

ITHPC 2014 - raport

Kacper Siora

25 kwietnia 2014

1 Opis zadania

W ramach laboratorium z przedmiotu *Introduction to High Performance Computing* zrealizowane zostało zadanie 3-cie, czyli mnożenie macierzy przez macierz. Zostały przygotowane trzy wersje programu: sekwencyjna, OpenMP oraz OpenMP dla Xeon Phi. Wszystkie programy i skrypty użyte do rozwiązania zadania można znaleźć w repozytorium <https://github.com/Siorski/usa>.

2 Rozwiązanie

Stworzone rozwiązania oraz zastosowane w nich kluczowe funkcje:

- program generujący macierze, wypełniający je losowo liczbami z zakresu od 0 do 100 oraz zapisujący macierze do pliku
- program mnożący macierze, które zapisane w plikach podawane są jako parametry uruchumieniowe
- skrypt automatyzujący uruchamianie kolejnych mnożeń
- dynamiczna alokacja pamięci dla macierzy
- mierzenie czasu wykonania całego programu, oraz jego kluczowych punktów, tj.
 - wczytywanie danych
 - główne obliczenia
 - zapis wyniku

2.1 Mierzenie czasu

Do mierzenia czasu została użyta biblioteka *time.h* oraz zawarta w niej funkcja *clock_gettime(CLOCK_MONOTONIC)*.

2.2 Wersje równoległe

Stworzenie równoległej wersji rozwiązania sprowadziło się do zrównoleglenia głównej części, w której macierze są mnożone:

```
#pragma omp parallel shared(macierzA, macierzB, macierzC) private(i,j,k)
{
    #pragma omp for schedule (static)
    for(i=0; i<ilosc_wierszy_A; i++){
        for(j=0; j<ilosc_kolumn_B; j++){
            for(k=0; k<ilosc_wierszy_B; k++){
                macierzC[i][j] += macierzA[i][k] * macierzB[k][j];
            }
        }
    }
}
```

Wszelkie próby zrównoleglenia fragmentu programu, w którym macierze są wczytywane z, bądź zapisywane do plików przynosiły niestety efekt odwrotny do pożądanego. Program OpenMP uruchamiany był na sigmie, gdzie korzystał on z 8 wątków.

2.3 Dane testowe

Do testów działania programu zostało wygenerowanych 6 par macierzy o rozmiarach:

Para 1 5 x 5 * 5 x 5

Para 2 517 x 740 * 740 x 493

Para 3 1029 x 1475 * 1475 x 981

Para 4 1541 x 2210 * 2210 x 1469

Para 5 2053 x 2945 * 2945 x 1957

Para 6 2565 x 3680 * 3680 x 2447

3 Porównanie czasów

3.1 Tabele

Wszystkie czasy w tabelach podane są w sekundach.

	5x5 * 5x5		
	<i>Sekwencyjny</i>	<i>OpenMP</i>	<i>XEON PHI</i>
<i>Czas wczytywania danych</i>	0,001477	0,001588	0,000627
<i>Czas obliczeń</i>	0,000002	0,000229	0,255085
<i>Czas zapisu danych</i>	0,001506	0,00099	0,000653
<i>Czas trwania całego programu</i>	0,003097	0,002898	0,256662

	517x740 * 740x493		
	<i>Sekwencyjny</i>	<i>OpenMP</i>	<i>XEON PHI</i>
<i>Czas wczytywania danych</i>	0,085524	0,090966	0,875284
<i>Czas obliczeń</i>	1,913065	0,357867	0,26042
<i>Czas zapisu danych</i>	0,074895	0,079641	0,426752
<i>Czas trwania całego programu</i>	2,074703	0,529671	1,565948

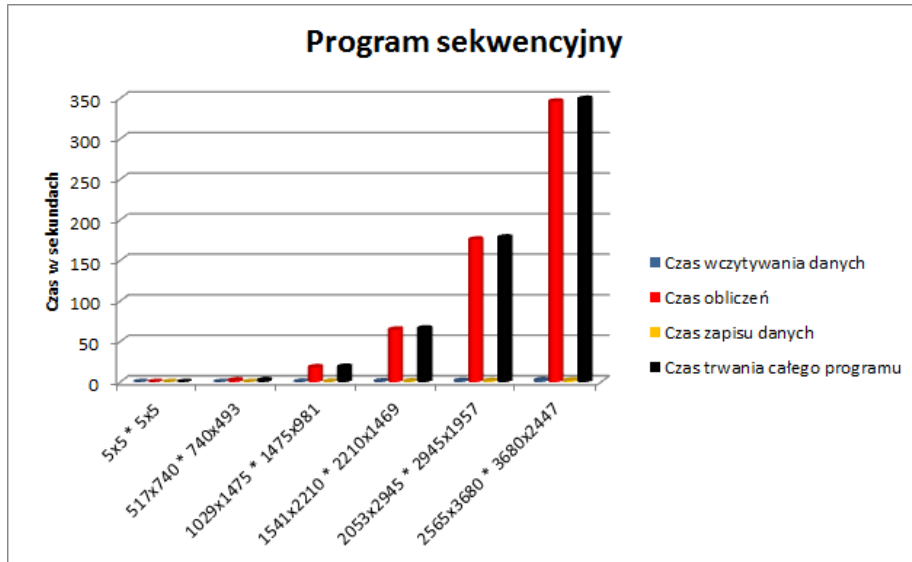
	1029x1475 * 1475x981		
	<i>Sekwencyjny</i>	<i>OpenMP</i>	<i>XEON PHI</i>
<i>Czas wczytywania danych</i>	0,332428	0,336021	3,481636
<i>Czas obliczeń</i>	18,193035	3,220273	0,292785
<i>Czas zapisu danych</i>	0,307626	0,322421	1,273766
<i>Czas trwania całego programu</i>	18,838361	3,884008	5,060543

	1541x2210 * 2210x1469		
	<i>Sekwencyjny</i>	<i>OpenMP</i>	<i>XEON PHI</i>
<i>Czas wczytywania danych</i>	0,750734	0,752672	7,78231
<i>Czas obliczeń</i>	64,640528	10,916685	0,384762
<i>Czas zapisu danych</i>	0,649202	0,687393	2,69992
<i>Czas trwania całego programu</i>	66,053089	12,368927	10,892418

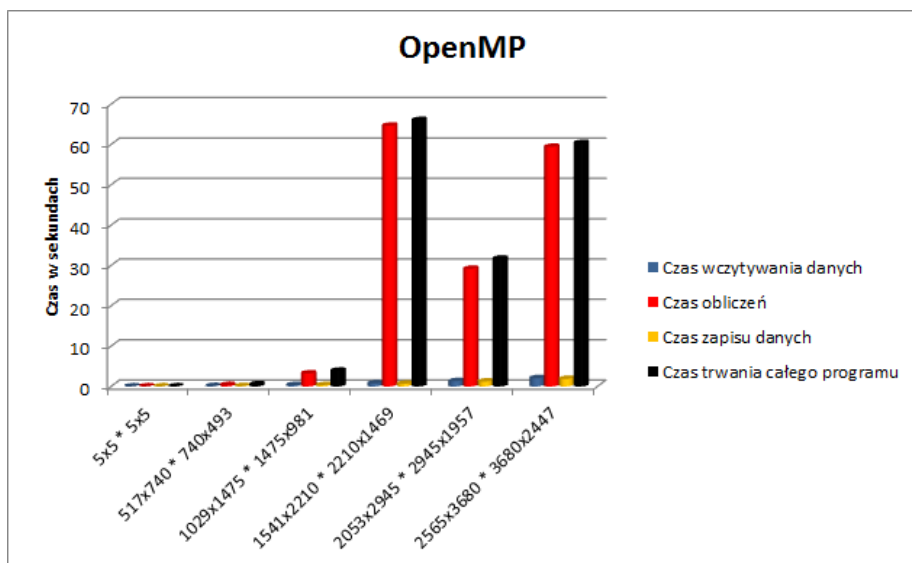
	2053x2945 * 2945x1957		
	<i>Sekwencyjny</i>	<i>OpenMP</i>	<i>XEON PHI</i>
<i>Czas wczytywania danych</i>	1,365159	1,308095	13,808813
<i>Czas obliczeń</i>	175,745871	29,156601	0,583053
<i>Czas zapisu danych</i>	1,155145	1,204566	4,947509
<i>Czas trwania całego programu</i>	178,28793	31,691602	19,382112

	2565x3680 * 3680x2447		
	<i>Sekwencyjny</i>	<i>OpenMP</i>	<i>XEON PHI</i>
<i>Czas wczytywania danych</i>	2,042326	2,037463	21,647846
<i>Czas obliczeń</i>	346,049103	59,394568	0,922356
<i>Czas zapisu danych</i>	1,741209	1,863713	7,723572
<i>Czas trwania całego programu</i>	349,253795	60,331661	30,358427

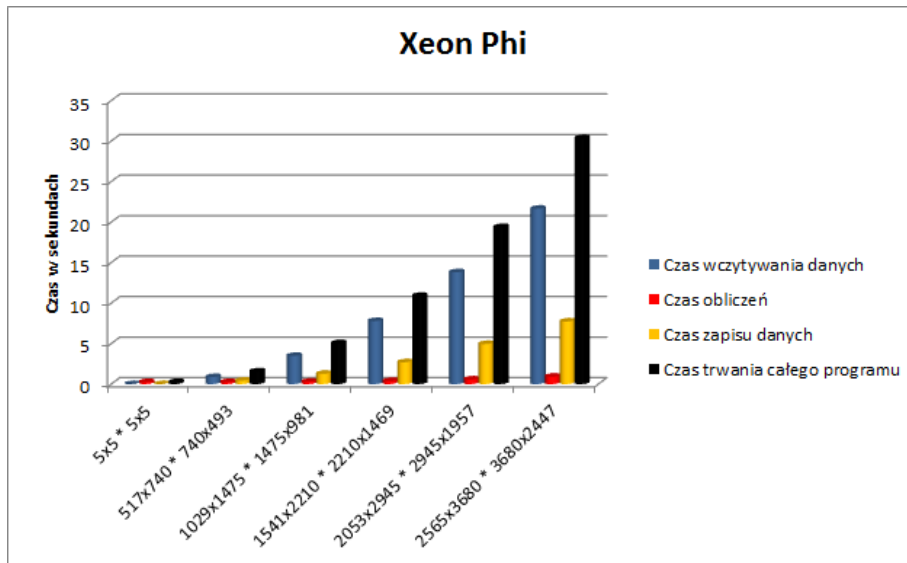
3.2 Wykresy



Wykres z czasami wykonania programu sekwencyjnego.



Wykres z czasami wykonania programu OpenMP.



Wykres z czasami wykonania programu na Xeon Phi.

4 Wnioski

Z wykresów oraz tabel, można jednoznacznie odczytać, iż programy wykonywane na Xeonie Phi działają najszybciej, co nie jest oczywiście żadnym zaskoczeniem. Dla małych macierzy różnice są jeszcze niezbyt znaczące, natomiast już dla największych macierzy jakie zostały przetestowane przyspieszenie jest widoczne - 2-krotnie szybciej niż program OpenMP oraz ponad 11-krotnie szybciej niż program sekwencyjny. Dwóm pierwszym programom najwięcej czasu zajmują główne obliczenia (ponad 90 % czasu). Xeon Phi natomiast na same obliczenia poświęca najmniej uwagi - zaledwie 3%. Zdecydowanie bardziej zajmujące dla procesora Intel jest wczytywanie danych oraz w mniejszym, ale nadal znacznym stopniu ich zapisywanie. Zatem sytuacja ma się zupełnie odwrotnie w przypadku programu sekwencyjnego oraz OpenMP w porównaniu do programu uruchamianego na procesorze Xeon Phi.