

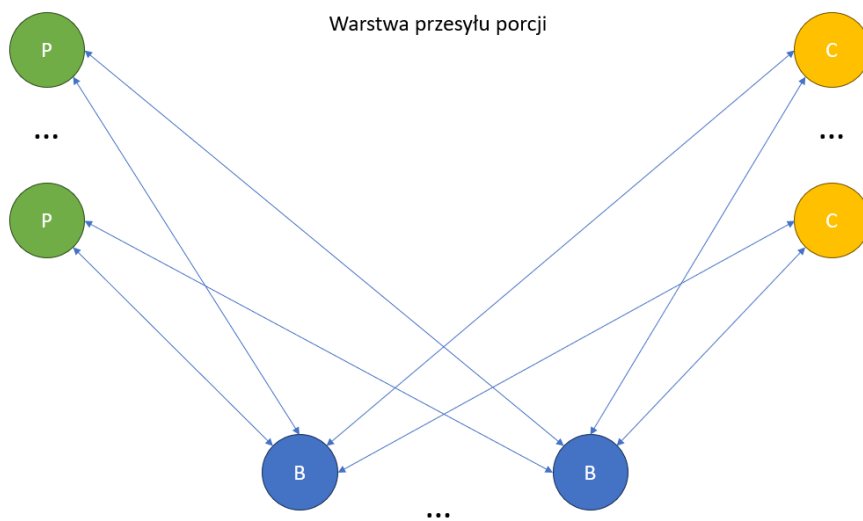
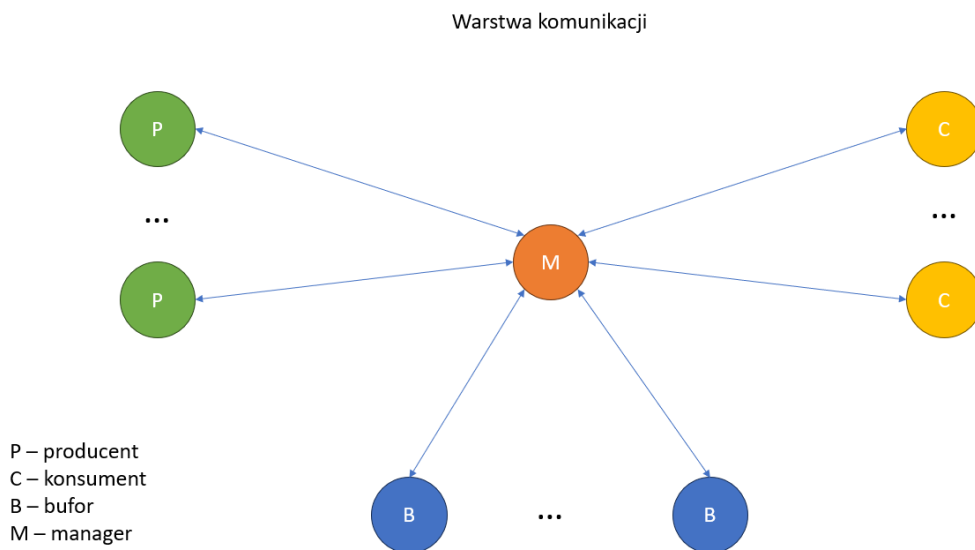
Teoria Współbieżności

Sprawozdanie 1 - Zadanie 3 - Bufor rozproszony w JCSP

Autorka:

Paulina Jędrychowska (grupa 10, śr. 11:20)

1. Schemat architektury połączeń:



2. Opis pomiarów:

W eksperymentach był mierzony czas wykonywania programu oraz czasy, w których producenci, konsumenci i bufor oczekiwali na odpowiedź od managera. Czas mierzony był w milisekundach. Dodatkowo zmierzono ilość konsumpcji poszczególnych konsumentów w celu sprawdzenia równoważenia obciążenia.

Wszystkie pomiary były wykonywane dziesięciokrotnie, a następnie uśrednione.

3. Eksperymenty:

We wszystkich testach była ta sama ilość producentów = 2, konsumentów = 2 i buforów = 2.

Pomiar 1:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 10
- Liczba produkcji pojedynczego klienta: 1000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 0 ms

Pomiar 2:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 10
- Liczba produkcji pojedynczego klienta: 10000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 0 ms

Pomiar 3:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 10
- Liczba produkcji pojedynczego klienta: 1000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 1 ms

Pomiar 4:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 10
- Liczba produkcji pojedynczego klienta: 10000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 1 ms

Pomiary 5-6 to te same pomiary co 1-4, tylko ze zmniejszonym buforem.

Pomiar 5:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 2
- Liczba produkcji pojedynczego klienta: 1000

- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 0 ms

Pomiar 6:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 2
- Liczba produkcji pojedynczego klienta: 10000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 0 ms

Pomiar 7:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 2
- Liczba produkcji pojedynczego klienta: 1000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 1 ms

Pomiar 8:

Ustawienia eksperymentu:

- Rozmiar pojedynczego bufora: 2
- Liczba produkcji pojedynczego klienta: 10000
- Sleep dla pojedynczego procesu (producenta, klienta i bufora): 1 ms

Na zielono w tabelach zostały oznaczone pomiary dla zmniejszonego bufora w celu większej czytelności:

Nr. pomiaru	Czas działania [ms]	Czas czekania producent 1 [ms]	Czas czekania producent 2 [ms]	Czas czekania konsument 1 [ms]	Czas czekania konsument 2 [ms]	Czas czekania bufor 1 [ms]	Czas czekania bufor 2 [ms]
1.	192	159	159	154	155	163	163
2.	1713	1441	1442	1398	1401	1483	1478
3.	4599	2298	2388	2310	2399	325	404
4.	45633	23009	23733	22828	23742	3195	4024
5.	203	169	169	163	164	173	172
6.	1772	1491	1491	1445	1445	1534	1530
7.	4688	2343	2430	2356	2422	293	358
8.	46598	23369	24254	23396	24116	3131	3821

Tabela 1. Porównanie czasów

Wnioski:

Dużą część czasu procesy spędzają na czekaniu na sygnał od managera. Przy dodaniu sleepa dla procesów ich czas oczekiwania dużo się zmniejsza. Dla producentów i klientów czas oczekiwania stanowi około 50% czasu programu, a dla buforów około 1/10 czasu, bo dostają one często sygnał od managera, że mogą się z kimś połączyć. Sleep symuluje zachowanie w rzeczywistości procesu, który zazwyczaj wykonuje więcej zadań niż requestowanie zasobu, więc zmniejszony czas oczekiwania dla procesów ze sleepem jest pożądany. Dodatkowo, można zauważyć, że dla zmniejszonego bufora czas praktycznie nie uległ zmianie, co ma sens ponieważ manager dokładnie zna stany wszystkich buforów i dobrze radzi sobie z balansowaniem obciążenia.

Nr. pomiaru	Ilość konsumpcji przez konsumenta 1	Ilość konsumpcji przez konsumenta 2
1.	1004	995
2.	10004	9993
3.	1028	969
4.	10195	9805
5.	1002	999
6.	10003	9998
7.	1002	999
8.	10257	9745

Tabela 2. Porównanie konsumpcji dla konsumentów 1 i 2

Wnioski:

Jak widać w tabeli, konsumpcje rozkładają się niemal po równo między konsumentami.

4. Wady i zalety rozwiązania:

1. Zalety:

- Manager minimalizuje liczbę zapytań, które nie skutkują późniejszym połączeniem (akcją produkcji albo konsumpcji). Dzięki знaniu dokładnych stanów buforów i producentów manager zawsze da odpowiedź zwrotną, z którym kanałem się połączyć.
- Procesy nie znają swoich stanów na wzajem, nie wiedzą co jest po drugiej stronie kanału. Wszystko jest zarządzane poprzez managera.

- Manager z łatwością radzi sobie z balansowaniem obciążenia. Dobrze sprawdzi się dla małych, często przepelniających się buforów.

2. Wady:

- Manager jest obciążony zapytaniami - wszystkie zapytania idą do managera, który przekierowuje je dalej, dlatego przy dużej ilości zapytań od procesów większość czekania odbywa się w pierwszej fazie komunikacji z managerem. Dla sytuacji, w której procesy często przesyłają zapytania ta implementacja może nie być najlepszym pomysłem.
- Złożoność pamięciowa - manager trzyma w sobie stany wszystkich buforów i dwie kolejki dla producentów i konsumentów, więc dla dużej ilości procesów zajęta będzie większa ilość pamięci. (choć i tak nie są to duże koszty)

3. Możliwości poprawy:

- Dodanie skalowania rozwiązania, na razie kod jest ustawiony dla 2 producentów, 2 konsumentów, 2 buforów.
- Zwiększenie ilości managerów w celu równoważenia obciążenia managera.