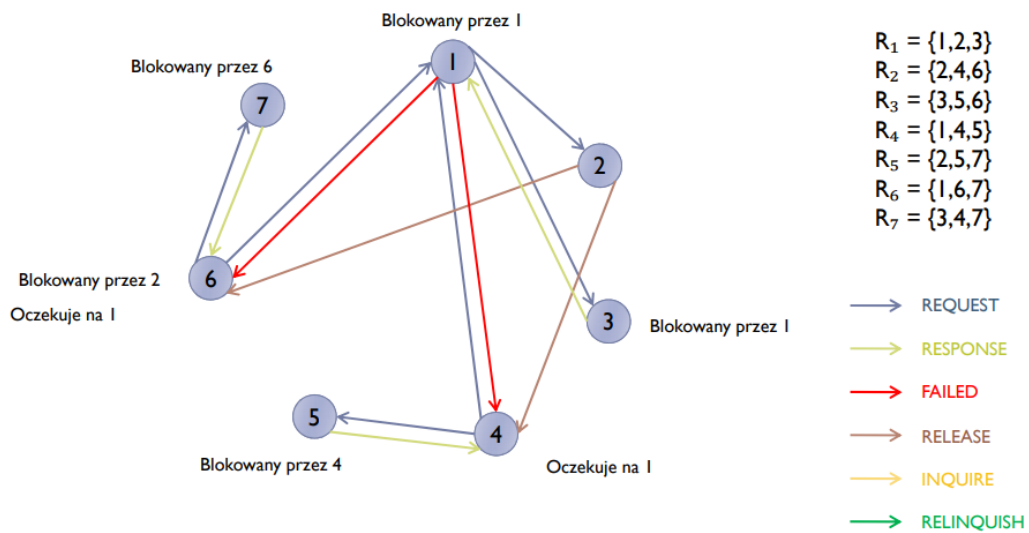
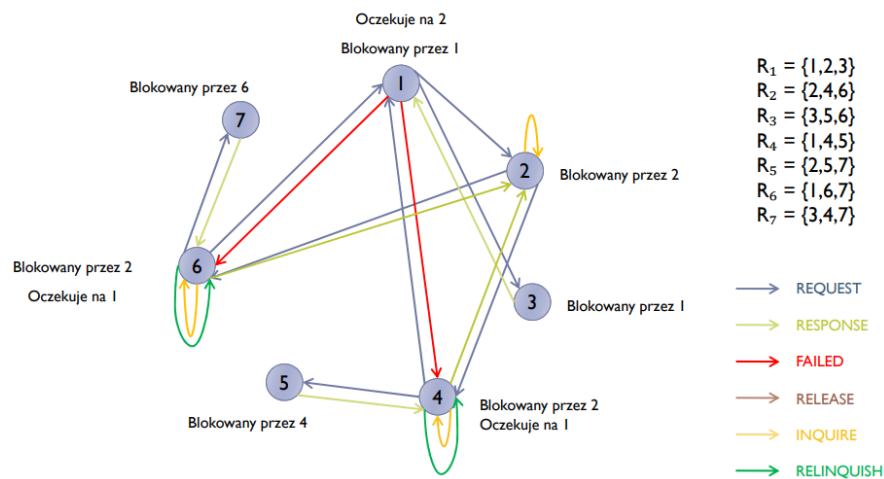
### a. Przypadek bez zakleszczeń



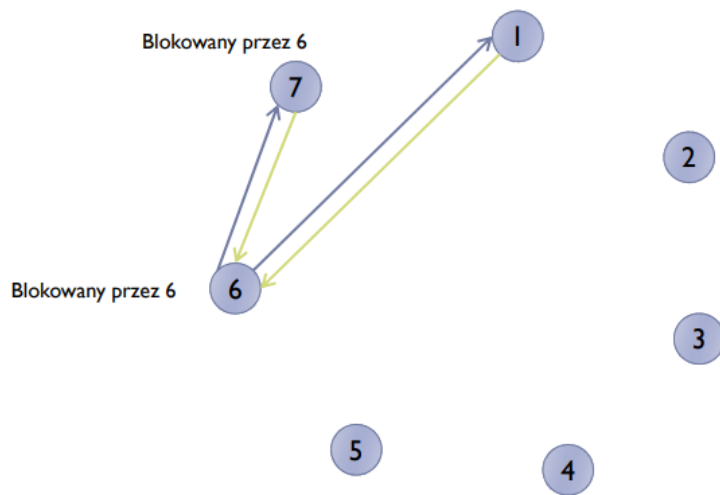
### b. Przypadek z zakleszczeniami





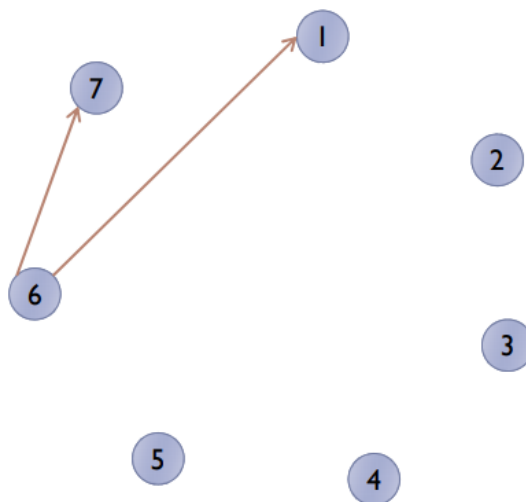






$R_1 = \{1,2,3\}$   
 $R_2 = \{2,4,6\}$   
 $R_3 = \{3,5,6\}$   
 $R_4 = \{1,4,5\}$   
 $R_5 = \{2,5,7\}$   
 $R_6 = \{1,6,7\}$   
 $R_7 = \{3,4,7\}$

→ REQUEST  
 → RESPONSE  
 → FAILED  
 → RELEASE  
 → INQUIRE  
 → RELINQUISH



$R_1 = \{1,2,3\}$   
 $R_2 = \{2,4,6\}$   
 $R_3 = \{3,5,6\}$   
 $R_4 = \{1,4,5\}$   
 $R_5 = \{2,5,7\}$   
 $R_6 = \{1,6,7\}$   
 $R_7 = \{3,4,7\}$

→ REQUEST  
 → RESPONSE  
 → FAILED  
 → RELEASE  
 → INQUIRE  
 → RELINQUISH