

Sprawozdanie Przetwarzanie równoległe Projekt 1 OMP

Dominik Doberski, Artur Olejnik

19 maja 2019

1 Wstęp

1.1 Temat

Mnożenie macierzy - porównanie efektywności metod –

- 3 pętle - kolejność pętli: jik, podział pracy przed pętlą 1
- 6 pętli -kolejność pętli: zewnętrznych ijk, wewnętrznych: ii,kk,jj podział pracy przed pętlą 2.

1.2 Autorzy

Artur Olejnik	122402	artur.olejnik@student.put.poznan.pl
Dominik Doberski	132207	dominik.doberski@student.put.poznan.pl

Grupa dziekańska I3

Termin zajęć: poniedziałek 16:50

1.3 Opis systemu obliczeniowego

Do wykonania pomiarów wykorzystano program CodeXL. Testy zrealizowano na jednostce obliczeniowej o poniższej specyfikacji:

- Nazwa: AMD FX-6300 Black Edition
- Częstotliwość taktowania: 3.5 GHz (max 3.8 GHz)
- Liczba procesorów fizycznych: 6
- Liczba procesorów logicznych: 6

- Liczba uruchamianych w systemie wątków: 6
- Wielkość i organizacja pamięci podręcznych procesora:
L1 dane: 6x16KB
L1 instrukcje: 3x64KB
L2: 3x2MB
L3: 8MB

2 Analiza algorytmu

2.1 Analiza kodu

Funkcja 1: 3 pętle – jik

```

1 void multiply_matrices_JIK() {
2 #pragma omp parallel
3 #pragma omp for
4 for (int j = 0; j < ROWS; j++){
5     for (int i = 0; i < ROWS; i++){
6         for (int k = 0; k < ROWS; k++){
7             matrix_r[i][j] += matrix_a[i][k] * matrix_b[k][j];
8         }
9     }
10 }
11 }
```

W powyższej funkcji pętla w 6. wierszu iteruje po kolumnach macierzy A i po wierszach macierzy B. Iloczyn tych dwóch elementów jest dodawany do pojedynczego elementu macierzy R znajdującego się w i-tym wierszu i j-tej kolumnie, które są wyznaczone w pętlach odpowiednio 5. i 4. wierszu. Złożoność algorytmu wynosi $O(n^3)$. Dyrektywy OpenMP w 2. i 3. wierszu wykorzystano do uruchomienia kodu na wielu wątkach, oraz podzielenia przetwarzania pętli na te wątki.

Funkcja 2: 6 pętli – zewnętrzne ijk, wewnętrzne ii, kk, jj

```

12 void multiply_matrices_6(int R){
13 #pragma omp parallel
14 for (int i = 0; i < ROWS; i += R) {
15 #pragma omp for
16     for (int j = 0; j < ROWS; j += R) {
17         for (int k = 0; k < ROWS; k += R) {
18             for (int ii = i; ii < i + R; ii++) {
19                 for (int kk = k; kk < k + R; kk++) {
20                     for (int jj = j; jj < j + R; jj++) {
21                         matrix_r[ii][jj] += matrix_a[ii][kk] * matrix_b[kk][jj];
```

```
22     }
23     }
24     }
25     }
26     }
27     }
28 }
```

3 Analiza przetwarzania

3.1 Przebieg

asdf

3.2 Wyniki

asdf

3.3 Prezentacja wzorców

asdf

3.4 Miary przetwarzania

asdf

4 Wnioski

Chujowy projekt.

4.1 Wypunktowania

To jest wypunktowanie:

- punkt
- punkt
- punkt

4.2 Wyliczenie

To jest wyliczenie:

1. pierwszy,
2. drugi,
3. trzeci.

4.3 Wykres

Na Rysunku ?? znajduje się przykładowy wykres.

4.4 Wzory

$$2 + 2 = 4 \tag{1}$$

$$E = mc^2 \tag{2}$$

$$\left[-\frac{\hbar^2}{2M} + V(X) \right] \Psi(x) = \mathcal{E} \Psi(x) \tag{3}$$

5 Podsumowanie