

ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»
Факультет компьютерных наук
Департамент программной инженерии

ОТЧЕТ

ВЫЧИСЛЕНИЕ АЛГЕБРАИЧЕСКОГО ДОПОЛНЕНИЯ
ВСЕХ ЭЛЕМЕНТОВ МАТРИЦЫ С ИСПОЛЬЗОВАНИЕМ OPENMP

Вариант 3.

Автор пояснительной записки, _____
студент ФКН,
“Программная инженерия”,
группа БПИ193

Доржсурэн Тоголдор

Москва 2020

1. ПОСТАНОВКА ЗАДАЧИ

Текст задания: Найти алгебраическое дополнение для каждого элемента матрицы. Входные данные: целое положительное число n , произвольная матрица A размерности $n \times n$. Количество потоков является входным параметром, при этом размерность матриц может быть не кратна количеству потоков. Использовать OpenMP.

2. ТЕОРЕТИЧЕСКАЯ СПРАВКА О ВЫЧИСЛЕНИИ ОБРАТНОЙ МАТРИЦЫ

Алгебраическим дополнением элемента a_{ij} матрицы A называется число

$$A_{ij} = (-1)^{i+j} * M_{ij}$$

где M_{ij} — дополнительный минор, определитель матрицы, получающейся из исходной матрицы A путём вычёркивания i -й строки и j -го столбца.

Для вычисления по многопоточности использован итеративный параллелизм с помощью прагмы “#pragma omp parallel”

```
//Forks tasks into {thread_count} number of threads
#pragma omp parallel shared(matrix, result) num_threads(thread_count)
{
    //Distributes loop into threads
    #pragma omp for
    for (int i = 0; i < dim; i++) {
        for (int j = 0; j < dim; j++) {
            result[i][j] = algebraic_Complement(matrix, i, j, dim);
        }
    }
    //Critical section
    #pragma omp critical
    {
        cout << "Thread: " << omp_get_thread_num() << " working. "
        << "Total threads: " << omp_get_num_threads() << endl;
    }
}
```

3. ОПИСАНИЕ ОБЛАСТИ ДОПУСТИМЫХ ЗНАЧЕНИЙ ВХОДНЫХ ПАРАМЕТРОВ

Программа имеет 2 входных параметра:

- размерность матрицы ($2 < \text{целое число} < 20$)
- количество потоков ($1 < \text{целое число} < 20$).

ПРИМЕРЫ РАБОТЫ ПРОГРАММЫ

Пример 1

```
Enter dimension length for square matrix: 5
Enter number of threads: 2
-----
Matrix
-----
8 8 4 6 6
5 0 6 9 3
7 4 2 7 0
4 3 3 0 1
1 6 7 0 9
Thread: 1 working. Total threads: 2
Thread: 0 working. Total threads: 2
-----
Result
-----
1161 -875 -1113 -343 1320
430 -770 56 -6 422
-1548 1740 882 830 -1674
1032 -558 444 -840 -88
-1032 902 674 324 -600
```

Пример 2

```
Enter dimension length for square matrix: 5
Enter number of threads: 3
-----
Matrix
-----
6 5 4 9 4
5 6 8 2 9
3 0 2 1 6
3 1 1 0 1
5 3 5 0 8
Thread: 0 working. Total threads: 3
Thread: 1 working. Total threads: 3
Thread: 2 working. Total threads: 3
-----
Result
-----
8 -109 131 -10 -46
-15 297 -400 -43 148
-42 387 -379 -71 118
-133 361 -418 -19 209
61 -615 721 109 -289
```

Пример 3

```
Enter dimension length for square matrix: 0
Dimension length must be more than 2 and less than 20.
Enter dimension length for square matrix: 3
Enter number of threads: 25
Number of threads cannot be less than 1 or more than 20.
Enter number of threads: █
```

Пример 4

```
Enter dimension length for square matrix: 7
Enter number of threads: 9
-----
Matrix
-----
0 9 6 2 3 7 7
7 0 3 1 2 9 8
9 6 3 3 6 5 4
0 7 9 7 7 9 5
6 5 0 1 6 4 4
0 5 1 0 9 6 9
2 6 7 4 8 3 5
Thread: 2 working. Total threads: 9
Thread: 1 working. Total threads: 9
Thread: 3 working. Total threads: 9
Thread: 4 working. Total threads: 9
Thread: 5 working. Total threads: 9
Thread: 7 working. Total threads: 9
Thread: 8 working. Total threads: 9
Thread: 6 working. Total threads: 9
Thread: 0 working. Total threads: 9
-----
Result
-----
204 522 4488 -8938 1944 3278 -4918
1355 -2371 17657 -34307 11006 13949 -20950
-4915 8989 -87875 173921 -57322 -70359 108998
-316 -332 1150 -2200 1278 2230 -2708
6410 -9572 100510 -200382 66326 81474 -126492
-3072 4054 -43278 84554 -27368 -34112 53566
2196 -3436 32108 -61424 20450 23372 -37690
```