

Systemy operacyjne - Laboratorium 4

Koncepcja rozwiązania

Treść zadania

1. Zaimplementować mechanizm priorytetowych kolejek wiadomości dla komunikacji międzyprocesowej.
2. Wykorzystując zaimplementowany mechanizm kolejkowy przygotować części składowe systemu przetwarzającego wiadomości w kolejkach.
3. Przeprowadzić prezentację systemu, odpowiedzieć na pytania.

Kolejki priorytetowe

Rozwiązanie zawierać będzie 3 bufora cykliczne przechowujące wiadomości oraz priorytety. W zależności od priorytetu wiadomość będzie umieszczana w odpowiednim miejscu.

Kolejki zostaną zaimplementowane jako statyczne bufora cykliczne – tablica o ustalonej wielkości. Zaletą bufora cyklicznego jest brak konieczności przesuwania elementów gdy chcemy usunąć pierwszy element lub dodać jakiś element na początek np. z wyższym priorytetem.

Wszystkie funkcje obsługujące bufor zostaną dodane w postaci biblioteki. Lekka modyfikacja biblioteki z zadania 3.

System przetwarzający wiadomości w kolejkach

Każda z kolejek będzie obsługiwana przez monitor zawierający

3 semaforów – wraz z odpowiednimi metodami obsługi:

- 2 zliczające
 - *empty* - Liczba pustych miejsc – inicjalizowane pojemnością kolejki
 - *full* - Liczba zajętych miejsc – inicjalizowane wartością 0
- 2 mutex
 - *mutex* - Realizujący wzajemne wykluczanie procesów chcących otrzymać dostęp do kolejki
 - *top_priority* – Informujący proces K czy jest w kolejce komunikat o najwyższym priorytecie

Kolejkę priorytetową z obsługą zaimplementowaną w oddzielnej bibliotece (pkt. 2) wraz ze zmienną oznaczającą stan zapelnienia.

Producent i producent specjalny realizuje założenia zadania 3 jednakże przy użyciu monitorów, które wykonują odpowiednie operacje semaforowe.

Konsument realizuje założenia zadania 3. Używając monitora kolejki uzyskuje dostęp do sekcji krytycznej. Następnie uciną pierwszy symbol z komunikatu i zgodnie z danym prawdopodobieństwem *pr* dopisuje lub nie nową literę na jego koniec. Mając gotowy nowy komunikat sprawdza czy ma dostęp do kolejki określonej uciętym symbolem poprzez jej monitor po czym zapisuje ten element w odpowiednim miejscu w zależności od priorytetu i opuszcza sekcję krytyczną – obsługa za pomocą opisanego monitora.

Oczekiwanie między działaniami realizowane będzie poprzez funkcję *sleep()* oraz *usleep()*, ponieważ *sleep* obsługuje jedynie całkowite wartości sekund.

Producent ochronny sprawdzając stan zapełnienia kolejek wybierał będzie tą najbardziej zapełnioną i wrzucał do niej komunikat o najwyższym priorytecie. Wzajemne wykluczanie realizowane przez monitor.

Istnieją dwie możliwości realizacji:

- Producent ochronny zawsze będzie wrzucał ten komunikat na początek kolejki (miejsce pierwsze do odczytu) po sprawdzeniu semafora full oraz mutex a następnie podnosił semafor *top_priority* wybudzając tym samym proces K.
- Producent ochronny z racji produkcji pustych komunikatów nie wstawia ich do kolejki lecz jedynie podnosi semafor *top_priority* i wybudza proces K, który obsługuje 5 komunikatów.

Różnica polega na zachowaniu się producenta ochronnego w przypadku gdy kolejka jest pełna. W pierwszym przypadku zostanie on wstrzymany a w drugim może zwolnić miejsce poprzez obsługę 5 komunikatów przez proces K – nie jest to pewne, ponieważ istnieje niewielkie prawdopodobieństwo, że wszystkie komunikaty znów wylądują w tej samej kolejce.

Proces K oczekiwał będzie na komunikat o najwyższym priorytecie a po wybudzeniu przez podniesienie semafora *top_priority* w monitorze dowolnej kolejki obsłuży on 5 komunikatów zgodnie z procedurą zwyczajnego konsumenta.

Każdy z typów procesów opisany zostanie przez funkcje wykonujące opisane w poleceniu zadania i korzystające z zaimplementowanych monitorów.

Prezentacja systemu

System zostanie zaprezentowany poprzez stworzenie trzech procesów producentów, trzech konsumentów, jednego producenta specjalnego, jednego producenta ochronnego i procesu K. W trakcie działania programu wypisywane będą informacje wraz z dokładną godziną:

- Co wykonuje dany proces – informowanie o najważniejszych momentach tj.
 - oczekiwanie na wejście do sekcji krytycznej
 - pobranie lub dopisanie komunikatu w danej kolejce
- Aktualny stan kolejki – wypisanie jej zawartości

Na koniec wyświetlona zostanie informacja o poprawności zakończenia programu oraz wyczyszczeniu wszelkich użytych zasobów.