

Pomiar czasu oraz zrównoleglenie pętli

PWIR_02_00.cpp

Pomiar czasu odbywa się przy pomocy biblioteki „chrono”. Przykładowe rozpoczęcie oraz zakończenie pomiaru czasu :

```
auto start = std::chrono::high_resolution_clock::now();  
auto end = std::chrono::high_resolution_clock::now();
```

PWIR_02_01.cpp

Zrównoleglenie pętli to jedna z podstawowych operacji OpenMP. Pozwala ona na znaczne skrócenie czasu potrzebnego na wykonanie zadań. Najlepszym sposobem na sprawdzenie działania jest prześledzenie przykładu. Zadany program mnoży wektor przez macierz. Bez zrównoleglenia operacji, czas potrzebny na wykonanie operacji jest dość spory.

Dzięki dyrektywie OpenMP wprowadzamy zrównoleglenie operacji. Dyrektywa przybiera następującą postać : *#pragma omp nazwa_dyrektywy opcje_i_parametry*

Dla omówienia składni dyrektywy niech posłużą nam przykład dyrektywy parallel, czyli operacja zrównoleglenia:

```
#pragma omp parallel for shared(A) private(B)
```

„for” – określa jaka pętla będzie zrównoleglona, w przykładzie będzie to pętla typu for,

„shared” – określenie zmiennych wspólnych,

„private” – określenie zmiennych prywatnych,

Do zmiennych wspólnych ma dostęp każdy wątek, natomiast do zmiennych prywatnych dostęp ma tylko dany wątek wykonujący operację.

Dla przykładu jeśli nasza pętla będzie miała 100 iteracji i mamy dostępne 4 wątki, to każdy wątek otrzyma 25 iteracji. Za zmienne wspólne przyjmuje się zmienne, które podczas obliczeń nie ulegają zmianie np. całkowita ilość iteracji. Zmienne prywatne natomiast będą wartościami które posłużą nam do obliczeń.

Zadanie:

Zmodyfikuj program tak by mnożenie macierzy wykonywało się kilka razy, następnie zrównolegl ten proces i wyświetl czas. Porównaj czasy przed i po modyfikacji.

Wyniki oraz program prześlij do swojego repozytorium. Umieść je w folderze o tej samej nazwie co ten PDF.