

Przetwarzanie współbieżne. Programowanie równoległe i rozproszone

Laboratorium 11

Cel:

- nabycie umiejętności pisania programów w środowisku OpenMP z wykorzystaniem puli wątków – zadań OpenMP

Zajęcia:

1. Utworzenie katalogu roboczego (np. lab_OpenMP_threadpool).
2. Utworzenie podkatalogu roboczego (np. sortowanie)
3. W podkatalogu roboczym rozpakowanie i uruchomienie dostarczonego programu realizującego sortowanie sekwencyjne i równoległe
4. Utworzenie podkatalogu roboczego (np. search_max)
5. W podkatalogu rozpakowanie i uruchomienie programu wyszukiwania wartości maksymalnej w tablicy
6. Uzupełnienie programu o definicje zadań (*tasks*) – dla wersji równoległej openmp wyszukiwania liniowego – w dyrektywie task użyć klauzuli *default(none)* i ustalić jak poprawnie i optymalnie przeprowadzać obliczenia (jakich zmiennych użyć)
7. Uzupełnienie programu o definicje zadań (*tasks*) – dla wersji równoległej openmp wyszukiwania binarnego (można wzorować się na procedurze sortowania przez scalanie *merge_sort_openmp* z programu sortowania) – należy założyć tworzenie pełnego drzewa wywołań binarnych, aż do momentu, kiedy podtablica ma już tylko jeden element (także w tym przypadku w dyrektywie task użyć klauzuli *default(none)*)
8. Zmodyfikowanie programu wyszukiwania binarnego, tak żeby po przekroczeniu pewnego poziomu w drzewie wywołań nie były generowane dalsze zadania, a algorytm korzystał z szybkiego wyszukiwania liniowego (można wzorować się na procedurze sortowania przez scalanie *merge_sort_openmp_2* z programu sortowania)

Dalsze kroki:

1. Napisanie programu liczącego całkę z wykorzystaniem puli wątków i dekompozycji w dziedzinie problemu
2. Napisanie zadań z Laboratorium 6 wykorzystując język C, dyrektywy OpenMP i pule wątków

Warunki zaliczenia:

1. Obecność na zajęciach i wykonanie co najmniej kroków 1-8
2. Oddanie jednostronicowego sprawozdania z krótkim odręcznym opisem zadania (cel, zrealizowane kroki, wnioski), kodem źródłowym procedur w C