МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение   
высшего профессионального образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №14

РАБОТА ЗАЩИЩЕНА С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | А. Ю. Сыщиков |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2  OpenMP, Распараллеливание вычислений |
|  |
| по дисциплине: [Системы с параллельной обработкой информации](https://pro.guap.ru/inside_s#subjects/2436975) |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 1742 |  |  |  |  |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2021

1. **Цель работы**

Реализовать и распараллелить с помощью OpenMP различные алгоритмы.

|  |  |  |  |
| --- | --- | --- | --- |
| № варианта | Размерность задачи | Тип элемента вектора | Алгоритм |
| 2 | 20x60 | Знаковый короткий целый | Сортировка столбцов матрицы по убыванию |

1. **Текст программы**

#include <omp.h>

#include <time.h>

#include <locale.h>

#include <iostream>

using namespace std;

constexpr auto N = 20, M = 6;

int main(int argc, char\* argv[])

{

setlocale(0, "");

/\*Инициализация и вывод A\*/

short int A[N][M];

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

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

A[i][j] = rand() % 100;

}

}

cout << "A:";

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

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

cout << " " << A[i][j];

}

cout << "\n ";

}

cout << endl;

int i, j, n, k;

//парал. область. Общий массив А, индивидуальные данные (i, n, j, k)

#pragma omp parallel shared(A) private(i, n, j, k)

{

n = omp\_get\_thread\_num();//получить номер нити

#pragma omp for //фор автоматически разделит свои иттераци между всеми нитями, что в него войдут

for (j = 0; j < M; j++) {

printf("Нить %d сортирует столбец %d\n", n, j);

//далее надо сортировать столбцы. J - индекс столбца уже дал нам фор выше, то есть далее идем по i до N, по строчкам

for (k = 0; k < N - 1; k++) {

for (i = 0; i < N - 1 - k; i++) {

if (A[i][j] < A[i + 1][j])

{

printf("\tНить %d сравнила элементы [%d][%d] и [%d][%d] и поменяла их местами\n", n, i, j, i + 1, j);

short int temp = A[i][j];

A[i][j] = A[i + 1][j];

A[i + 1][j] = temp;

}

else {

printf("\t\tНить %d сравнивает элементы [%d][%d] и [%d][%d] и не поменяла их.\n", n, i, j, i + 1, j);

}

}

}

}

}

cout << "A:";

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

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

cout << " " << A[i][j];

}

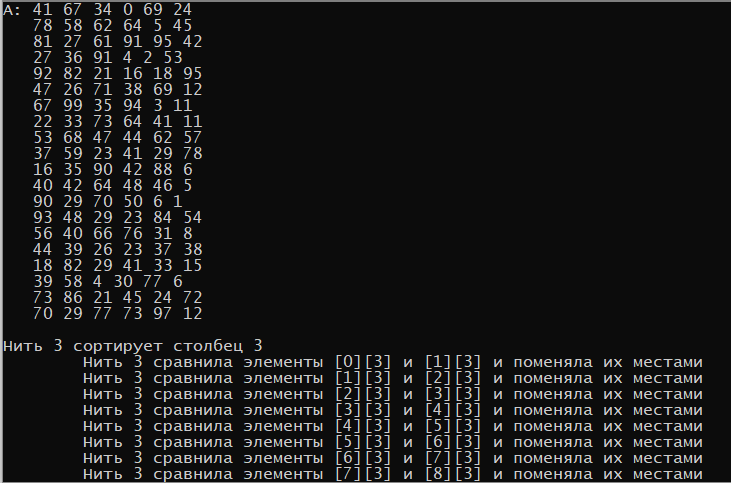
cout << "\n ";

}

cout << endl;

}

1. **Результат работы программы**



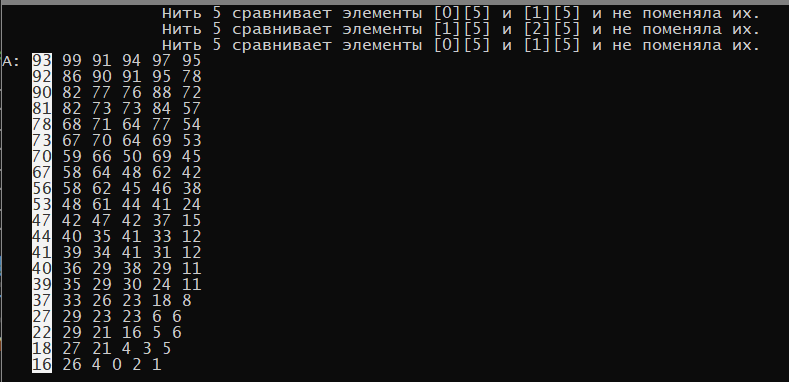


Рисунок 1. Результат работы программы.