

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
**«САМАРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
имени академика С.П. Королева»**

ПАРАЛЛЕЛЬНОЕ ПРОГРАММИРОВАНИЕ

Отчёт по лабораторной работе №2

РЕАЛИЗАЦИЯ ПЕРЕМНОЖЕНИЯ МАТРИЦ ПО ТЕХНОЛОГИИ
OPENMP

Томашайтис Павел

Группа 6313-100503D

1 Цель работы

- 1) Добавить в программу перемножения двух матриц возможность использования технологии OpenMP.
- 2) Проверить корректность перемножения матриц с использованием технологии OpenMP.
- 3) Сравнить статистические показатели для времени перемножения матриц по технологии OpenMP и времени последовательного перемножения матриц.

2 Добавление в программу перемножения двух матриц возможности использования технологии OpenMP

Программа перемножения двух матриц, написанная на языке C++, была модифицирована для реализации перемножения 2 матриц по технологии OpenMP. К классу SquareMatrix была добавлен метод `openmp_dot()`, который выполняет перемножение 2 матриц по технологии OpenMP и замеряет время выполнения этой операции. Содержимое метода представлено на рисунке 1.

```
double openmp_dot(const SquareMatrix& rhs)
{
    if (rhs._size != _size)
        throw std::invalid_argument("Error! Matrix size was not the same.");
    SquareMatrix result = SquareMatrix(_size);
    auto a = _data;
    auto b = rhs._data;
    auto c = result._data;
    double begin = omp_get_wtime();
    int i, j, k;
    #pragma omp parallel for collapse(2) shared(a, b, c) private(i, j, k)
    for (i = 0; i < _size; i++)
    {
        for (j = 0; j < _size; j++)
        {
            for (k = 0; k < _size; k++)
            {
                c[i][k] += a[i][j] * b[j][k];
            }
        }
    }
    double end = omp_get_wtime();
    *this = result;
    return end - begin;
}
```

Рисунок 1 – метод перемножения 2 матриц по технологии OpenMP.

При реализации технологии OpenMP распараллеливались итерации цикла for с учётом вложенности (за счёт использования collapse). Время измерялось встроенной в библиотеку omp.h функцией omp_get_wtime().

К доступным операциям для взаимодействия с программой добавилась операция перемножения матриц по технологии OpenMP (команда --openmp или -o). При этом результаты перемножения матриц и файлы со статистикой сохраняются в отдельные каталоги. Так, для последовательного перемножения выделены каталоги result1 и statistics1, для перемножения по технологии OpenMP – каталоги result2 и statistics2.

Пример работы программы:

```
Input command -> --openmp
OpenMP calculating starts
Available threads: 8
Matrices 100x100 – products calculated (0.000417 s).
Matrices 200x200 – products calculated (0.003019 s).
Matrices 300x300 – products calculated (0.009473 s).
Matrices 400x400 – products calculated (0.022417 s).
Matrices 500x500 – products calculated (0.043930 s).
Matrices 600x600 – products calculated (0.079409 s).
Matrices 700x700 – products calculated (0.131148 s).
Matrices 800x800 – products calculated (0.177053 s).
Matrices 900x900 – products calculated (0.257966 s).
Matrices 1000x1000 – products calculated (0.383683 s).
```

Рисунок 2 – Пример выполнения перемножения двух матриц по технологии OpenMP.

3 Проверка корректности перемножения двух матриц по технологии OpenMP

Для проверки корректности перемножения двух матриц по технологии OpenMP был использован частично изменённый скрипт check_matrix_production.py, написанный на языке Python в 1 лабораторной работе. Результат проверки корректности перемножения матриц оказался

положительным, следовательно ошибок в алгоритме перемножения матриц допущено не было.

Результат проверки корректности перемножения двух матриц с использованием технологии OpenMP:

```
Check matrix products for OpenMP calculating
100x100 100/100 matrix products corrected
200x200 100/100 matrix products corrected
300x300 100/100 matrix products corrected
400x400 100/100 matrix products corrected
500x500 100/100 matrix products corrected
600x600 100/100 matrix products corrected
700x700 100/100 matrix products corrected
800x800 100/100 matrix products corrected
900x900 100/100 matrix products corrected
1000x1000 100/100 matrix products corrected
```

Рисунок 3 – Результат проверки корректности перемножения двух матриц по технологии OpenMP

4 Сравнение статистических показателей для времени перемножения матриц по технологии OpenMP и времени последовательного перемножения матриц

Для сравнения статистических показателей для времени перемножения матриц по технологии OpenMP и времени последовательного перемножения матриц использовался частично изменённый скрипт `statistics.py`, написанный на языке Python в 1 лабораторной работе. Кроме вычисления среднего, медианы, дисперсии среднеквадратического отклонения, коэффициента эксцесса, коэффициента асимметрии, доверительного интервала (надёжность = 0,95) для выборки из временных интервалов, полученных при перемножении матриц, дополнительно сравнивается изменение времени

перемножения матриц в зависимости от их размера при использовании различных технологий.

На рисунке 4 представлены значения статистических показателей для времени произведения матриц размером 1000 на 1000 по технологии последовательного произведения и OpenMP.

```
Matrices 1000x1000:  
Mean: 2.08070430000000003  
Median: 2.08956500000000003  
Dispersion: 0.08731959164851  
STD: 0.2954988860359883  
Skewness: 0.21097764315233125  
Kurtosis: -0.9553595271210895  
Confidence interval for GAMMA = 0.95: (2.021775525931297, 2.1396330740687035)
```

```
Matrices 1000x1000:  
Mean: 0.38368271  
Median: 0.3706115  
Dispersion: 0.0014127826735459  
STD: 0.03758700139071884  
Skewness: 3.341752744817938  
Kurtosis: 12.817175461849965  
Confidence interval for GAMMA = 0.95: (0.37618706105666033, 0.39117835894333963)
```

Рисунок 4 – Значения статистических показателей для времени произведения матриц размером 1000 на 1000 по технологии последовательного произведения (сверху) и OpenMP (снизу).

Сравнение изменения времени перемножения матриц в зависимости от их размера при использовании различных технологий выполнено графическим образом и представлено на рисунке 5.

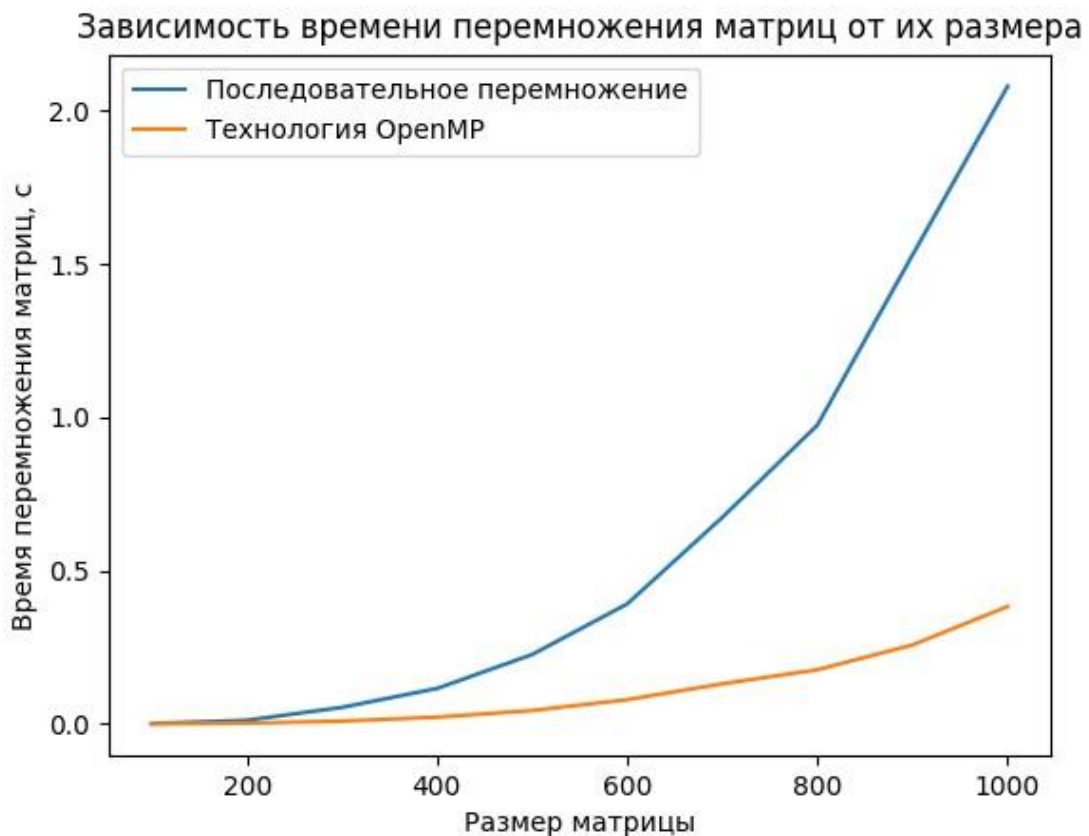


Рисунок 5 – График зависимости времени перемножения матриц от их размера для двух технологий.

5 Выводы

В данной лабораторной работе было реализовано перемножение матриц с использованием технологии OpenMP, а также выполнено сравнение скорости перемножения матриц при параллельном и последовательном исполнении программы.

Последовательное перемножение пары матриц размером 1000 на 1000 занимает в среднем 2,81 секунды, из чего следует, что 100 пар матриц

данного размера будут перемножаться в течение 5 минут, а для перемножения всех пар матриц необходимо будет затратить около 15 минут. При использовании технологии OpenMP пара матриц размером 1000 на 1000 перемножается за 0,38 секунды, 100 пар матриц 1000 на 1000 – за 38 секунд, а на перемножение всех пар будет затрачено около 2 минут. Преимущество использования технологии OpenMP в сравнении с последовательным исполнением программы также легко проследить по графику, представленному на рисунке 5.

Таким образом, использование концепции параллелизма в соответствии с технологией OpenMP позволяет существенно повысить производительность программы перемножения матриц, из чего можно сделать вывод о полезности использования концепций параллелизма при работе высокопроизводительных программ.