

Teoria Współbieżności

Ćwiczenie 4

1 Cel ćwiczenia

Celem ćwiczenia jest porównanie różnych rozwiązań problemu pięciu filozofów w dwóch paradygmacie programowania wielowątkowego (Java). Ćwiczenie zakłada znajomość podstaw programowania współbieżnego w powyższym języku i środowisku.

2 Wprowadzenie do ćwiczenia

Problem pięciu filozofów jest jednym z klasycznych problemów teorii współbieżności. Podstawowe sformułowanie problemu jest następujące :

- N filozofów zasiada przy okrągłym stole
- Pomiedzy sąsiednimi filozofami leży pałeczka (łącznie jest N pałeczek)
- Każdy filozof działa ciągle według schematu „myślenie - jedzenie - myślenie - jedzenie - ...”. Każdy z etapów (myślenie i jedzenie) jest skończony.
- Aby zjeść, filozof musi podnieść oba sąsiadujące pałeczki

Zadanie: zaprojektuj algorytm jednoczesnej alokacji współdzielonych zasobów (pałeczek) przez konkurujące procesy (filozofowie), tak aby uniknąć zakleszczenia i zagłodzenia.

Niektóre z rozwiązań problemu pięciu filozofów są następujące [1].

1. **Rozwiązanie naiwne (z możliwością blokady).** Każdy filozof czeka, aż wolny będzie lewa pałeczka, a następnie ją podnosi (zajmuje), następnie podobnie postępuje z prawą pałeczką.

2. **Rozwiązanie z możliwością zagłodzenia.** Każdy filozof sprawdza czy oba sąsiednie pałeczki są wolne i dopiero wtedy zajmuje je jednocześnie. Rozwiązanie to jest wolne od blokady, jednak w przypadku, gdy zawsze któryś z sąsiadów będzie zajęty jedzeniem, nastąpi zagłodzenie, gdyż obie pałeczki nigdy nie będą wolne.
3. **Rozwiązanie asymetryczne.** Filozofowie są ponumerowani. Filozof z parzystym numerem najpierw podnosi prawą pałeczkę, filozof z nieparzystym numerem najpierw podnosi lewą pałeczkę.
4. **Rozwiązanie z arbitrem.** Zewnętrzny arbiter (lokaj, kelner) pilnuje, aby jednocześnie co najwyżej czterech (w ogólnym przypadku $N-1$) filozofów konkurowało o pałeczki. Jeśli naraz wszyscy filozofowie będą chcieli jeść, arbiter powstrzymuje jednego z nich aż do czasu, gdy któryś z filozofów skończy jeść.

3 Plan ćwiczenia

3.1 Implementacja

Zaimplementuj rozwiązanie problemu pięciu Filozofów w Javie, wykorzystując mechanizm semaforów, we wszystkich wspomnianych wariantach:

1. Rozwiązanie naiwne
2. Rozwiązanie z możliwością zagłodzenia
3. Rozwiązanie asymetryczne
4. Rozwiązanie z arbitrem

Można skorzystać z rozwiązań podanych w książce T. Weissa i Z. Gruzlewskiego [1] na stronach 55-56.

3.2 Porównanie rozwiązań

Korzystając z wykonanych implementacji:

- Uruchom eksperymenty dla różnej liczby filozofów i dla każdego wariantu implementacji (nie powodującego zakleszczenia).
- Zmierz średni czas oczekiwania każdego filozofa na dostęp do pałeczek. Wykonaj kilka pomiarów dla każdego przypadku testowego.

- Wyniki przedstaw na wykresach porównawczych, dbając o odpowiednią wizualizację (można wykorzystać np. wykresy pudełkowe).
- Sformułuj i zapisz wnioski. Czy średnie czasy oczekiwania są wyższe dla wariantu z możliwością zagłodzenia? Czy brak mechanizmów synchronizacji zwiększa czas oczekiwania na dostęp do zasobów?

Literatura

- [1] Z. Weiss, T. Gruzlewski, Programowanie współbieżne i rozproszone, WNT, Warszawa 1993.
- [2] Problem uczujących filozofów:
https://pl.wikipedia.org/wiki/Problem_ucztujcych_filozofw