

# Systemy operacyjne (I)

## Ćwiczenie III – semafor

### 1. Treść zadania

Jest danych 2 producentów, którzy produkują losowe liczby i umieszczają je w 9-cio elementowym buforze FIFO. Jest też danych 2 konsumentów, którzy konsumują produkty z bufora (usuwiają je), przy czym konsument pierwszy konsumuje tylko liczby parzyste, a drugi tylko liczby nieparzyste. Ponadto konsumenci mogą konsumować tylko, jeśli w buforze znajdują się co najmniej 3 produkty. Zadanie należy wykonać korzystając z semaforów.

### 2. Proponowane rozwiązanie

Po pierwsze, należy pamiętać o zabezpieczeniu się przed próbą umieszczenia produktu w pełnym buforze przez producenta. Bufor nie jest nieskończony i posiada narzuconą liczbę wolnych miejsc – 9, zgodnie z założeniami zadania. W przypadku konsumentów, należy sprawdzić ilość elementów znajdujących się w buforze (trzeba zapewnić zgodność z założeniami zadania). Jeśli będą w nim mniej niż trzy elementy, pobranie elementu nie może nastąpić, a proces zostanie umieszczony w kolejce (sytuacja analogiczna do klasycznego problemu producent – konsument, w którym pobranie elementu nie mogło nastąpić przy pustym buforze).

Bufor zostanie zaimplementowany jako kolejka FIFO.

W związku z tym, iż klient i producent operują na różnych elementach bufora dopuszczalne jest, aby jednocześnie wykonywać na nim operacje wstawiania i pobierania.

W rozważanym wariantcie mogą pojawić się dwa rodzaje wyścigów. Pierwszym z nich jest sytuacja, gdy dwóch producentów chce skorzystać z tego samego zasobu w tym samym momencie. Druga natomiast, to specyficzny przypadek jednoczesnej próby pobrania dwóch produktów przez dwóch konsumentów. W zasadzie nic nie stoi na przeszkodzie, by konsumenci współdziałali ze sobą – każdego interesuje inny rodzaj produktów, nie istnieje więc zagrożenie próby pobrania tego samego elementu z bufora w jednej chwili. Niestety, ale jak wspomniano wyżej, takie rozwiązanie nie sprawdza się w jednym przypadku: gdy mamy 3 elementy w buforze, a na pobranie czekają liczba parzysta i nieparzysta. Współdziałanie klientów doprowadzi więc do sytuacji, gdy jeden z konsumentów pobiera produkt, chociaż w buforze znajdują się mniej niż trzy elementy (założenia zadania nie są spełnione).

W celu realizacja zadania mam więc zamiar wykorzystać cztery semafor: pusty – który będzie podniesiony, jeśli w buforze będą wolne elementy, pełny – podniesiony, jeśli w buforze będą zajęte elementy i dwa semafor binarne typu *mutex* dbające o to, aby tylko jeden producent/konsument mógł skorzystać z danego zasobu.

Aktualne wartości dwóch pierwszych wskazują odpowiednio ilość wolnych i zajętych miejsc w buforze.

Producenci produkują i wstawiają produkty do bufora tak długo, aż nie dostaną informacji o całkowitym wypełnieniu miejsc. Wtedy procesy, które pozostają bez przydziału (znajdują się w kolejce), czekają do momentu, aż zwolni się miejsce. Analogiczna sytuacja występuje w przypadku konsumentów.

Dodatkowo: zostanie zaimplementowana struktura semafor, której elementami będą kolejka i licznik - zmienna typu integer, w funkcji producenta losowe liczby będą generowane za pomocą funkcji rand(), a jako jej zarodek będzie wstawiany czas pobrany w sekundach, zostaną zaimplementowane dwie różne funkcje konsumenta; każda z nich będzie "zainteresowana" innym typem produktu (tu: parzystością liczby).