

Lab nr 1 (część 2)

1. Określenie wykorzystywanego systemu równoległego: procesor, system operacyjny, kompilator

Procesor: Intel(R) Core(TM) i5-7300HQ CPU @ 2.50GHz (4 CPUs)

System: Windows 10

Kompilator: Visual Studio 2017

2. Tabela (następna strona).

2. Tabela.

Nazwa	Czas	Wpółczynnik	Uzasadnienie dla uzyskanej wielkości przyspieszenia
P1	7,466	-	-
P2 (4 wątki)	3,234	2,308596166	Przyspieszenie na skutek podziału pracy między wątki. Wynik niepoprawny ze względu na brak kontroli nad współdzielonymi zasobami i występujący przez to wyścig
P2 (2 wątki)	5,763	1,295505813	j. w. + spowolnienie pracy ze względu na mniejszą liczbę wątków
P3 (4 wątki)	119,822	0,062309092	Mocne spowolnienie wynika z zastosowania klauzuli atomic, która w każdej iteracji pętli dla każdego wątku wyłącza zmienną przechowującą sumę. Przez to wątki nie mogą działać współbieżnie i zmuszone są czekać na zwolnienie zmiennej, co rzutuje na długi czas działania
P3 (2 wątki)	63,567	0,117450879	j. w. + czas jest krótszy ze względu na mniejszą liczbę wątków (a zatem także mniejszą liczbę wątków stopowanych przez jeden wątek)
P4 (4 wątki)	3,827	1,950875359	Czas krótszy ze względu na podział pracy między wątki.
P4 (2 wątki)	5,291	1,411075411	j. w. + czas wydłużony względem poprzednika ze względu na mniejszą liczbę wątków
P5 (4 wątki)	3,791	1,969401213	Sytuacja analogiczna do kodu z pi4, lecz z wykorzystaniem innej implementacji
P5 (2 wątki)	5,294	1,410275784	j. w.
P6 (4 wątki)	12,618	0,591694405	Wzrost czasu przetwarzania spowodowany przez fakt fałszywego współdzielenia danych.
P6 (2 wątki)	11,871	0,628927639	j. w.

3. Określenie długości linii wg przygotowanego wzoru i na podstawie wyników eksperymentu z kodem.

Szybsze działanie programu zanotowałem co 8 miejsc w tablicy (większa szybkość występowała kolejno w: 1, 9, 17, 25, 33, 41 i 49 iteracji pętli – wyniki w dołączonym pliku „pi7-wyniki.txt”). Jako, że zadeklarowana przeze mnie tablica była typu **double** (słowa 4-bajtowe) to długość linii można obliczyć w następujący sposób:

$$4B * 8 = 64B$$

Zatem długość linii pp procesora w moim komputerze to **64 bajty**.

4. Opisanie własnymi słowami przebiegu eksperymentu wyznaczenia długości linii pp procesora, opis ewentualnych trudności napotkanych podczas realizacji zadania.

Pierwszym problemem jaki napotkałem było zrozumienie co eksperyment ma na celu wyliczyć. Gdy zrozumiałem wreszcie sens zadania napisanie kodu nie sprawiło mi problemów, lecz zaskoczył mnie długi czas wykonywania pojedynczych iteracji głównej pętli przechodzącej po utworzonej tablicy. Ze względu na to, że jedna próba obliczenia wartości PI zajmowała ok. 30 sec, zdecydowałem zmniejszyć liczbę iteracji (zmienną *num_steps*) dziesięciokrotnie, skracając czas wykonywania obliczeń. Następnie po analizie wyników zauważyłem stałą odległość między szybszymi czasami i wyliczyłem długość linii pp procesora.