Экзаменационная работа по дисциплине

«Параллельное программирование»

Студент гр. ПВ-31

Заруднев Павел

27.10.2014

Задание №14

Разработайте программы. Дана матрица вещественных чисел. Преобразовать матрицу таким образом, чтобы значение всех положительных элементов стало равным нулю.

1. Однопоточное приложение. Проверьте его работоспособность на небольшом объеме исходных данных. Добавить Open MP.
2. Приложение с использованием технологии MPI. Проверьте его работоспособность на очень большом объеме исходных данных на трех различных количествах узлов.

Решение.

#include <iostream>

#include <omp.h>

#include "time.h"

#include <mpi.h>

#define N 10000

float matrix[N][N];

void createMatrix();

void printMatrix();

void processMatrix();

void processMatrixByMPI();

int main() {

createMatrix();

printMatrix();

processMatrix();

puts("--------------------------------------------------");

printMatrix();

getchar();

return 0;

}

void createMatrix() {

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

matrix[i][j] = (float)(rand() % 1000) - 500;

}

}

}

void printMatrix() {

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

printf("%.1f ", matrix[i][j]);

}

printf("\n");

}

}

void processMatrix() {

#pragma omp parallel for

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

if (matrix[i][j] > 0) {

matrix[i][j] = 0;

}

}

}

}

void processMatrixByMPI() {

int rank;

MPI\_Init(&argc, &argv); // получаем кол-во процессов

MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank); // опредляем норме процесса в комуникаторе

MPI\_Scatter(source\_matrix, N \* N, MPI\_FLOAT, dest\_matrix, N \* N, MPI\_FLOAT, 0, MPI\_COMM\_WORLD);

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

if (matrix[i][j] > 0) {

matrix[i][j] = 0;

}

}

}

MPI\_Finalize();

}

**Тестовые данные**

|  |  |  |  |
| --- | --- | --- | --- |
| Количество данных | 5 000 | 10 000 | 20 000 |
| Однопоточное приложение | 188 | 703 | 2703 |
| OpenMP | 172 | 671 | 2688 |
| MPI |  |  |  |

188/172 = 1,093

703/671 = 1,044

2703/2688 = 1,006



