Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

Лабораторная работа №2 по курсу

«Операционные системы»

Группа: М80-206Б-22

Студентка: Шипилова Т.П.

Преподаватель: Миронов Е.С.

Оценка: ____________________

Дата: 01.12.2023 г.

Москва, 2023

Постановка задачи

Вариант 6.

Умножение матриц, содержащих комплексные числа.

Составить программу на языке Си, обрабатывающую данные в многопоточном режиме. При обработки использовать стандартные средства создания потоков операционной системы (Windows/Unix).

Ограничение максимального количества потоков, работающих в один момент времени, должно быть задано ключом запуска вашей программы. Так же необходимо уметь продемонстрировать количество потоков, используемое вашей программой с помощью стандартных средств операционной системы.

Общий метод и алгоритм решения

Использованные системные вызовы:

1. thread(matrix_multiple, ref(matr1), ref(matr2), ref(answer), start, end, m, n, k); – создаёт новый поток;
2. threads[i].join(); – ожидает завершения потока.

Программа условно делит матрицы на части, передает их потокам, где в одном потоке обрабатывается только определенное количество строк первой матрицы, то есть формируем ответ также построчно.

Сами матрицы храним в линеаризованном виде, используя вектор пары. Такой способ хранения данных выбран из-за особенностей данных. Они представлены комплексными числами, то есть содержат действительную и мнимую части. Умножение таких чисел имеет вид: $(a + bi) * (c + di) = ac + adi + bci - bd.$

Код программы

```
#include<iostream>
#include<chrono>
#include<vector>
#include<thread>
#include<mutex>
using namespace std;
using namespace std::chrono;

mutex mtx;


void matrix_multiple(std::vector<std::pair <double, double> >& a,
std::vector<std::pair <double, double> >& b,
                   std::vector<std::pair <double, double> >& ans, size_t start,
size_t end, int m, int n, int k){
   for (size_t i = start; i < end; i++){
       for (size_t j = 0; j < n; j++){
           mtx.lock();
           double d_ans = 0.0; double m_ans = 0.0;
           for (size_t l = 0; l < k; l++){
               std::pair pair1 = a[m*i+l];
               std::pair pair2 = b[n*l+j];
               d_ans += pair1.first * pair2.first - pair1.second * pair2.second;
               m_ans += pair1.first * pair2.second + pair1.second * pair2.first;
               }
           ans.push_back(make_pair(d_ans,m_ans));
           mtx.unlock();
       }
   }
}


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

   if (argc != 2){
       cout << "Incorrect usage. Arguments are not instantiated" << endl;
       cout << "Usage: ./lr2 number_of_treads" << endl;
       exit(1);
   }

   size_t num_threads = atoi(argv[1]);

   //Матрица хранится в векторы пары. В первой ячейке действительная часть, во
второй мнимая
   //Пример a + b*i, arr.first = a, arr.second = b
   //Умножение таких чисел: (a+bi)*(c+di) = ac+adi+bci-bd, тогда результат:
res.first=ac-bd, res.second=ad+bc

   cout << "Enter the dimension of the matrices to be multiplied to fill them
with random numbers" << endl;
   cout << "Matrix format: m*n, n*k, enter 3 natural numbers" << endl;
   int m,n,k; cin >> m >> n >> k; cout << endl;
   if (m < num_threads){
       cout << "No need for treads. Parameter m is less then number of threads"
<< endl;
```

```
        exit(1);
    }
    if (m < 0 || n < 0 || k < 0){
        cout << "Incorrect values of matrix dimensions. Values should be natural"
<< endl;
        exit(1);
    }

    vector <pair <double, double> > matr1;
    for (int i = 0; i < m*n; i++){
        double a,b;
        a = rand() % 100;
        b = rand() % 100;
        matr1.push_back(make_pair(a,b));
    }

    vector <pair <double, double> > matr2;
    for (int i = 0; i < k*n; i++){
        double a,b;
        a = rand() % 100;
        b = rand() % 100;
        matr2.push_back(make_pair(a,b));
    }
    vector <pair <double, double> > answer(n*k);

    vector<thread> threads(num_threads);

    size_t block_size = m / num_threads;
    size_t remainder = m % num_threads;

    auto begining = std::chrono::high_resolution_clock::now();

    size_t start = 0;
    for (int i = 0; i < num_threads; i++) {
        size_t end = start + block_size;
        if (i < remainder) {
            end++;
        }
        threads[i] = thread( matrix_multiple, ref(matr1), ref(matr2),
ref(answer), start, end, m, n, k);
        start = end;
    }

    for (int i = 0; i < num_threads; i++) {
        threads[i].join();
    }

    auto ending = std::chrono::high_resolution_clock::now();

    duration<double> sec = ending - begining;
    cout << "Result: ";
    cout << sec.count() << " s" << std::endl;
    answer.clear(); matr1.clear(); matr2.clear();

    return 0;
}
```

Протокол работы программы

Тестирование:

```
./lr2 1
Enter the dimension of the matrices to be multiplied to fill them with random numbers
Matrix format: m*n, n*k, enter 3 natural numbers
6 6 6
Result: 0.000411662 s
./lr2 2
Enter the dimension of the matrices to be multiplied to fill them with random numbers
Matrix format: m*n, n*k, enter 3 natural numbers
6 6 6
Result: 0.000893935 s
./lr2 3
Enter the dimension of the matrices to be multiplied to fill them with random numbers
Matrix format: m*n, n*k, enter 3 natural numbers
6 6 6
Result: 0.00234427 s
./lr2 4
Enter the dimension of the matrices to be multiplied to fill them with random numbers
Matrix format: m*n, n*k, enter 3 natural numbers
6 6 6
Result: 0.00123548 s
./lr2 5
Enter the dimension of the matrices to be multiplied to fill them with random numbers
Matrix format: m*n, n*k, enter 3 natural numbers
6 6 6
Result: 0.00164288 s
```

Количество потоков	Время, с	Ускорение	Эффективность
1	0.000411662	1	1
2	0.000893935	0,460505518	0,230252759
3	0.00234427	0,175603493	0,058534498
4	0.00123548	0,333200052	0,083300013
5	0.00164288	0,250573383	0,050114677

При анализе таблицы становится понятно, что выделение потоков значительно превышает по времени математические операции. При 4 потоках можем увидеть небольшое повышение эффективности. Можно объяснить это полным задействованием ресурсов машины, то есть 4 потоков на 2 ядрах.

Strace:

```
strace ./lr2 1
```

```
execve("./lr2", ["./lr2", "1"], 0x7fff0007a8a8 /* 74 vars */) = 0
brk(NULL)                               = 0x5650ca00b000
arch_prctl(0x3001 /* ARCH_??? */, 0x7fff18ea20b0) = -1 EINVAL (Недопустимый аргумент)
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f447022b000
access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (Нет такого файла или каталога)
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=68035, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 68035, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f447021a000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2260296, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 2275520, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f446fe00000
mprotect(0x7f446fe9a000, 1576960, PROT_NONE) = 0
mmap(0x7f446fe9a000, 1118208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x9a000) = 0x7f446fe9a000
mmap(0x7f446ffab000, 454656, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1ab000) = 0x7f446ffab000
mmap(0x7f447001b000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x21a000) = 0x7f447001b000
mmap(0x7f4470029000, 10432, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f4470029000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=125488, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 127720, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f44701fa000
mmap(0x7f44701fd000, 94208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7f44701fd000
mmap(0x7f4470214000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1a000) = 0x7f4470214000
mmap(0x7f4470218000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1d000) = 0x7f4470218000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) = 832
```

```
     pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64)
= 784

     pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48,
848) = 48

     pread64(3,
"\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\244;\374\204(\337f#\315I\214\234\f\256\271\32"..., 68, 896)
= 68

     newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2216304, ...}, AT_EMPTY_PATH) = 0

     pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64)
= 784

     mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f446fa00000

     mmap(0x7f446fa28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x28000) = 0x7f446fa28000

     mmap(0x7f446fbbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1bd000) = 0x7f446fbbd000

     mmap(0x7f446fc15000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x214000) = 0x7f446fc15000

     mmap(0x7f446fc1b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS,
-1, 0) = 0x7f446fc1b000

     close(3)                               = 0

     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=940560, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 942344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f4470113000

     mmap(0x7f4470121000, 507904, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0xe000) = 0x7f4470121000

     mmap(0x7f447019d000, 372736, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x8a000) = 0x7f447019d000

     mmap(0x7f44701f8000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0xe4000) = 0x7f44701f8000

     close(3)                               = 0

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f4470111000

     arch_prctl(ARCH_SET_FS, 0x7f44701123c0) = 0

     set_tid_address(0x7f4470112690)        = 7795

     set_robust_list(0x7f44701126a0, 24)    = 0

     rseq(0x7f4470112d60, 0x20, 0, 0x53053053) = 0

     mprotect(0x7f446fc15000, 16384, PROT_READ) = 0

     mprotect(0x7f44701f8000, 4096, PROT_READ) = 0

     mprotect(0x7f4470218000, 4096, PROT_READ) = 0

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f447010f000
```

```
     mprotect(0x7f447001b000, 45056, PROT_READ) = 0

     mprotect(0x5650c939b000, 4096, PROT_READ) = 0

     mprotect(0x7f4470265000, 8192, PROT_READ) = 0

     prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

     munmap(0x7f447021a000, 68035)          = 0

     getrandom("\x9a\x2f\xd0\xb6\x33\xfd\xc0\x66", 8, GRND_NONBLOCK) = 8

     brk(NULL)                              = 0x5650ca00b000

     brk(0x5650ca02c000)                    = 0x5650ca02c000

     futex(0x7f447002977c, FUTEX_WAKE_PRIVATE, 2147483647) = 0

     newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...},
AT_EMPTY_PATH) = 0

     write(1, "Enter the dimension of the matri"..., 86Enter the dimension of the matrices
to be multiplied to fill them with random numbers

     ) = 86

     write(1, "Matrix format: m*n, n*k, enter 3"..., 49Matrix format: m*n, n*k, enter 3
natural numbers

     ) = 49

     newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...},
AT_EMPTY_PATH) = 0

     read(0, 6 6 6

     "6 6 6\n", 1024)                = 6

     write(1, "\n", 1

     )                          = 1

     rt_sigaction(SIGRT_1, {sa_handler=0x7f446fa91870, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_RESTART|SA_SIGINFO, sa_restorer=0x7f446fa42520}, NULL, 8)
= 0

     rt_sigprocmask(SIG_UNBLOCK, [RTMIN RT_1], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f446f1ff000

     mprotect(0x7f446f200000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f446f9ff910,
parent_tid=0x7f446f9ff910, exit_signal=0, stack=0x7f446f1ff000, stack_size=0x7fff00,
tls=0x7f446f9ff640} => {parent_tid=[7816]}, 88) = 7816

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     futex(0x7f446f9ff910, FUTEX_WAIT_BITSET|FUTEX_CLOCK_REALTIME, 7816, NULL,
FUTEX_BITSET_MATCH_ANY) = 0

     write(1, "Result: 0.00216319 s\n", 21Result: 0.00216319 s

     )  = 21
     lseek(0, -1, SEEK_CUR)                 = -1 ESPIPE (Недопустимая операция смещения)

     exit_group(0)                          = ?

     +++ exited with 0 +++

     strace ./lr2 4

     execve("./lr2", ["./lr2", "4"], 0x7ffe5378aeb8 /* 74 vars */) = 0

     brk(NULL)                              = 0x55d515213000

     arch_prctl(0x3001 /* ARCH_??? */, 0x7ffc053dbdb0) = -1 EINVAL (Недопустимый аргумент)

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f14d9da9000

     access("/etc/ld.so.preload", R_OK)     = -1 ENOENT (Нет такого файла или каталога)

     openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=68035, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 68035, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f14d9d98000

     close(3)                               = 0

     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2260296, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 2275520, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f14d9a00000

     mprotect(0x7f14d9a9a000, 1576960, PROT_NONE) = 0

     mmap(0x7f14d9a9a000, 1118208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x9a000) = 0x7f14d9a9a000

     mmap(0x7f14d9bab000, 454656, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1ab000) = 0x7f14d9bab000

     mmap(0x7f14d9c1b000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x21a000) = 0x7f14d9c1b000

     mmap(0x7f14d9c29000, 10432, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS,
-1, 0) = 0x7f14d9c29000

     close(3)                               = 0

     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=125488, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 127720, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f14d9d78000

     mmap(0x7f14d9d7b000, 94208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x3000) = 0x7f14d9d7b000

     mmap(0x7f14d9d92000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1a000)
= 0x7f14d9d92000

     mmap(0x7f14d9d96000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x1d000) = 0x7f14d9d96000

     close(3)                               = 0
```

```
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) = 832
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48
pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\244;\374\204(\337f#\315I\214\234\f\256\271\32"..., 68, 896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2216304, ...}, AT_EMPTY_PATH) = 0
pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784
mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f14d9600000
mmap(0x7f14d9628000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f14d9628000
mmap(0x7f14d97bd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7f14d97bd000
mmap(0x7f14d9815000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) = 0x7f14d9815000
mmap(0x7f14d981b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f14d981b000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3
read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=940560, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 942344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f14d9c91000
mmap(0x7f14d9c9f000, 507904, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe000) = 0x7f14d9c9f000
mmap(0x7f14d9d1b000, 372736, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x8a000) = 0x7f14d9d1b000
mmap(0x7f14d9d76000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe4000) = 0x7f14d9d76000
close(3)                                = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f14d9c8f000
arch_prctl(ARCH_SET_FS, 0x7f14d9c903c0) = 0
set_tid_address(0x7f14d9c90690)         = 7886
set_robust_list(0x7f14d9c906a0, 24)     = 0
rseq(0x7f14d9c90d60, 0x20, 0, 0x53053053) = 0
mprotect(0x7f14d9815000, 16384, PROT_READ) = 0
mprotect(0x7f14d9d76000, 4096, PROT_READ) = 0
```

```
     mprotect(0x7f14d9d96000, 4096, PROT_READ) = 0

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f14d9c8d000

     mprotect(0x7f14d9c1b000, 45056, PROT_READ) = 0

     mprotect(0x55d513e5d000, 4096, PROT_READ) = 0

     mprotect(0x7f14d9de3000, 8192, PROT_READ) = 0

     prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

     munmap(0x7f14d9d98000, 68035)          = 0

     getrandom("\xba\x1a\xc1\x17\xb2\x3f\x7c\x5a", 8, GRND_NONBLOCK) = 8

     brk(NULL)                              = 0x55d515213000

     brk(0x55d515234000)                    = 0x55d515234000

     futex(0x7f14d9c2977c, FUTEX_WAKE_PRIVATE, 2147483647) = 0

     newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...},
AT_EMPTY_PATH) = 0

     write(1, "Enter the dimension of the matri"..., 86Enter the dimension of the matrices
to be multiplied to fill them with random numbers

     ) = 86

     write(1, "Matrix format: m*n, n*k, enter 3"..., 49Matrix format: m*n, n*k, enter 3
natural numbers

     ) = 49

     newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...},
AT_EMPTY_PATH) = 0

     read(0, 6 6 6

     "6 6 6\n", 1024)                = 6

     write(1, "\n", 1

     )                         = 1

     rt_sigaction(SIGRT_1, {sa_handler=0x7f14d9691870, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_RESTART|SA_SIGINFO, sa_restorer=0x7f14d9642520}, NULL, 8)
= 0

     rt_sigprocmask(SIG_UNBLOCK, [RTMIN RT_1], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f14d8dff000

     mprotect(0x7f14d8e00000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f14d95ff910,
parent_tid=0x7f14d95ff910, exit_signal=0, stack=0x7f14d8dff000, stack_size=0x7fff00,
tls=0x7f14d95ff640} => {parent_tid=[7890]}, 88) = 7890

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f14d85fe000
```

```
     mprotect(0x7f14d85ff000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f14d8dfe910,
parent_tid=0x7f14d8dfe910, exit_signal=0, stack=0x7f14d85fe000, stack_size=0x7fff00,
tls=0x7f14d8dfe640} => {parent_tid=[0]}, 88) = 7891

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f14d37ff000

     mprotect(0x7f14d3800000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f14d3fff910,
parent_tid=0x7f14d3fff910, exit_signal=0, stack=0x7f14d37ff000, stack_size=0x7fff00,
tls=0x7f14d3fff640} => {parent_tid=[7892]}, 88) = 7892

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f14d2ffe000

     mprotect(0x7f14d2fff000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f14d37fe910,
parent_tid=0x7f14d37fe910, exit_signal=0, stack=0x7f14d2ffe000, stack_size=0x7fff00,
tls=0x7f14d37fe640} => {parent_tid=[7893]}, 88) = 7893

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     write(1, "Result: 0.00305929 s\n", 21Result: 0.00305929 s

     )  = 21

     lseek(0, -1, SEEK_CUR)                  = -1 ESPIPE (Недопустимая операция смещения)

     exit_group(0)                           = ?

     +++ exited with 0 +++

     tanya@tanya:~/Рабочий стол/OOS/OS3sem/2.1$ strace ./lr2 4

     execve("./lr2", ["./lr2", "4"], 0x7ffce3478138 /* 74 vars */) = 0

     brk(NULL)                               = 0x55dff86e8000

     arch_prctl(0x3001 /* ARCH_??? */, 0x7ffe0a207260) = -1 EINVAL (Недопустимый аргумент)

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f83af5e2000

     access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (Нет такого файла или каталога)

     openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=68035, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 68035, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f83af5d1000

     close(3)                                = 0
     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=2260296, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 2275520, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f83af200000

     mprotect(0x7f83af29a000, 1576960, PROT_NONE) = 0

     mmap(0x7f83af29a000, 1118208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x9a000) = 0x7f83af29a000

     mmap(0x7f83af3ab000, 454656, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1ab000) = 0x7f83af3ab000

     mmap(0x7f83af41b000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x21a000) = 0x7f83af41b000

     mmap(0x7f83af429000, 10432, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS,
-1, 0) = 0x7f83af429000

     close(3)                                = 0

     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=125488, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 127720, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f83af5b1000

     mmap(0x7f83af5b4000, 94208, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x3000) = 0x7f83af5b4000

     mmap(0x7f83af5cb000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1a000)
= 0x7f83af5cb000

     mmap(0x7f83af5cf000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x1d000) = 0x7f83af5cf000

     close(3)                                = 0

     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0P\237\2\0\0\0\0\0"..., 832) =
832

     pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64)
= 784

     pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48,
848) = 48

     pread64(3,
"\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\244;\374\204(\337f#\315I\214\234\f\256\271\32"..., 68, 896)
= 68

     newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2216304, ...}, AT_EMPTY_PATH) = 0

     pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64)
= 784

     mmap(NULL, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f83aee00000

     mmap(0x7f83aee28000, 1658880, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x28000) = 0x7f83aee28000
```

```
     mmap(0x7f83aefbd000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1bd000) = 0x7f83aefbd000

     mmap(0x7f83af015000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x214000) = 0x7f83af015000

     mmap(0x7f83af01b000, 52816, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS,
-1, 0) = 0x7f83af01b000

     close(3)                               = 0

     openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3

     read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

     newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=940560, ...}, AT_EMPTY_PATH) = 0

     mmap(NULL, 942344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f83af4ca000

     mmap(0x7f83af4d8000, 507904, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0xe000) = 0x7f83af4d8000

     mmap(0x7f83af554000, 372736, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x8a000) = 0x7f83af554000

     mmap(0x7f83af5af000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0xe4000) = 0x7f83af5af000

     close(3)                               = 0

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f83af4c8000

     arch_prctl(ARCH_SET_FS, 0x7f83af4c93c0) = 0

     set_tid_address(0x7f83af4c9690)        = 7923

     set_robust_list(0x7f83af4c96a0, 24)    = 0

     rseq(0x7f83af4c9d60, 0x20, 0, 0x53053053) = 0

     mprotect(0x7f83af015000, 16384, PROT_READ) = 0

     mprotect(0x7f83af5af000, 4096, PROT_READ) = 0

     mprotect(0x7f83af5cf000, 4096, PROT_READ) = 0

     mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f83af4c6000

     mprotect(0x7f83af41b000, 45056, PROT_READ) = 0

     mprotect(0x55dff7a94000, 4096, PROT_READ) = 0

     mprotect(0x7f83af61c000, 8192, PROT_READ) = 0

     prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

     munmap(0x7f83af5d1000, 68035)          = 0

     getrandom("\xe6\x09\xcb\x4b\x6f\xf2\xcd\x7e", 8, GRND_NONBLOCK) = 8

     brk(NULL)                              = 0x55dff86e8000

     brk(0x55dff8709000)                    = 0x55dff8709000

     futex(0x7f83af42977c, FUTEX_WAKE_PRIVATE, 2147483647) = 0

     newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...},
AT_EMPTY_PATH) = 0
```

```
     write(1, "Enter the dimension of the matri"..., 86Enter the dimension of the matrices
to be multiplied to fill them with random numbers

     ) = 86

     write(1, "Matrix format: m*n, n*k, enter 3"..., 49Matrix format: m*n, n*k, enter 3
natural numbers

     ) = 49

     newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...},
AT_EMPTY_PATH) = 0

     read(0, 6

     "6\n", 1024)                    = 2

     read(0, 6

     "6\n", 1024)                    = 2

     read(0, 6

     "6\n", 1024)                    = 2

     write(1, "\n", 1

     )                         = 1

     rt_sigaction(SIGRT_1, {sa_handler=0x7f83aee91870, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_RESTART|SA_SIGINFO, sa_restorer=0x7f83aee42520}, NULL, 8)
= 0

     rt_sigprocmask(SIG_UNBLOCK, [RTMIN RT_1], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f83ae5ff000

     mprotect(0x7f83ae600000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f83aedff910,
parent_tid=0x7f83aedff910, exit_signal=0, stack=0x7f83ae5ff000, stack_size=0x7fff00,
tls=0x7f83aedff640} => {parent_tid=[7924]}, 88) = 7924

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f83addfe000

     mprotect(0x7f83addff000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f83ae5fe910,
parent_tid=0x7f83ae5fe910, exit_signal=0, stack=0x7f83addfe000, stack_size=0x7fff00,
tls=0x7f83ae5fe640} => {parent_tid=[7925]}, 88) = 7925

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f83ad5fd000

     mprotect(0x7f83ad5fe000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0
     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f83addfd910,
parent_tid=0x7f83addfd910, exit_signal=0, stack=0x7f83ad5fd000, stack_size=0x7fff00,
tls=0x7f83addfd640} => {parent_tid=[7926]}, 88) = 7926

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7f83acdfc000

     mprotect(0x7f83acdfd000, 8388608, PROT_READ|PROT_WRITE) = 0

     rt_sigprocmask(SIG_BLOCK, ~[], [], 8)   = 0

     clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CL
ONE_SETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f83ad5fc910,
parent_tid=0x7f83ad5fc910, exit_signal=0, stack=0x7f83acdfc000, stack_size=0x7fff00,
tls=0x7f83ad5fc640} => {parent_tid=[7927]}, 88) = 7927

     rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0

     write(1, "Result: 0.00470203 s\n", 21Result: 0.00470203 s

     )  = 21

     lseek(0, -1, SEEK_CUR)                 = -1 ESPIPE (Недопустимая операция смещения)

     exit_group(0)                          = ?

     +++ exited with 0 +++
```

Вывод

В ходе данной лабораторной работы я научилась создавать в своей программе потоки и использовать их для экономии времени вычислений. Я разделяла матрицу на части и передавала ее потокам. К сожалению, это, по сути, не могло ускорить решения, так как сложность все равно осталась $O(n*m*k)$, где n,m,k - размерности умножаемых матриц. Но при выделении определенного числа потоков можно было увидеть небольшое повышение эффективности, что свидетельствует о том, что распараллеливание вычислений способствует ускорению работы программы.