

Programowanie Równoległe - Zadanie 2

Dariusz Max Adamski

Specyfikacja

Intel Core i5-7200U @ 3.1GHz (2 fizyczne, 4 logiczne (2x2))

Windows 10

Visual C++

Punkt 1

Liczba iteracji	Czas (sekundy)	Wynik
10 ⁹	2.5119	3.141592653590

Punkt 2

Wynik jest niepoprawny dla wielowątkowego programu ponieważ wątki mogą jednocześnie modyfikować zmienne x i sum.

Zmienne prywatne: i

Zmienna współdzielona: x, sum

Większy czas przy 4 wątkach, niż przy 2 wiąże z koniecznością zmiany kontekstu procesorów fizycznych.

W punktach 2-6 występuje unieważnienie kopii linii pamięci.

Liczba wątków	Czas (sekundy)	Wynik	Przyspieszenie
4 wątki	1.2580	0.567588218417	1.997
2 wątki	1.1659	1.287002217587	2.154
1 wątek	2.3421	3.141592653590	1.072

Punkt 3

Dyrektywa `#pragma omp atomic` może sprawić, że kompilator zastąpi aktualizację zmiennej atomową instrukcją sprzętową (przykład: inkrementacja).

Niepodzielność można zapewnić na przykład dyrektywą `#pragma omp critical`.

Występuje drastyczne spowolnienie, ponieważ dostęp do sum musi być atomowy, nie mamy przyspieszenia w obliczaniu tej sumy w porównaniu do sekwencyjnego przetw., a w dodatku musimy zakładać blokady aby zapewnić atomowość.

Liczba wątków	Czas (sekundy)	Wynik	Przyspieszenie
4 wątki	98.0828	3.141592653590	0.026

2 wątki	80.0018	3.141592653590	0.031
1 wątek	20.4526	3.141592653590	0.123

Punkt 4

Jedna z najszybciej działających wersji programu.

Liczba wątków	Czas (sekundy)	Wynik	Przyspieszenie
4 wątki	1.2666	3.141592653590	1.983
2 wątki	1.2723	3.141592653590	1.974
1 wątek	2.4944	3.141592653590	1.007

Punkt 5

Podobnie do punktu 4, tylko z użyciem reduction, żeby nie pisać za dużo kodu :)

Liczba wątków	Czas (sekundy)	Wynik	Przyspieszenie
4 wątki	1.2630	3.141592653590	1.989
2 wątki	1.3656	3.141592653590	1.839
1 wątek	2.7695	3.141592653590	0.907

Punkt 6

Spowolnienia są spowodowane przez false-sharing.

Liczba wątków	Czas (sekundy)	Wynik	Przyspieszenie
4 wątki	1.4929	3.141592653590	1.683
2 wątki	2.7759	3.141592653590	0.905
1 wątek	4.8262	3.141592653590	0.520

Punkt 7

Krótszy czas obliczeń wystąpił przy słowach (3, 4), (11, 12), (19, 20), (27, 28), (35, 36) i (43, 44). Nasze słowo to double o wielkości 8 bajtów ($w = 8B$), a ponieważ odległości pomiędzy krótszymi czasami wykonania wynoszą 8 indeksów ($p_{n+1} - p_n = 11 - 3 = 19 - 11 = \dots = 8$), to długość linii pamięci wynosi $x = w(p_{n+1} - p_n) = 8B \cdot 8 = 64B$.

Napotkane trudności: Podczas wykonywania eksperymentu zapomniałem oznaczyć zmienną tablicy jako volatile, ale po zrobieniu tego eksperyment zakończył się powodzeniem.