

**МИНОБРНАУКИ РОССИИ**

Федеральное государственное автономное образовательное  
учреждение высшего образования «Южный федеральный  
университет»

Институт математики, механики и компьютерных наук им.  
И.И.Воровича Кафедра информатики и вычислительного  
эксперимента

Направление подготовки 02.04.02 – Фундаментальная информатика и  
информационные технологии

**ОТЧЕТ ПО ТЕМЕ**  
**«Блочное перемножение матриц»**

Выполнил:  
магистр 1 года 6 группы  
Домбровская А.В.

## Реферат

В основе эксперимента лежит задача перемножения двух вещественных матриц с использованием технологии OpenMP, где симметричная матрица  $A$  умножается справа на верхне-треугольную матрицу  $B$ . В памяти матрица  $A$  хранится как нижне-треугольная матрица в виде одномерного массива по блочным столбцам. Матрица  $B$  хранится в виде одномерного массива по блочным столбцам.

Задачу требовалось решить тремя способами:

- выполнить блочное последовательное перемножение без использования параллельности;
- выполнить перемножение различных пар блоков параллельно;
- выполнить перемножение каждого отдельного блока параллельно;

По результатам численных вычислений был проведен анализ реализаций решения задачи с матрицами двух различных типов данных и с использованием двух компиляторов. Результаты работы созданной программы были сверены с результатами работы не блочного перемножения матриц.

## **Основная часть**

### **Характеристики компьютера**

Processor: Intel® Core™ i5-8300 CPU @ 2.30GHz; 4 cores;

L3: 8 MB;

L2: 1 MB;

L1: 256 KB;

RAM: DDR4, 8 GB, clock speed: 2667 MHz.

### **Характеристики компиляторов**

Programming language: C++;

Compiler: MSVC2019 v14.16.27023, compiler option: -O2; OpenMP v. 2.0;

Compiler: GCC v. 8.3.0, compiler option: -Ofast; OpenMP v. 4.5;

OS: Windows 10 Education v1903

## Результаты для MSVC2019

Table 1: Double

Размер блока	Последовательное выполнение	Параллельное умножение различных пар блоков	Параллельное умножение каждого отдельного блока
1	138.053846	>2000	>2000
6	27.846353	4.180622	161.967205
10	20.785998	3.361242	43.516863
15	18.506775	3.187226	18.357782
20	16.594656	3.293959	17.703939
24	16.254516	2.840943	11.640018
30	17.348134	3.028692	11.695159
36	17.254998	2.904548	12.209977
40	17.584581	3.084133	13.200694
60	18.209153	2.949771	10.034205
72	19.155187	2.916666	9.517371
80	19.119615	3.228302	9.243961
96	20.945467	3.720901	9.612271
120	20.590226	3.269624	9.660214
144	21.693518	4.227735	10.275369
160	23.026291	5.179367	10.967122
180	22.659279	3.988872	10.420435
240	23.959650	5.115210	11.061807
360	25.057874	5.204494	12.109950
480	27.247476	6.830716	14.326010
720	31.604413	11.754522	24.239679

Table 2: Float

<b>Размер блока</b>	<b>Последовательное выполнение</b>	<b>Параллельное умножение различных пар блоков</b>	<b>Параллельное умножение каждого отдельного блока</b>
<b>1</b>	126.76485	>2000	>2000
<b>6</b>	26.831554	6.443908	135.507158
<b>10</b>	19.616943	4.603957	38.301464
<b>15</b>	17.104209	3.968277	17.048714
<b>20</b>	15.852124	3.447534	14.182890
<b>24</b>	15.655131	3.398218	9.956785
<b>30</b>	16.751926	3.406890	10.086870
<b>36</b>	16.455042	3.164320	9.887745
<b>40</b>	16.435133	3.226078	8.873160
<b>60</b>	17.461879	3.064032	8.907447
<b>72</b>	18.286148	3.062220	8.203078
<b>80</b>	18.817116	3.283811	8.487900
<b>96</b>	19.400041	3.374524	8.609305
<b>120</b>	20.274852	3.094009	8.407928
<b>144</b>	20.872614	3.750691	8.605896
<b>160</b>	21.778877	4.566494	9.063349
<b>180</b>	21.730373	3.719191	8.948711
<b>240</b>	22.680135	4.614893	10.270536
<b>360</b>	24.215600	4.807746	11.031471
<b>480</b>	25.814551	5.477103	11.143063
<b>720</b>	27.479652	7.066395	14.969616

## Результаты для GCC

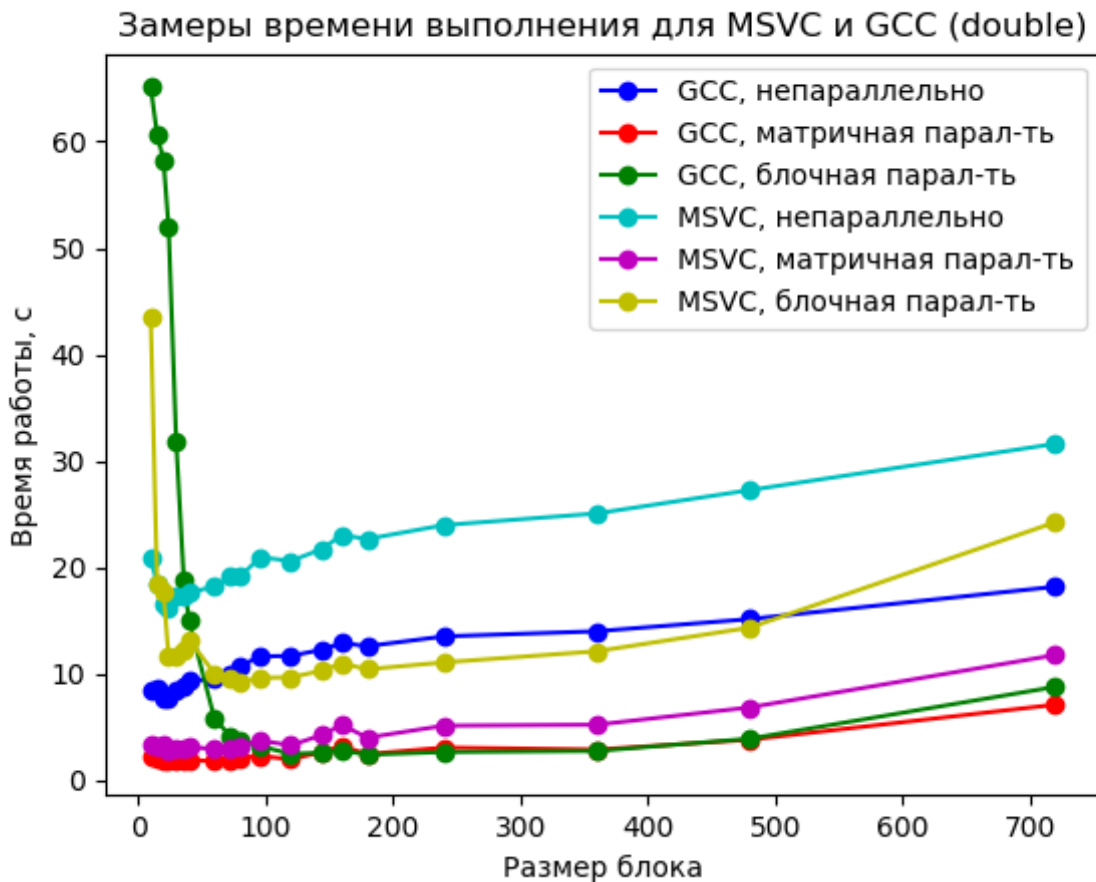
Table 3: Double

Размер блока	Последовательное выполнение	Параллельное умножение различных пар блоков	Параллельное умножение каждого отдельного блока
1	48.546000	13.410000	>4000
6	8.46	2.341000	>2000
10	8.344000	2.202000	841.107000
15	8.583000	1.935000	248.689000
20	7.758000	1.844000	105.270000
24	7.678000	1.817000	62.030000
30	8.524000	1.839000	31.774000
36	8.754000	1.887000	18.780000
40	9.323000	1.910000	15.029000
60	9.506000	1.766000	5.792000
72	9.978000	1.762000	4.190000
80	10.716000	2.081000	3.657000
96	11.628000	2.272000	3.184000
120	11.676000	1.952000	2.461000
144	12.247000	2.624000	2.570000
160	12.887000	3.188000	2.779000
180	12.584000	2.443000	2.391000
240	13.507000	3.058000	2.642000
360	13.986000	2.870000	2.738000
480	15.131000	3.792000	3.891000
720	18.176000	7.093000	8.766000

Table 4: Float

<b>Размер блока</b>	<b>Последовательное выполнение</b>	<b>Параллельное умножение различных пар блоков</b>	<b>Параллельное умножение каждого отдельного блока</b>
<b>1</b>	46.56000	11.35000	>4000
<b>6</b>	7.806000	2.043000	>2000
<b>10</b>	7.090000	1.769000	770.822000
<b>15</b>	7.072000	2.144000	244.338000
<b>20</b>	7.923000	2.030000	110.936000
<b>24</b>	8.089000	1.995000	66.423000
<b>30</b>	8.678000	1.966000	34.324000
<b>36</b>	9.389000	2.081000	20.549000
<b>40</b>	9.733000	2.087000	15.998000
<b>60</b>	9.916000	1.959000	6.118000
<b>72</b>	10.387000	1.952000	4.563000
<b>80</b>	11.175000	2.232000	3.913000
<b>96</b>	11.639000	2.122000	3.205000
<b>120</b>	12.275000	2.048000	2.685000
<b>144</b>	12.600000	2.385000	2.551000
<b>160</b>	12.998000	3.081000	2.640000
<b>180</b>	13.310000	2.500000	2.682000
<b>240</b>	13.610000	3.226000	2.562000
<b>360</b>	14.683000	3.466000	3.127000
<b>480</b>	16.126000	3.169000	3.169000
<b>720</b>	17.575000	4.155000	4.512000

## Сравнительный анализ компиляторов



По результатам проведенных экспериментов было установлено, что результаты компиляции программы с помощью GCC v8.3.0 значительно превосходят по скорости результаты, полученные при компиляции MSVC2019. Также на графике можно заметить, что в случае компиляции MSVC2019 параллельное перемножение двух различных пар блоков в два раза быстрее параллельного перемножения каждого отдельного блока. В случае GCC компиляции скорость между соответственными способами отличается мало.

Следует также обратить внимание, что скорость работы программы в случае, когда размер блока равен 1 или 6, для последовательного перемножения с компилятором GCC в три раза быстрее, чем соответствующее перемножение с MSVC2019, однако в случае параллельного перемножения каждого отдельного блока GCC значительно проигрывает в скорости MSVC2019.



## Заключение

По результатам проведенных исследований можно сделать вывод, что скорость перемножения матриц во многом зависит от выбора способа их перемножения. В частности, в данной работе было показано, что параллельное перемножение различных пар блоков в подавляющем большинстве случаев превосходит по скорости выполнения параллельное умножение каждого блока отдельно и непараллельное умножение. Однако данное утверждение справедливо исключительно для компилятора GCC. В случае использования компилятора MSVC2019 необходимо также подбирать размер блоков, на которые будут разбиваться матрицы, так как это оказывает большое влияние на скорость выполнения программы.