Konrad Winnicki,

Warszawa, 9 grudnia 2018

Systemy Operacyjne

Konspekt projektu drugiego

**Temat:**

Komunikacja międzyprocesowa z wykorzystaniem mechanizmu semaforów oraz pamięci współdzielonej.

**Zadania do zrealizowania:**

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.

**Propozycja rozwiązania zadania 1:**

* Struktury kolejek i komunikatów przechowywane w pamięci współdzielonej.
* Implementacja list dynamicznych komunikatów z różniących się od standardowej implementacji tym, że wskazanie na kolejny element nie jest absolutnym wskaźnikiem na komunikat, a­le określa pozycję komunikatu względem początku tablicy komunikatów.
* Taka metoda wynika z faktu wykorzystania przez procesy pamięci współdzielonej, która nie ma gwarancji dowiązania pod ten sam zakres przestrzeni adresowej poszczególnych procesów.
* Wszystkie komunikaty zawarte są w tablicy komunikatów o określonej pojemności.
* Stworzenie tablicy struktur typu Message zawierającej:
  + trzyelementową tablicę znaków z zakresu (‘A’, ‘B’, ‘C’), lub NULL
  + priorytet komunikatu
  + wskazanie na następny komunikat zawierające:
    - adres następnego komunikatu względem początku tablicy komunikatów lub
    - w przypadku braku następnego komunikatu wartość spoza zakresu tablicy komunikatów, np. wartość wskazującą na element o jeden dalej niż ostatni komunikat w tablicy.
  + timestamp chwili powstania komunikatu
* Stworzenie biblioteki do obsługi kolejek priorytetowych z funkcjonalnościami:
  + Nieblokującymi:
    - sprawdzenie czy kolejka pełna,
    - czy kolejka pusta,
    - stan zajętości kolejki.
      * stan zajętości określany na podstawie ilości aktywnych komunikatów w tablicy komunikatów.
  + Blokującymi:
    - Pobranie komunikatu z kolejki:
      * pobranie pierwszego komunikatu z listy.
    - Wpisanie komunikatu do kolejki:
      * komunikaty zwykłe wpisywane na koniec kolejki,
      * komunikaty priorytetowe wpisywane na początek kolejki zaraz za istniejącymi komunikatami priorytetowymi.
  + Proponowana funkcja blokująca przyjmująca komendę READ/WRITE umożliwiająca w danej chwili dostęp do kolejki tylko jednemu procesowi,
  + Blokowanie procesów realizowane przy wykorzystaniu semaforów.

**Propozycja rozwiązania zadania 2:**

* Utworzenie trzech kolejek komunikatów o identyfikatorach „A”, „B” i „C” o odpowiednich pojemnościach w przestrzeni pamięci współdzielonej,
* Każdy z producentów, konsumentów i producentów specjalnych zaimplementowany jako osobny program wykorzystujący bibliotekę kolejek priorytetowych,
* Producent jako argument wywołania przyjmuje identyfikator kolejki:

jeśli kolejka nie istnieje, tworzy nową kolejkę, a w przeciwnym wypadku dołącza się do niej i rozpoczyna cykliczne publikowanie w danej kolejce,

Konsument jako argument wywołania przyjmuje identyfikator kolejki:

Konsument bedący właściwie prosumentem sprawdza czy istnieją już kolejki o identyfikatorach „A”, „B” i „C”:

jeśli nie istnieją takie kolejki, tworzy je.

Producent specjalny różni się od producenta zwykłego nadawanym priorytetem oraz losowością wybierania kolejek do których trafią losowo wygenerowane komunikaty.

**Propozycja rozwiązania zadania 3:**

* Testy systemu realizowane przy pomocy dodatkowej biblioteki Historian zbierającej informacje w czasie testu o stanie kolejek oraz zawartych w nich komunikatach
* Zbierane informacje to:
  + stan zajętości kolejek w danej chwili,
  + dodawany do średniej czas obsługi komunikatów o priorytetach zwykłych i wysokich.