Synchronizacja procesów z wykorzystaniem semaforów

Tomasz Załuska

**Problem**

Napisać program rozwiązujący problem producenci-konsumenci. W programie występuje jeden konsument oraz wielu producentów. W systemie występuje bufor, do którego producenci wkładają produkty, a konsument odbiera je z niego. Użytkownik wybiera liczbę producentów, wielkość bufora oraz liczbę produktów jaką musi wyprodukować każdy producent.

**Założenia wstępne**

- nie dopuścić do czytania z pustego bufora,

-nie dopuścić do zapisu do pełnego bufora,

-zadbać o "nie przeszkadzanie sobie" procesów zapisujących do tego samego bufora i procesów czytających z tego samego bufora,

**Zmienne globalne**

-int \*buffer( wskaźnik na tablicę, która będzie służyć jako bufor),

-int buff\_size(wielkość bufora jaką ustawił użytkownik),

-int num\_producer(liczba producentów jaką ustawił użytkownik),

-int num\_items(liczba produktów jaką ustawił użytkownik),

-int count, in, out(zmienne służące do obsługi bufora, licznik produktów w buforze, wskaźnik ja pierwszy i ostatni element bufora),

-semafory mutex, full, empty(służą do zabezpieczenia bufora, aby tylko jeden wątek mógł z niego korzystać),

**Funkcje**

1. Int insert\_item(int item)

Funkcja służąca do wkładania produktów do bufora. Zapewnia ona dostęp do bufora tylko jednemu wątkowi przy użyciu semaforów. Zwraca 0 w przypadku powodzenia oraz -1 w przypadku błędu.

1. Int remove\_item(int \*item)

Funkcja służąca do wyjmowania produktów z bufora. Zapewnia ona dostęp do bufora tylko jednemu wątkowi przy użyciu semaforów. Zwraca 0 w przypadku powodzenia oraz -1 w przypadku wystąpienia błędu.

1. Void \*producer(void)

Funkcja służąca do utworzenia wątku producenta. Wykonuje ona funkcje dodania do bufora określoną liczbę razy.

1. Void \*consumer(void)

Funkcja służąca do utworzenia wątku konsumenta. Wykonuje ona funkcje wyjęcia z bufora tyle razy co liczba producentów pomnożona przez liczbę produktów.

1. Int setId(void)

Funkcja służąca do nadawania numerów id procesom producenta.