**«Основные операции с матрицами»**

1. Постановка задачи

Разработать программу для демонстрации основных операций с матрицами при помощи параллельного программирования.

1. Краткое описание

Распараллеливание было произведено по внешнему циклу, i, j независимы, индивидуальны для каждого потока.

1. Исходный код

#include <iostream>

#include <ctime>

#include "omp.h"

using namespace std;

int main()

{

setlocale(LC\_ALL, "");

int n = 1000;

//int s = 0;

int \*\*firstm = new int\*[n];

int \*\*secondm = new int\*[n];

int \*\*resultm = new int\*[n];

for (int i = 0; i < n; i++)

{

firstm[i] = new int[n];

secondm[i] = new int[n];

resultm[i] = new int[n];

}

srand(unsigned(NULL));

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

firstm[i][j] = rand() % 10;

secondm[i][j] = rand() % 10;

}

}

clock\_t begin = clock();

#pragma omp parallel for

////////////////Сложение матриц////////////////////

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

resultm[i][j] = firstm[i][j] + secondm[i][j];

}

}

clock\_t end1 = clock();

cout << "Сложение произведено\n\nВремя:" << (double)(end1 - begin) / CLOCKS\_PER\_SEC << endl;

///////////////Вычитание матриц//////////////////

#pragma omp parallel for

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

resultm[i][j] = firstm[i][j] - secondm[i][j];

}

}

clock\_t end2 = clock();

cout << "\nВычитание произведено\n\nВремя:" << (double)(end2 - begin) / CLOCKS\_PER\_SEC << endl;

/////////////Перемножение матриц///////////////

#pragma omp parallel for

for (int i = 0; i < n; i++)

{

for (int j = 0; j < n; j++)

{

int s = 0;

for (int l = 0; l < n; l++)

s += firstm[i][l] \* secondm[l][j];

resultm[i][j] = s;

}

}

clock\_t end = clock();

cout << "\nУмножение произведено\n\nВремя:" << (double)(end - begin) / CLOCKS\_PER\_SEC << endl;

for (int i = 0; i < n; i++)

{

delete[] firstm[i];

delete[] secondm[i];

delete[] resultm[i];

}

delete[] firstm;

delete[] secondm;

delete[] resultm;

system("pause");

return 0;

}

