

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-209БВ-24

Студент: Хамзин Т.Н.

Преподаватель: Миронов Е.С.

Оценка: _____

Дата: 01.12.24

Москва, 2024

Постановка задачи

Вариант 6.

Составить программу на языке Си, производящую перемножение 2-ух матриц, содержащих комплексные числа, в многопоточном режиме. При обработке использовать стандартные средства создания потоков операционной системы (Windows/Unix).

Общий метод и алгоритм решения

Использованные системные вызовы:

- `int pthread_create(pthread_t *thread, const pthread_attr_t *attr, void *(*start_routine)(void*), void *arg);` - Создает новый поток. Поток начинает выполнение функции `start_routine` с аргументом `arg`.
- `int pthread_join(pthread_t thread, void **retval);` - Блокирует вызывающий поток до завершения указанного потока `thread`. Получает значение, возвращенное потоком.

Я заполнил массивы, содержащие информацию о перемножаемых матрицах случайными числами и создал n -ое количество потоков, каждому из которых передал для подсчета M/n строк результирующей матрицы, в случае, если деления происходило нецелочисленно, последний (n -ый) процесс забирал оставшиеся строки себе

Код программы

Main.c

```
#include <pthread.h>

#include <stdio.h>

#include <unistd.h>

#include <stdlib.h>

#include <time.h>

#include <sys/time.h>

// #define ROWS1 2

// #define COLS1 3

// #define ROWS2 3

// #define COLS2 2

#define ROWS1 400

#define COLS1 400

#define ROWS2 400

#define COLS2 400
```

```
#define SIZE 400
```

```
int MAX_THREADS;
```

```
typedef struct{  
    double Re;  
    double Im;  
}complex_matrix;
```

```
typedef struct{  
    int index;  
    complex_matrix (*result)[COLS2];  
    complex_matrix (*a)[COLS1];  
    complex_matrix (*b)[COLS2];  
}args_for_thread;//вынес структуры для глобального значения
```

```
void* multiply_matrixes(void* arg){
```

```
    args_for_thread* args = (args_for_thread*)arg;  
    int thread_index = args->index;  
    complex_matrix (*result)[COLS2] = args->result;  
    complex_matrix (*matrix_a)[COLS1] = args->a;  
    complex_matrix (*matrix_b)[COLS2] = args->b;
```

```
int per_thread=ROWS2/MAX_THREADS;
```

```
if (thread_index!=MAX_THREADS-1){  
    for (int i=per_thread*thread_index;i<per_thread*(thread_index+1);i++){  
        for (int j=0;j<COLS2;j++){  
            double real_sum = 0.0;  
            double imag_sum = 0.0;
```

```

    for (int k=0;k<COLS1;k++){

        double a_real = matrix_a[i][ k].Re;

        double a_imag = matrix_a[i][k].Im;

        double b_real = matrix_b[k][j].Re;

        double b_imag = matrix_b[k][j].Im;

        real_sum += a_real * b_real - a_imag * b_imag;

        imag_sum += a_real * b_imag + a_imag * b_real;

    }

    result[i][j].Re=real_sum;

    result[i][j].Im=imag_sum;

}

}

}else{

    for (int i=per_thread*thread_index;i<ROWS1;i++){

        for (int j=0;j<COLS2;j++){

            double real_sum = 0.0;

            double imag_sum = 0.0;

            for (int k=0;k<COLS1;k++){

                double a_real = matrix_a[i][k].Re;

                double a_imag = matrix_a[i][k].Im;

                double b_real = matrix_b[k][j].Re;

                double b_imag = matrix_b[k][j].Im;

                real_sum += a_real * b_real - a_imag * b_imag;

                imag_sum += a_real * b_imag + a_imag * b_real;

            }

            result[i][j].Re=real_sum;

            result[i][j].Im=imag_sum;

        }

    }

}

```

```

    }
}

void random_matrix(complex_matrix (*matrix)[SIZE][SIZE]) {

    for (int i = 0; i < SIZE; i++) {

        for (int j = 0; j < SIZE; j++) {

            (*matrix)[i][j].Re = rand() % 20;

            (*matrix)[i][j].Im = rand() % 20; // значения от 0 до 19

        }

    }

}

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

    // clock_t start, end;

    // start=clock();

    struct timeval start, end;

    gettimeofday(&start, NULL);

    if (argc != 2){

        printf("False format\n");

    }

    MAX_THREADS=atoi(argv[1]);

    pthread_t threads[MAX_THREADS];

    // complex_matrix complex_matrix1[ROWS1][COLS1]={{{3,1},{1,1},{-3,1}},{{2,0},{-1,1},{-1,-1}}};

    // complex_matrix complex_matrix2[ROWS2][COLS2]={{{1,3},{2,1}},{{2,0},{1,3}},{{0,1},{-1,2}}};

    complex_matrix complex_matrix1[ROWS1][COLS1]={0};

    complex_matrix complex_matrix2[ROWS2][COLS2]={0};

    random_matrix(&complex_matrix1);

    random_matrix(&complex_matrix2);

    complex_matrix result_matrix[ROWS1][COLS2]={0};

```

```

args_for_thread args[MAX_THREADS];

for (int i=0;i<MAX_THREADS;i++){
    args[i].index=i;
    args[i].result=result_matrix;
    args[i].a=complex_matrix1;
    args[i].b=complex_matrix2;
    pthread_create(&threads[i],NULL,multiply_matrixes,&args[i]);
}
for (int i=0;i<MAX_THREADS;i++){
    pthread_join(threads[i],NULL);
}
// end=clock();
// for (int i=0;i<ROWS1;i++){
//     for (int j=0;j<COLS2;j++){
//         if (result_matrix[i][j].Im>0){
//             printf("%d+%df ",result_matrix[i][j].Re,result_matrix[i][j].Im);
//         }else if (result_matrix[i][j].Im==0){
//             printf("%d ",result_matrix[i][j].Re);
//         }else{
//             printf("%d%ff ",result_matrix[i][j].Re,result_matrix[i][j].Im);
//         }
//     }
//     printf("\n");
// }
gettimeofday(&end, NULL); // Конец измерения

double delta = (end.tv_sec - start.tv_sec) + (end.tv_usec - start.tv_usec) / 1000000.0;
// double delta = ((double) (end - start)) / CLOCKS_PER_SEC;

```

```
printf("Время выполнения программы с %d потоками: %f секунд\n",MAX_THREADS,delta);

return 0;

}
```

Протокол работы программы

Тестирование:

\$./main 1

1.000000+9.000000i 4.000000+2.000000i

1.000000+7.000000i 3.000000-1.000000i

Время выполнения программы с 1 потоками: 0.000170 секунд

\$./main 2

1.000000+9.000000i 4.000000+2.000000i

1.000000+7.000000i 3.000000-1.000000i

Время выполнения программы с 2 потоками: 0.000238 секунд

Таблица составлена по результатам работы программы с матрицами 400×400 чисел

Число потоков	Время исполнения (мс)	Ускорение	Эффективность
1	301.494	1	1
2	151.597	1.99	0.995
3	108.120	2.78	0.93
4	113.618	2.65	0.66
5	71.916	4.19	0.84
6	69.849	4.32	0.72

Strace:

strace -f ./main 4

```
execve("./main", [ "./main", "4"], 0x7ffff4ae8ec0 /* 31 vars */) = 0
brk(NULL)                                = 0x18b14000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x72a1ac6de000
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=33091, ...}, AT_EMPTY_PATH) = 0
mmap(NULL, 33091, PROT_READ, MAP_PRIVATE, 3, 0) = 0x72a1ac6d5000
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\3\0\>\0\1\0\0\0\20t\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
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=1926232, ...}, 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, 1974096, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x72a1ac4f3000
mmap(0x72a1ac519000, 1400832, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x26000) = 0x72a1ac519000
mmap(0x72a1ac66f000, 339968, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x17c000) = 0x72a1ac66f000
mmap(0x72a1ac6c2000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1cf000) = 0x72a1ac6c2000
mmap(0x72a1ac6c8000, 53072, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x72a1ac6c8000
close(3)                                 = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x72a1ac4f0000
arch_prctl(ARCH_SET_FS, 0x72a1ac4f0740) = 0
set_tid_address(0x72a1ac4f0a10)          = 4951
set_robust_list(0x72a1ac4f0a20, 24)      = 0
rseq(0x72a1ac4f1060, 0x20, 0, 0x53053053) = 0
mprotect(0x72a1ac6c2000, 16384, PROT_READ) = 0
```

```

mprotect(0x403000, 4096, PROT_READ)      = 0

mprotect(0x72a1ac711000, 8192, PROT_READ) = 0

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

munmap(0x72a1ac6d5000, 33091)             = 0

rt_sigaction(SIGRT_1, {sa_handler=0x72a1ac579720, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_RESTART|SA_SIGINFO, sa_restorer=0x72a1ac52f050}, 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) = 0x72a1abcef000

mprotect(0x72a1abcf0000, 8388608, PROT_READ|PROT_WRITE) = 0

getrandom("\xcd\x61\xb4\x5f\x21\xbd\xaf\xb6", 8, GRND_NONBLOCK) = 8

brk(NULL)                                = 0x18b14000

brk(0x18b35000)                          = 0x18b35000

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

clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_S
ETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x72a1ac4ef990,
parent_tid=0x72a1ac4ef990, exit_signal=0, stack=0x72a1abcef000, stack_size=0x7fff80,
tls=0x72a1ac4ef6c0}, 88) = -1 ENOSYS (Function not implemented)

clone(child_stack=0x72a1ac4eef70,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDtrace: Process 4952 attached
, parent_tid=[4952], tls=0x72a1ac4ef6c0, child_tidptr=0x72a1ac4ef990) = 4952

[pid 4951] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>

[pid 4952] rseq(0x72a1ac4effe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 4951] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 4952] <... rseq resumed>              = 0

[pid 4951] mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0
<unfinished ...>

[pid 4952] set_robust_list(0x72a1ac4ef9a0, 24 <unfinished ...>

[pid 4951] <... mmap resumed>              = 0x72a1ab4ee000

[pid 4952] <... set_robust_list resumed>) = 0

[pid 4951] mprotect(0x72a1ab4ef000, 8388608, PROT_READ|PROT_WRITE <unfinished ...>

[pid 4952] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>

[pid 4951] <... mprotect resumed>          = 0

[pid 4952] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 4951] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0

```

```

[pid 4951]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_S
ETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x72a1abcee990,
parent_tid=0x72a1abcee990, exit_signal=0, stack=0x72a1ab4ee000, stack_size=0x7fff80,
tls=0x72a1abcee6c0}, 88) = -1 ENOSYS (Function not implemented)

[pid 4951] clone(child_stack=0x72a1abcedf70,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTID|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 4953 attached

, parent_tid=[4953], tls=0x72a1abcee6c0, child_tidptr=0x72a1abcee990) = 4953

[pid 4953] rseq(0x72a1abceefe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 4951] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>

[pid 4953] <... rseq resumed>          = 0

[pid 4951] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 4953] set_robust_list(0x72a1abcee9a0, 24 <unfinished ...>

[pid 4951] mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0
<unfinished ...>

[pid 4953] <... set_robust_list resumed>) = 0

[pid 4951] <... mmap resumed>          = 0x72a1aaced000

[pid 4953] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>

[pid 4951] mprotect(0x72a1aacee000, 8388608, PROT_READ|PROT_WRITE <unfinished ...>

[pid 4953] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 4951] <... mprotect resumed>      = 0

[pid 4951] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0

[pid 4951]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_S
ETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x72a1ab4ed990,
parent_tid=0x72a1ab4ed990, exit_signal=0, stack=0x72a1aaced000, stack_size=0x7fff80,
tls=0x72a1ab4ed6c0}, 88) = -1 ENOSYS (Function not implemented)

[pid 4951] clone(child_stack=0x72a1ab4ecf70,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTID|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 4954 attached

, parent_tid=[4954], tls=0x72a1ab4ed6c0, child_tidptr=0x72a1ab4ed990) = 4954

[pid 4954] rseq(0x72a1ab4edfe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 4951] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>

[pid 4954] <... rseq resumed>          = 0

[pid 4951] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 4954] set_robust_list(0x72a1ab4ed9a0, 24 <unfinished ...>

[pid 4951] mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0
<unfinished ...>

```

```

[pid 4954] <... set_robust_list resumed>) = 0
[pid 4951] <... mmap resumed>          = 0x72a1aa4ec000
[pid 4954] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>
[pid 4951] mprotect(0x72a1aa4ed000, 8388608, PROT_READ|PROT_WRITE <unfinished ...>
[pid 4954] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 4951] <... mprotect resumed>      = 0
[pid 4951] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0
[pid 4951]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_S
ETTLS|CLONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x72a1aac990,
parent_tid=0x72a1aac990, exit_signal=0, stack=0x72a1aa4ec000, stack_size=0x7fff80,
tls=0x72a1aac6c0}, 88) = -1 ENOSYS (Function not implemented)
[pid 4951] clone(child_stack=0x72a1aacebf70,
flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTIDstrace: Process 4955 attached
, parent_tid=[4955], tls=0x72a1aac6c0, child_tidptr=0x72a1aac990) = 4955
[pid 4951] rt_sigprocmask(SIG_SETMASK, [], <unfinished ...>
[pid 4955] rseq(0x72a1aac6cfe0, 0x20, 0, 0x53053053 <unfinished ...>
[pid 4951] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 4955] <... rseq resumed>          = 0
[pid 4951] futex(0x72a1ac4ef990, FUTEX_WAIT_BITSET|FUTEX_CLOCK_REALTIME, 4952, NULL,
FUTEX_BITSET_MATCH_ANY <unfinished ...>
[pid 4955] set_robust_list(0x72a1aac99a0, 24) = 0
[pid 4955] rt_sigprocmask(SIG_SETMASK, [], NULL, 8) = 0
[pid 4953] rt_sigprocmask(SIG_BLOCK, ~[RT_1], NULL, 8) = 0
[pid 4953] madvise(0x72a1ab4ee000, 8368128, MADV_DONTNEED) = 0
[pid 4953] exit(0)                      = ?
[pid 4953] +++ exited with 0 +++
[pid 4955] rt_sigprocmask(SIG_BLOCK, ~[RT_1], NULL, 8) = 0
[pid 4955] madvise(0x72a1aa4ec000, 8368128, MADV_DONTNEED) = 0
[pid 4955] exit(0)                      = ?
[pid 4955] +++ exited with 0 +++
[pid 4954] rt_sigprocmask(SIG_BLOCK, ~[RT_1], NULL, 8) = 0
[pid 4954] madvise(0x72a1aaced000, 8368128, MADV_DONTNEED) = 0
[pid 4954] exit(0)                      = ?
[pid 4952] rt_sigprocmask(SIG_BLOCK, ~[RT_1], <unfinished ...>

```

[pid 4954] +++ exited with 0 +++

[pid 4952] <... rt_sigprocmask resumed>NULL, 8) = 0

[pid 4952] madvise(0x72a1abcef000, 8368128, MADV_DONTNEED) = 0

[pid 4952] exit(0) = ?

[pid 4951] <... futex resumed>) = 0

[pid 4952] +++ exited with 0 +++

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

write(1, "\320\222\321\200\320\265\320\274\321\217
\320\262\321\213\320\277\320\276\320\273\320\275\320\265\320\275\320\270\321\217 "...,
95Время выполнения программы с 4 потоками: 0.107280 секунд

) = 95

exit_group(0) = ?

+++ exited with 0 +++

Вывод

В ходе выполнения лабораторной работы были получены практические навыки создания и управления потоками в ОС Linux, обеспечена корректная синхронизация между ними. Результаты исследования подтвердили теоретические положения о параллельных вычислениях - достигнуто существенное ускорение при сохранении высокой эффективности использования потоков. Полученный опыт может быть применен при разработке высокопроизводительных приложений, требующих параллельной обработки данных.