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: М8О-211Б-23

Студент: Амелина А.Е.

Преподаватель: Бахарев В.Д.

Оценка: ____________________

Дата: 25.12.24

Москва, 2024

Постановка задачи

Цель работы.

Целью является приобретение практических навыков в управлении потоками в ОС и в обеспечении синхронизации между потоками.

Задание.

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.

Также необходимо уметь продемонстрировать количество потоков, используемое вашей программой с помощью стандартных средств операционной системы.

В отчете привести исследование зависимости ускорения и эффективности алгоритма от входных данных и количества потоков. Получившиеся результаты необходимо объяснить.

Вариант 10.

Решить систему линейных уравнений методом Гаусса.

Общий метод и алгоритм решения

Использованные системные вызовы:

- int sem_wait(sem_t *semaphore); уменьшает значение семафора (semaphore), если значение = 0, то вызов блокируется до тех пор, пока нельзя будет выполнить вычитание (пока не произойдет sem_post)
- int sem_post(sem_t *semaphore); увеличивает значение семафора (semaphore) на единицу. Работает в паре с sem_wait.
- int sem_destroy(sem_t *semaphore); уничтожает семафор, на который указывает semaphore.
- int sem_init(sem_t *semaphore, int (0), unsigned int max_threads); инициализирует семафор по адресу на который указывает semaphore. Второй аргумент отвечает за то, каким им пользоваться. Если значение = 0, то семафор является общим для потоков процесса, иначе он общий для процессов.
- int pthread_create((pthread_t *thread, const pthread_attr_t *attr, void *(*routine) (void *), void *arg); Создает поток с начальной функцией и заданными аргументами.
- int pthread_join(pthread_t threads, void ** value); Дожидается завершения потока

Для выполнения данной лабораторной работы я изучила указанные выше системные вызовы.

Программа принимает 2 аргумента – размер матрицы n и количество потоков max_threads, используемое программой. Создается и заполняется значениями случайная матрица коэффициентов matrix и вектор свободных членов b.

В функции Gauss_method создаются потоки, каждый из которых получает копию данных(ThreadData), включая матрицу и вектор. Потоки выполняют прямой ход метода Гаусса. Каждый поток обрабатывает определенные строки матрицы, исключая элементы ниже главной диагонали и таким образом матрица приводится к верхнетреугольному виду. Для синхронизации используется семафор, чтобы основной поток дождался завершения всех потоков.

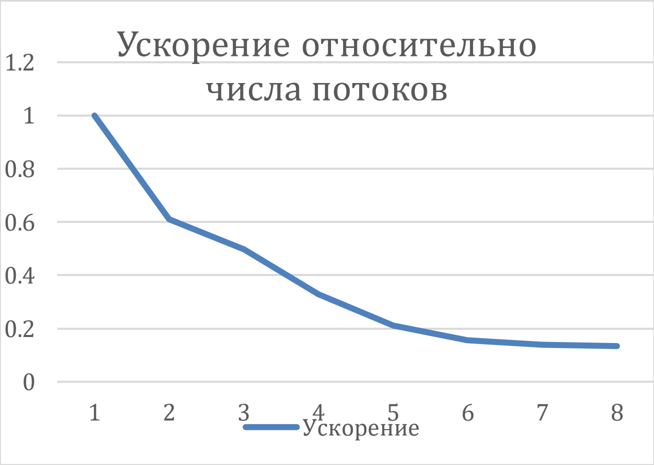
Основной поток выполняет обратный ход метода Гаусса, вычисляя значения неизвестных x(корни уравнения) и сохраняет их в векторе b.

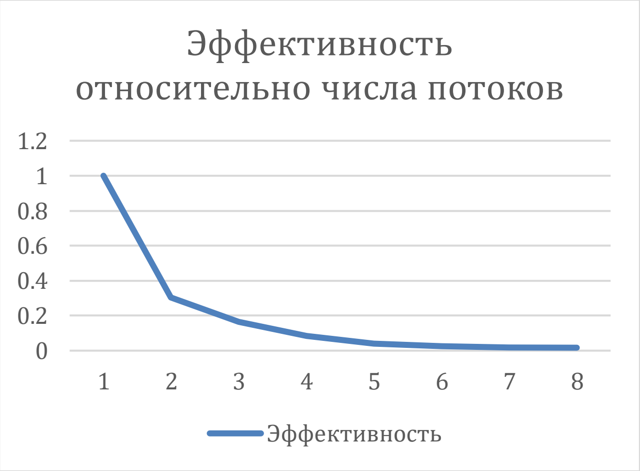
После всех вычислений на экран выводятся корни системы и время выполнения программы. Семафор уничтожается, и программа завершает работу

Ниже приведены данные, показывающие изменения ускорения и эффективности, с разным количеством потоков, для этой реализации.

Число потоков	Время выполнения(мкс)	Ускорение	Эффективность
1	264	1	1
2	435	0,61	0,305
3	531	0,497	0,166
4	799	0,33	0,083
5	1254	0,21	0,042
6	1678	0,157	0,026
7	1908	0,138	0,0197
8	1972	0,134	0,017

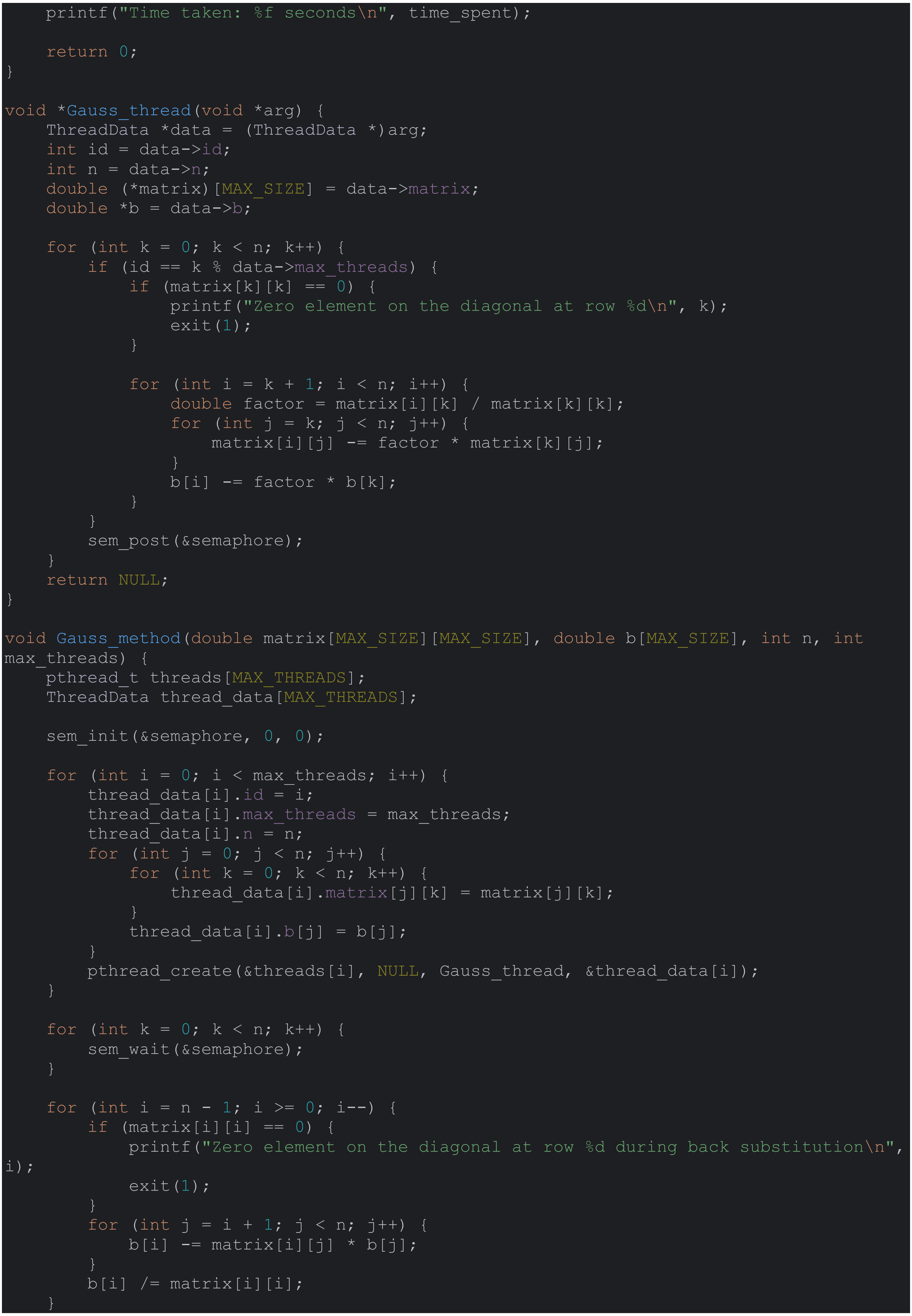
Количество раундов	Время выполнения(мс)
1	279
4	333
9	443
25	885
49	1028
100	1803





Код программы

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <semaphore.h>
#include <time.h>
#define MAX_THREADS 10
#define MAX_SIZE 10

typedef struct {
    int id;
    int max_threads;
    double matrix[MAX_SIZE][MAX_SIZE];
    double b[MAX_SIZE];
    int n;
} ThreadData;

sem_t semaphore;

void *Gauss_thread(void *arg);
void Gauss_method(double matrix[MAX_SIZE][MAX_SIZE], double b[MAX_SIZE], int n, int
max_threads);
void Print_matrix(double matrix[MAX_SIZE][MAX_SIZE], double b[MAX_SIZE], int n);

int main(int argc, char *argv[]) {
    if (argc != 3) {
        printf("Incorrect input\n");
        return 1;
    }

    srand(time(NULL));

    int n = atoi(argv[1]);
    int max_threads = atoi(argv[2]);

    if (n <= 0 || n > MAX_SIZE  || max_threads <= 0 || max_threads > MAX_THREADS) {
        printf("Incorrect matrix size or number of threads.\n");
        return 1;
    }

    double matrix[MAX_SIZE][MAX_SIZE];
    double b[MAX_SIZE];

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            matrix[i][j] = (double)(rand() % 100);
        }
        b[i] = (double)(rand() % 100);
    }

    Print_matrix(matrix, b, n);

    clock_t start_time = clock();

    Gauss_method(matrix, b, n, max_threads);

    clock_t end_time = clock();

    printf("Answers are:\n");
    for (int i = 0; i < n; i++) {
        printf("x%d = %f\n", i, b[i]);
    }

    double time_spent = (double)(end_time - start_time) / CLOCKS_PER_SEC;
```



```
    printf("Time taken: %f seconds\n", time_spent);

    return 0;
}

void *Gauss_thread(void *arg) {
    ThreadData *data = (ThreadData *)arg;
    int id = data->id;
    int n = data->n;
    double (*matrix)[MAX_SIZE] = data->matrix;
    double *b = data->b;

    for (int k = 0; k < n; k++) {
        if (id == k % data->max_threads) {
            if (matrix[k][k] == 0) {
                printf("Zero element on the diagonal at row %d\n", k);
                exit(1);
            }

            for (int i = k + 1; i < n; i++) {
                double factor = matrix[i][k] / matrix[k][k];
                for (int j = k; j < n; j++) {
                    matrix[i][j] -= factor * matrix[k][j];
                }
                b[i] -= factor * b[k];
            }
        }
        sem_post(&semaphore);
    }
    return NULL;
}

void Gauss_method(double matrix[MAX_SIZE][MAX_SIZE], double b[MAX_SIZE], int n, int
max_threads) {
    pthread_t threads[MAX_THREADS];
    ThreadData thread_data[MAX_THREADS];

    sem_init(&semaphore, 0, 0);

    for (int i = 0; i < max_threads; i++) {
        thread_data[i].id = i;
        thread_data[i].max_threads = max_threads;
        thread_data[i].n = n;
        for (int j = 0; j < n; j++) {
            for (int k = 0; k < n; k++) {
                thread_data[i].matrix[j][k] = matrix[j][k];
            }
            thread_data[i].b[j] = b[j];
        }
        pthread_create(&threads[i], NULL, Gauss_thread, &thread_data[i]);
    }

    for (int k = 0; k < n; k++) {
        sem_wait(&semaphore);
    }

    for (int i = n - 1; i >= 0; i--) {
        if (matrix[i][i] == 0) {
            printf("Zero element on the diagonal at row %d during back substitution\n",
i);
            exit(1);
        }
        for (int j = i + 1; j < n; j++) {
            b[i] -= matrix[i][j] * b[j];
        }
        b[i] /= matrix[i][i];
    }
    sem_destroy(&semaphore);
}

void Print_matrix(double matrix[MAX_SIZE][MAX_SIZE], double b[MAX_SIZE], int n) {
    printf("Matrix:\n");
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            printf("%8.2f ", matrix[i][j]);
        }
        printf("| %8.2f\n", b[i]);
    }
    printf("\n");
}

}
```

Протокол работы программы

```
anegamelina@LAPTOP-0ED9K3JN:/mnt/c/Users/Anega/CLionProjects/osi_labs/lab2$ gcc -g -pthread lab2.c -o lab2

anegamelina@LAPTOP-0ED9K3JN:/mnt/c/Users/Anega/CLionProjects/osi_labs/lab2$ time ./lab2 3 1

Matrix:

   86.00    55.00    12.00 |    11.00

   66.00     2.00     3.00 |    93.00

   98.00    69.00    36.00 |    13.00

Answers are:

x0 = -29.314438

x1 = 45.958333

x2 = 0.361111

Time taken: 0.000169 seconds

real    0m0.011s

user    0m0.000s

sys     0m0.004s

anegamelina@LAPTOP-0ED9K3JN:/mnt/c/Users/Anega/CLionProjects/osi_labs/lab2$ time ./lab2 5 10

Matrix:

    8.00    74.00    78.00    34.00    62.00 |    75.00

   25.00    86.00    65.00    98.00    28.00 |     6.00

   89.00    11.00     2.00     3.00    50.00 |    66.00

   75.00    70.00     4.00    48.00    28.00 |    81.00

   99.00    45.00    76.00    14.00    28.00 |     7.00


Answers are:

x0 = -50.173026

x1 = -20.238614

x2 = 24.437500

x3 = 1.541667

x4 = 0.250000
```

```
Time taken: 0.001844 seconds


real    0m0.012s
user    0m0.004s
sys     0m0.001s
anegamelina@LAPTOP-0ED9K3JN:/mnt/c/Users/Anega/CLionProjects/osi_labs/lab2$ time ./lab2 10 3
Matrix:
   3.00   65.00   22.00   91.00    6.00    4.00   86.00   98.00   73.00   59.00 |    5.00
  94.00   71.00   78.00   92.00   23.00   25.00   43.00   76.00   17.00   65.00 |   57.00
  93.00   86.00   93.00   75.00   37.00   35.00   97.00   92.00   68.00    0.00 |   57.00
  42.00   91.00   64.00   98.00   29.00   62.00   72.00   89.00   67.00   18.00 |   60.00
  97.00   10.00   35.00   74.00    6.00   11.00   92.00   71.00   69.00   37.00 |   57.00
  14.00   12.00   94.00    1.00   61.00   39.00   21.00   14.00   96.00   63.00 |    5.00
  12.00   62.00   87.00   74.00   86.00   76.00   93.00    4.00   88.00   90.00 |   67.00
  23.00   17.00   73.00   34.00    9.00   44.00   55.00   46.00   53.00   22.00 |   10.00
  99.00   23.00   71.00   90.00   97.00   85.00   87.00   60.00   43.00   99.00 |   74.00
  30.00   26.00   60.00   58.00   71.00   17.00   98.00   14.00   84.00   21.00 |   31.00

Answers are:
x0 = -36.932168
x1 = -3.957187
x2 = 1.871212
x3 = 2.432851
x4 = -11.610343
x5 = 0.914828
x6 = 0.817272
x7 = 1.444436
x8 = -1.677741
x9 = 1.476190
Time taken: 0.000670 seconds
```

```
real    0m0.012s

user    0m0.004s

sys     0m0.001s

anegamelina@LAPTOP-0ED9K3JN:/mnt/c/Users/Anega/CLionProjects/osi_labs/lab2$ time ./lab2 4 4

Matrix:

   29.00    99.00     7.00    55.00 |    77.00

   65.00    55.00    13.00    87.00 |    48.00

   27.00    50.00     3.00     8.00 |    43.00

    1.00    12.00    97.00     1.00 |    78.00


Answers are:

x0 = 163.422989

x1 = -76.733333

x2 = -193.666667

x3 = 78.000000

Time taken: 0.000825 seconds


real    0m0.011s

user    0m0.000s

sys     0m0.004s

anegamelina@LAPTOP-0ED9K3JN:/mnt/c/Users/Anega/CLionProjects/osi_labs/lab2$ strace -f time
./lab2 3 3

execve("/usr/bin/time", ["time", "./lab2", "3", "3"], 0x7fffba1f81f0 /* 27 vars */) = 0

brk(NULL)                               = 0x5634b197a000

arch_prctl(0x3001 /* ARCH_??? */, 0x7ffd3ff38120) = -1 EINVAL (Invalid argument)

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f833aec0000

access("/etc/ld.so.preload", R_OK)      = -1 ENOENT (No such file or directory)

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

newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=19779, ...}, AT_EMPTY_PATH) = 0
```

```
mmap(NULL, 19779, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f833aebb000

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\302\211\332Pq\2439\235\350\223\322\257\201\326\243\f"..., 68, 896)
= 68

newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2220400, ...}, 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, 2264656, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f833ac92000

mprotect(0x7f833acba000, 2023424, PROT_NONE) = 0

mmap(0x7f833acba000, 1658880, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f833acba000

mmap(0x7f833ae4f000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE,
3, 0x1bd000) = 0x7f833ae4f000

mmap(0x7f833aea8000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x215000) = 0x7f833aea8000

mmap(0x7f833aeae000, 52816, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f833aeae000

close(3)                                = 0

mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f833ac8f000

arch_prctl(ARCH_SET_FS, 0x7f833ac8f740) = 0

set_tid_address(0x7f833ac8fa10)         = 142598

set_robust_list(0x7f833ac8fa20, 24)     = 0

rseq(0x7f833ac900e0, 0x20, 0, 0x53053053) = 0

mprotect(0x7f833aea8000, 16384, PROT_READ) = 0

mprotect(0x5634852c0000, 4096, PROT_READ) = 0

mprotect(0x7f833aefa000, 8192, PROT_READ) = 0

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

```
munmap(0x7f833aebb000, 19779)           = 0

clone(child_stack=NULL,
flags=CLONE_CHILD_CLEARTID|CLONE_CHILD_SETTID|SIGCHLDstrace: Process 142600
attached

, child_tidptr=0x7f833ac8fa10) = 142600

[pid 142600] set_robust_list(0x7f833ac8fa20, 24 <unfinished ...>

[pid 142598] rt_sigaction(SIGINT, {sa_handler=SIG_IGN, sa_mask=[INT],
sa_flags=SA_RESTORER|SA_RESTART, sa_restorer=0x7f833acd4520},  <unfinished ...>

[pid 142600] <... set_robust_list resumed>) = 0

[pid 142598] <... rt_sigaction resumed>{sa_handler=SIG_DFL, sa_mask=[], sa_flags=0}, 8) = 0

[pid 142598] rt_sigaction(SIGQUIT, {sa_handler=SIG_IGN, sa_mask=[QUIT],
sa_flags=SA_RESTORER|SA_RESTART, sa_restorer=0x7f833acd4520},  <unfinished ...>

[pid 142600] execve("./lab2", ["./lab2", "3", "3"], 0x7ffd3ff38310 /* 27 vars */ <unfinished ...>

[pid 142598] <... rt_sigaction resumed>{sa_handler=SIG_DFL, sa_mask=[], sa_flags=0}, 8) = 0

[pid 142598] wait4(-1,  <unfinished ...>

[pid 142600] <... execve resumed>)      = 0

[pid 142600] brk(NULL)                  = 0x556ac6097000

[pid 142600] arch_prctl(0x3001 /* ARCH_??? */, 0x7ffc1d58dee0) = -1 EINVAL (Invalid argument)

[pid 142600] mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f2d50055000

[pid 142600] access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)

[pid 142600] openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

[pid 142600] newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=19779, ...}, AT_EMPTY_PATH) = 0

[pid 142600] mmap(NULL, 19779, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f2d50050000

[pid 142600] close(3)                   = 0

[pid 142600] openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

[pid 142600] 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

[pid 142600] 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

[pid 142600] 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
```

```
[pid 142600] pread64(3,
"\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\302\211\332Pq\2439\235\350\223\322\257\201\326\243\f"..., 68, 896)
= 68

[pid 142600] newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2220400, ...}, AT_EMPTY_PATH) =
0

[pid 142600] 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

[pid 142600] mmap(NULL, 2264656, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f2d4fe27000

[pid 142600] mprotect(0x7f2d4fe4f000, 2023424, PROT_NONE) = 0

[pid 142600] mmap(0x7f2d4fe4f000, 1658880, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f2d4fe4f000

[pid 142600] mmap(0x7f2d4ffe4000, 360448, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) = 0x7f2d4ffe4000

[pid 142600] mmap(0x7f2d5003d000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x215000) = 0x7f2d5003d000

[pid 142600] mmap(0x7f2d50043000, 52816, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f2d50043000

[pid 142600] close(3)                    = 0

[pid 142600] mmap(NULL, 12288, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f2d4fe24000

[pid 142600] arch_prctl(ARCH_SET_FS, 0x7f2d4fe24740) = 0

[pid 142600] set_tid_address(0x7f2d4fe24a10) = 142600

[pid 142600] set_robust_list(0x7f2d4fe24a20, 24) = 0

[pid 142600] rseq(0x7f2d4fe250e0, 0x20, 0, 0x53053053) = 0

[pid 142600] mprotect(0x7f2d5003d000, 16384, PROT_READ) = 0

[pid 142600] mprotect(0x556aa0ea5000, 4096, PROT_READ) = 0

[pid 142600] mprotect(0x7f2d5008f000, 8192, PROT_READ) = 0

[pid 142600] prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0

[pid 142600] munmap(0x7f2d50050000, 19779) = 0

[pid 142600] newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...},
AT_EMPTY_PATH) = 0

[pid 142600] getrandom("\x6d\xe1\xdf\x5b\x40\x42\x31\xd6", 8, GRND_NONBLOCK) = 8
```

```
[pid 142600] brk(NULL)                  = 0x556ac6097000
[pid 142600] brk(0x556ac60b8000)        = 0x556ac60b8000
[pid 142600] write(1, "Matrix:\n", 8Matrix:
)   = 8
[pid 142600] write(1, "   32.00    58.00    28.00 |    "..., 38   32.00    58.00    28.00 |     8.00
) = 38
[pid 142600] write(1, "   82.00    43.00    41.00 |    "..., 38   82.00    43.00    41.00 |    62.00
) = 38
[pid 142600] write(1, "   90.00    43.00    98.00 |    "..., 38   90.00    43.00    98.00 |    74.00
) = 38
[pid 142600] write(1, "\n", 1
)         = 1
[pid 142600] clock_gettime(CLOCK_PROCESS_CPUTIME_ID, {tv_sec=0, tv_nsec=8644300}) = 0
[pid 142600] rt_sigaction(SIGRT_1, {sa_handler=0x7f2d4feb8870, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_RESTART|SA_SIGINFO,
sa_restorer=0x7f2d4fe69520}, NULL, 8) = 0
[pid 142600] rt_sigprocmask(SIG_UNBLOCK, [RTMIN RT_1], NULL, 8) = 0
[pid 142600] mmap(NULL, 8392704, PROT_NONE,
MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) = 0x7f2d4f623000
[pid 142600] mprotect(0x7f2d4f624000, 8388608, PROT_READ|PROT_WRITE) = 0
[pid 142600] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0
[pid 142600]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|
CLONE_SYSVSEM|CLONE_SET

TLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f2d4fe23910,
parent_tid=0x7f2d4fe23910, exit_signal=0, stack=0x7f2d4f623000, stack_size=0x7fff00,
tls=0x7f2d4fe23640}strace: Process 142601 attached
 <unfinished ...>
[pid 142601] rseq(0x7f2d4fe23fe0, 0x20, 0, 0x53053053 <unfinished ...>
[pid 142600] <... clone3 resumed> => {parent_tid=[142601]}, 88) = 142601
[pid 142601] <... rseq resumed>)        = 0
[pid 142600] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 142601] set_robust_list(0x7f2d4fe23920, 24 <unfinished ...>

[pid 142600] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142601] <... set_robust_list resumed>) = 0

[pid 142600] mmap(NULL, 8392704, PROT_NONE,
MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0 <unfinished ...>

[pid 142601] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>

[pid 142600] <... mmap resumed>)        = 0x7f2d4ee22000

[pid 142601] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142600] mprotect(0x7f2d4ee23000, 8388608, PROT_READ|PROT_WRITE <unfinished ...>

[pid 142601] rt_sigprocmask(SIG_BLOCK, ~[RT_1],  <unfinished ...>

[pid 142600] <... mprotect resumed>)    = 0

[pid 142601] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142600] rt_sigprocmask(SIG_BLOCK, ~[],  <unfinished ...>

[pid 142601] madvise(0x7f2d4f623000, 8368128, MADV_DONTNEED <unfinished ...>

[pid 142600] <... rt_sigprocmask resumed>[], 8) = 0

[pid 142601] <... madvise resumed>)     = 0

[pid 142600]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|
CLONE_SYSVSEM|CLONE_SET

TLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f2d4f622910,
parent_tid=0x7f2d4f622910, exit_signal=0, stack=0x7f2d4ee22000, stack_size=0x7fff00,
tls=0x7f2d4f622640} <unfinished ...>

[pid 142601] exit(0strace: Process 142602 attached

)                   = ?

[pid 142600] <... clone3 resumed> => {parent_tid=[142602]}, 88) = 142602

[pid 142602] rseq(0x7f2d4f622fe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 142601] +++ exited with 0 +++

[pid 142602] <... rseq resumed>)        = 0

[pid 142600] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>

[pid 142602] set_robust_list(0x7f2d4f622920, 24 <unfinished ...>

[pid 142600] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142602] <... set_robust_list resumed>) = 0
```

```
[pid 142600] mmap(NULL, 8392704, PROT_NONE,
MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0 <unfinished ...>

[pid 142602] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>

[pid 142600] <... mmap resumed>)        = 0x7f2d4e621000

[pid 142602] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142600] mprotect(0x7f2d4e622000, 8388608, PROT_READ|PROT_WRITE <unfinished ...>

[pid 142602] rt_sigprocmask(SIG_BLOCK, ~[RT_1],  <unfinished ...>

[pid 142600] <... mprotect resumed>)    = 0

[pid 142602] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142600] rt_sigprocmask(SIG_BLOCK, ~[],  <unfinished ...>

[pid 142602] madvise(0x7f2d4ee22000, 8368128, MADV_DONTNEED <unfinished ...>

[pid 142600] <... rt_sigprocmask resumed>[], 8) = 0

[pid 142602] <... madvise resumed>)     = 0

[pid 142600]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|
CLONE_SYSVSEM|CLONE_SET

TLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7f2d4ee21910,
parent_tid=0x7f2d4ee21910, exit_signal=0, stack=0x7f2d4e621000, stack_size=0x7fff00,
tls=0x7f2d4ee21640} <unfinished ...>

[pid 142602] exit(0strace: Process 142603 attached

)                   = ?

[pid 142600] <... clone3 resumed> => {parent_tid=[142603]}, 88) = 142603

[pid 142603] rseq(0x7f2d4ee21fe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 142602] +++ exited with 0 +++

[pid 142603] <... rseq resumed>)        = 0

[pid 142600] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>

[pid 142603] set_robust_list(0x7f2d4ee21920, 24 <unfinished ...>

[pid 142600] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142603] <... set_robust_list resumed>) = 0

[pid 142600] clock_gettime(CLOCK_PROCESS_CPUTIME_ID,  <unfinished ...>

[pid 142603] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>

[pid 142600] <... clock_gettime resumed>{tv_sec=0, tv_nsec=12007700}) = 0
```

```
[pid 142603] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142600] write(1, "Answers are:\n", 13 <unfinished ...>

Answers are:

[pid 142603] rt_sigprocmask(SIG_BLOCK, ~[RT_1],  <unfinished ...>

[pid 142600] <... write resumed>)       = 13

[pid 142603] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 142600] write(1, "x0 = -1.719121\n", 15 <unfinished ...>

x0 = -1.719121

[pid 142603] madvise(0x7f2d4e621000, 8368128, MADV_DONTNEED <unfinished ...>

[pid 142600] <... write resumed>)       = 15

[pid 142603] <... madvise resumed>)     = 0

[pid 142600] write(1, "x1 = 0.721879\n", 14 <unfinished ...>

[pid 142603] exit(0x1 = 0.721879

 <unfinished ...>

[pid 142600] <... write resumed>)       = 14

[pid 142603] <... exit resumed>)        = ?

[pid 142600] write(1, "x2 = 0.755102\n", 14 <unfinished ...>

[pid 142603] +++ exited with 0 +++

x2 = 0.755102

[pid 142600] <... write resumed>)       = 14

[pid 142600] write(1, "Time taken: 0.003363 seconds\n", 29Time taken: 0.003363 seconds

) = 29

[pid 142600] exit_group(0)              = ?

[pid 142600] +++ exited with 0 +++

<... wait4 resumed>[{WIFEXITED(s) && WEXITSTATUS(s) == 0}], 0, {ru_utime={tv_sec=0,
tv_usec=0}, ru_stime={tv_sec=0, tv_usec=15637}, ...}) = 142600

--- SIGCHLD {si_signo=SIGCHLD, si_code=CLD_EXITED, si_pid=142600, si_uid=1000, si_status=0,
si_utime=0, si_stime=3} ---

rt_sigaction(SIGINT, {sa_handler=SIG_DFL, sa_mask=[INT],
sa_flags=SA_RESTORER|SA_RESTART, sa_restorer=0x7f8
```

```
33acd4520}, {sa_handler=SIG_IGN, sa_mask=[INT], sa_flags=SA_RESTORER|SA_RESTART,
sa_restorer=0x7f833acd4520}, 8) = 0

rt_sigaction(SIGQUIT, {sa_handler=SIG_DFL, sa_mask=[QUIT],
sa_flags=SA_RESTORER|SA_RESTART, sa_restorer=0x7

f833acd4520}, {sa_handler=SIG_IGN, sa_mask=[QUIT], sa_flags=SA_RESTORER|SA_RESTART,
sa_restorer=0x7f833acd4520}, 8) = 0

write(2, "0.00", 40.00)                  = 4

write(2, "u", 1u)                        = 1

write(2, "s", 1s)                        = 1

write(2, "e", 1e)                        = 1

write(2, "r", 1r)                        = 1

write(2, " ", 1 )                        = 1

write(2, "0.01", 40.01)                  = 4

write(2, "s", 1s)                        = 1

write(2, "y", 1y)                        = 1

write(2, "s", 1s)                        = 1

write(2, "t", 1t)                        = 1

write(2, "e", 1e)                        = 1

write(2, "m", 1m)                        = 1

write(2, " ", 1 )                        = 1

write(2, "0:00.05", 70:00.05)            = 7

write(2, "e", 1e)                        = 1

write(2, "l", 1l)                        = 1

write(2, "a", 1a)                        = 1

write(2, "p", 1p)                        = 1

write(2, "s", 1s)                        = 1

write(2, "e", 1e)                        = 1

write(2, "d", 1d)                        = 1

write(2, " ", 1 )                        = 1

write(2, "25%", 325%)                    = 3

write(2, "C", 1C)                        = 1
write(2, "P", 1P)                    = 1
write(2, "U", 1U)                     = 1
write(2, " ", 1 )                   = 1
write(2, "(", 1()                   = 1
write(2, "0", 10)                    = 1
write(2, "a", 1a)                    = 1
write(2, "v", 1v)                    = 1
write(2, "g", 1g)                    = 1
write(2, "t", 1t)                   = 1
write(2, "e", 1e)                    = 1
write(2, "x", 1x)                    = 1
write(2, "t", 1t)                   = 1
write(2, "+", 1+)                    = 1
write(2, "0", 10)                    = 1
write(2, "a", 1a)                    = 1
write(2, "v", 1v)                    = 1
write(2, "g", 1g)                    = 1
write(2, "d", 1d)                    = 1
write(2, "a", 1a)                    = 1
write(2, "t", 1t)                   = 1
write(2, "a", 1a)                    = 1
write(2, " ", 1 )                   = 1
write(2, "1876", 41876)                    = 4
write(2, "m", 1m)                     = 1
write(2, "a", 1a)                    = 1
write(2, "x", 1x)                    = 1
write(2, "r", 1r)                   = 1
write(2, "e", 1e)                    = 1
write(2, "s", 1s)                    = 1
write(2, "i", 1i)                   = 1
write(2, "d", 1d)                  = 1
write(2, "e", 1e)                  = 1
write(2, "n", 1n)                  = 1
write(2, "t", 1t)                  = 1
write(2, ")", 1))                  = 1
write(2, "k", 1k)                  = 1
write(2, "\n", 1
)                       = 1
write(2, "48", 248)                = 2
write(2, "i", 1i)                  = 1
write(2, "n", 1n)                  = 1
write(2, "p", 1p)                  = 1
write(2, "u", 1u)                  = 1
write(2, "t", 1t)                  = 1
write(2, "s", 1s)                  = 1
write(2, "+", 1+)                  = 1
write(2, "0", 10)                  = 1
write(2, "o", 1o)                  = 1
write(2, "u", 1u)                  = 1
write(2, "t", 1t)                  = 1
write(2, "p", 1p)                  = 1
write(2, "u", 1u)                  = 1
write(2, "t", 1t)                  = 1
write(2, "s", 1s)                  = 1
write(2, " ", 1 )                  = 1
write(2, "(", 1()                  = 1
write(2, "1", 11)                  = 1
write(2, "m", 1m)                  = 1
write(2, "a", 1a)                  = 1
write(2, "j", 1j)                  = 1
write(2, "o", 1o)                    = 1

write(2, "r", 1r)                    = 1

write(2, "+", 1+)                    = 1

write(2, "80", 280)                  = 2

write(2, "m", 1m)                    = 1

write(2, "i", 1i)                    = 1

write(2, "n", 1n)                    = 1

write(2, "o", 1o)                    = 1

write(2, "r", 1r)                    = 1

write(2, ")", 1))                    = 1

write(2, "p", 1p)                    = 1

write(2, "a", 1a)                    = 1

write(2, "g", 1g)                    = 1

write(2, "e", 1e)                    = 1

write(2, "f", 1f)                    = 1

write(2, "a", 1a)                    = 1

write(2, "u", 1u)                    = 1

write(2, "l", 1l)                    = 1

write(2, "t", 1t)                    = 1

write(2, "s", 1s)                    = 1

write(2, " ", 1 )                    = 1

write(2, "0", 10)                    = 1

write(2, "s", 1s)                    = 1

write(2, "w", 1w)                    = 1

write(2, "a", 1a)                    = 1

write(2, "p", 1p)                    = 1

write(2, "s", 1s)                    = 1

write(2, "\n", 1

)                  = 1

exit_group(0)                        = ?
```

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

Вывод

В ходе написания данной лабораторной работы я научилась создавать программы, работающие с несколькими потоками, а также синхронизировать их между собой. В результате тестирования программы, я проанализировала каким образом количество потоков влияет на эффективность и ускорение работы программы. Лабораторная работа была довольно интересна, так как я впервые работала с многопоточностью и синхронизацией на СИ.