

Sprawozdanie z Laboratorium 8: Przetwarzanie równoległe i rozproszone - Obliczanie liczby π z szeregu Leibniza

Piotr Trojan

Cel:

Celem zadania było doskonalenie umiejętności programowania z wykorzystaniem kolektywnego przesyłania komunikatów w środowisku MPI (Message Passing Interface). Zadanie polegało na opracowaniu programu obliczającego liczbę π z wykorzystaniem szeregu Leibniza oraz rozdzieleniu obliczeń pomiędzy wiele procesów w systemie rozproszonym. Program miał wykorzystywać komunikację równoległą, dzieląc liczbę składników szeregu między dostępne procesy.

Kroki:

1. Opracowanie algorytmu obliczania liczby π z szeregu Leibniza:

Szereg Leibniza dla liczby π jest opisany wzorem:

$$1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots = \frac{\pi}{4},$$

Główne zadanie polegało na równoległym obliczaniu sumy tego szeregu przez wiele procesów. Proces o ranku 0 miał pobrać liczbę składników, a następnie rozdzielić te składniki pomiędzy pozostałe procesy.

2. Rozdzielenie pracy między procesy:

Każdy proces w systemie obliczał sumę częściową szeregu Leibniza dla przypisanego zakresu składników. Jeśli liczba składników nie dzieliła się równo na procesy, pozostałe składniki były rozdzielane równomiernie pomiędzy procesy.

3. Użycie MPI do komunikacji i synchronizacji:

Zastosowano funkcje MPI do przesyłania danych pomiędzy procesami:

- **MPI_Bcast:** do wysyłania liczby składników (n) z procesu głównego (rank 0) do wszystkich pozostałych procesów.
- **MPI_Reduce:** do zbierania częściowych sum obliczonych przez procesy i sumowania ich w procesie głównym.

4. **Obliczenie końcowego wyniku:**

Po zebraniu wszystkich częściowych sum, proces o ranku 0 obliczał ostateczną wartość liczby π przez przemnożenie sumy przez 4, ponieważ szereg Leibniza daje wartość π podzieloną przez 4.

5. **Testowanie poprawności programu:**

Program był testowany dla różnych liczby składników szeregu i procesów. Wyniki były porównywane z wartością π , dostępną w standardowych bibliotekach matematycznych.

Testowanie programu:

- Program został przetestowany przy różnych liczbach składników szeregu oraz różnych liczbach procesów.
- Wyniki były zgodne z oczekiwaniami i bardzo zbliżone do wartości rzeczywistej liczby π .
- Program poprawnie obliczał π , nawet dla dużej liczby składników, przy dużej liczbie procesów.

Wnioski:

1. **Skalowalność:**

Program skalownie rozdzielał obliczenia pomiędzy procesy. Dzięki zastosowaniu MPI, obliczenia były wykonywane równolegle, co pozwoliło na szybsze obliczenia w porównaniu do wersji jednoprecinkowej.

2. **Efektywność komunikacji:**

Zastosowanie funkcji **MPI_Bcast** i **MPI_Reduce** umożliwiło efektywną wymianę danych między procesami, co przyczyniło się do prawidłowego obliczenia wartości π .

3. **Poprawność:**

Program poprawnie obliczał wartość liczby π , a wyniki były bardzo bliskie wartości rzeczywistej liczby π , co świadczy o poprawności implementacji.

Podsumowanie:

Zadanie pozwoliło na zapoznanie się z zaawansowanymi technikami programowania równoległego i rozproszonego w środowisku MPI. Implementacja algorytmu obliczania liczby π z szeregu Leibniza pokazała, jak efektywnie można rozdzielić obliczenia pomiędzy wiele procesów oraz jak wykorzystać kolektywne operacje MPI do synchronizacji procesów i przesyłania danych.