

Modele Programowania Równoległego

[Strona główna](#) / [Moje kursy](#) / [MPR'20](#) / [Lab 4: MPI - badanie efektywności programu równoległego](#)
/ [Badanie efektywności programu równoległego](#)

Badanie efektywności programu równoległego

Cel ćwiczenia

Celem ćwiczenia jest zbadanie efektywności zrównoleglonego programu bazującego na metodzie Monte Carlo.

Wprowadzenie

Przebieg tych zajęć jest bezpośrednią kontynuacją poprzednich, a więc opiera się na następujących założeniach:

- Każdy posiada program korzystający z MPI, który potrafi policzyć liczbę π metodą Monte Carlo i poprawnie zmierzyć czas obliczeń. Komentarze załączone do zadania powinny pomóc z poprawną implementacją tym, którym nie udało się to w pełni za pierwszym podejściem.
- Każdy posiada dostęp do Zeusa i aktywny grant testowy, a także dostęp do Prometeusza i należy do grupy plggmpr2020. Usprawiedliwieniem od tego warunku będzie posiadanie zgłoszenia na helpdesku z datą nie nowszą niż piątek lub wyjątkowe przypadki (ustalane bezpośrednio).

W instrukcji nie będą powtórzone instrukcje dotyczące uruchamiania zadań na Zeusie / Prometeuszu / vnodach - w razie potrzeby proszę korzystać z poprzednich materiałów. Instrukcję można wykonać z wykorzystaniem vnodów, ale to środowisko nie daje warunków odpowiednich, żeby osiągnąć właściwe wnioski w sprawozdaniu. Z tego powodu bardzo istotne jest posiadanie działającej konfiguracji konta na Zeusie / Prometeuszu.

Przebieg ćwiczenia

Przygotowanie eksperymentu - 3 pkt

Upewnij się, że kod liczący π działa poprawnie - jeżeli zawierał inne elementy, np. wypisy częściowych czasów lub kilkakrotne powtórzenia i wyliczanie odchylenia standardowego, to na potrzeby tego laboratorium usuń te elementy. Schemat kodu powinien wyglądać następująco:

- inicjalizacja,
- bariera,
- pomiar czasu,
- obliczenia,
- komunikacja pomiędzy wątkami,
- pomiar czasu,
- wypis,
- finalizacja.

Uruchom program ze zmieniającą się liczbą wątków, od 1 do 12. Każde uruchomienie powinno policzyć w sumie tyle samo punktów - rozmiar problemu pozostaje stały, praca jednego procesora maleje (eksperyment nieskalowany). **Przygotuj skrypt, zapisz zrzut ekranu.**

Uruchom program jak poprzednio, ale za każdym razem każdy procesor powinien liczyć tyle samo punktów - rozmiar problemu rośnie, praca jednego procesora pozostaje stała (eksperyment skalowany). **Przygotuj skrypt, zapisz zrzut ekranu.**

Oba powyższe zadania połącz w jeden skrypt - **prześlij go jako wyniki tego laboratorium, razem ze zrzutami ekranu i kodem w ostatecznej formie.**

Uwaga: Nie ma potrzeby przejmować się zaokrągleniem przy podziale punktów pomiędzy wątki. Dzielenie całkowitoliczbowe spowoduje, że w najgorszym przypadku wszystkie wątki poza jednym będą liczyć o 1 punkt mniej, a więc dla N procesorów "zgubi się" N-1 punktów. W porównaniu z liczbami punktów wymaganymi do wykonania zadania, jest to pomijalny szczegół.

Wykonanie eksperymentu i sprawozdanie - 15 pkt

Tak przygotowany eksperyment posłuży do przygotowania **sprawozdania - załącz je w wyznaczonym miejscu.**

Uruchom swój kod w wersji skalowanej i nieskalowanej, dla liczb wątków 1-12, dla 3 różnych rozmiarów problemu:

- duży problem - taki, żeby wersja liczona na maksymalnej liczbie wątków wykonywała się co najmniej około minuty,
- mały problem - taki, żeby wersja liczona na jednym wątku wykonywała się poniżej sekundy (ale na tyle, żeby pomiar nie zwrócił 0),
- średni problem - według własnej inwencji.

Eksperyment powtórz kilkukrotnie, uśrednij czasy wykonania z wszystkich powtórzeń danego kroku (tzn. czas wykonania danej wersji dla danej liczby procesorów i danego rozmiaru powinien być średnią z kilku powtórzeń). Dla każdej serii utwórz 4 wykresy:

- zależność czasu wykonania od liczby procesorów,
- zależność przyspieszenia (speedup) od liczby procesorów (pamiętaj, że dla wersji skalowanej czas wykonania programu sekwencyjnego musi zostać pomierzony / obliczony dla każdej wielkości problemu),
- zależność efektywności (efficiency) od liczby procesorów,
- zależność części sekwencyjnej (serial fraction) od liczby procesorów.

Dla wykresów metryk dodaj do porównania wartości idealne. Sugerowane grupowanie wykresów: dwa bloki (skalowany i nieskalowany) po 4 wykresy przedstawiające 3 serie z różnymi rozmiarami problemu oraz wartości idealne.

Zwróć szczególną uwagę na / zawrzyj w sprawozdaniu następujące elementy:

- Wykresy powinny być czytelne bez dużego wysiłku - pamiętaj o tytule, podpisach osi, legendzie (jeżeli jest potrzebna), jednostkach na osiach. Zastanów się, jaka forma wykresu będzie poprawna - punkty, linie, punkty połączone liniami, wygładzone linie, słupki... Napisz dlaczego stosujesz akurat ten rodzaj wykresu - przy dobrym wyjaśnieniu normalnie nieakceptowalne formy mogą być w porządku.
- Każda grupa wykresów powinna być opatrzona komentarzem informującym z jakiego eksperymentu pochodzi, a każdy wykres wnioskami komentującymi znaczenie takiego przebiegu.
- Poprawnie wykonane ćwiczenie powinno dać co najmniej dwie istotnie różne charakterystyki metryk - opisz dlaczego widoczne jest akurat takie zachowanie i z czego ono wynika, a jeżeli nie zgadza się z idealnymi wartościami - dlaczego.
- Jeżeli nie jesteś w stanie uzyskać zadowalających wyników, a masz pewność, że rozmiary problemów są dobrze dobrane, opisz jakie wyniki były spodziewane i zaproponuj wyjaśnienie dlaczego nie udało się ich uzyskać.
- W komentarzach zawrzyj informacje o ilości komunikacji między wątkami dla każdego rozmiaru problemu.
- Wyjaśnij, czy wyliczasz przyspieszenie względne czy bezwzględne oraz jakie to ma konsekwencje w tym przypadku i w ogólnym przypadku.
- Jeżeli czegoś zupełnie nie rozumiesz, skontaktuj się z prowadzącym.

Preferowane formy sprawozdania to:

1. Arkusz kalkulacyjny z pomiarami, wyliczeniami, wykresami i komentarzami, o ile wyraźnie widoczne jest czego dotyczy dany tekst.
2. Plik PDF z wykresami i komentarzami; do tego w osobnym pliku lub na końcu PDF pomiary i wyliczenia.

Stosowanie się do takich formatów i sugerowanego grupowania przyspieszy proces oceniania.

Uwagi do Zeusa / Prometeusza:

- Upewnij się w trybie interaktywnym na małej próbce, że wszystko poprawnie się uruchamia i nie zaskoczy Cię błąd po długim oczekiwaniu na zasoby.
- Jeżeli zleciłeś zadanie do kolejki i po 10 minutach nadal nie zostało uruchomione (patrz lista statusów zadania w systemie SLURM), sprawdź czy ma przypisany czas startu: `squeue --start`. Jeżeli nie ma lub jest on bardzo odległy (np. kilka dni), anuluj zadanie i spróbuj ponownie bez flagi "exclusive". Upewnij się jednak, że wiesz co ona oznacza i jaki będzie to mieć wpływ na wyniki (należy to zawrzeć w sprawozdaniu).

Ostatnia modyfikacja: poniedziałek, 23 marca 2020, 09:39

[◀ Wyniki z Zeusa](#)

Przejdź do...

[Pytania do Lab 4 ▶](#)



Platforma e-Learningowa obsługiwana jest przez:
Centrum e-Learningu AGH oraz Uczelniane Centrum Informatyki AGH

Podsumowanie zasad przechowywania danych
Pobierz aplikację mobilną