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: Леоненкова Е.А.

Преподаватель: Бахарев В.Д.

Оценка: _____

Дата: 03.12.24

Москва, 2024

Постановка задачи

Цель работы:

Целью является приобретение практических навыков в:

- **Управление потоками в ОС**
- **Обеспечение 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. Так же 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 1. Отсортировать массив целых чисел при помощи битонической сортировки

Общий метод и алгоритм решения

Использованные системные вызовы:

- int pthread_create(pthread_t *__restrict__newthread, const pthread_attr_t *__restrict_____attr, void *(*__start_routine)(void *), void *__restrict__arg) — создаёт поток с рутиной (стартовой функцией) и заданными аргументами
- int pthread_join(pthread_t __th, void **_thread_return) — дожидается завершения потока.

Для mutex реализации были использованы:
pthread_mutex_t – тип данных;
int pthread_mutex_init(pthread_mutex_t *mutex, const pthread_mutexattr_t *mutexattr) – инициализация мьютекса;
int pthread_mutex_lock(pthread_mutex_t *mutex) – блокировка мьютекса;
int pthread_mutex_unlock(pthread_mutex_t *mutex) – разблокировка мьютекса;
int pthread_mutex_destroy(pthread_mutex_t *mutex) – уничтожение мьютекса;

Программа получает на вход два аргумента – размер массива, который должен быть степенью двойки, и максимально количество потоков для сортировки. Проверяются входные данные на корректность. Для полученного размера массива, происходит его заполнение случайными значениями.

Мьютекс используется для синхронизации доступа к глобальной переменной thread_count, которая отслеживает общее количество созданных потоков. Реализуем битоническую сортировку с использованием многопоточности.

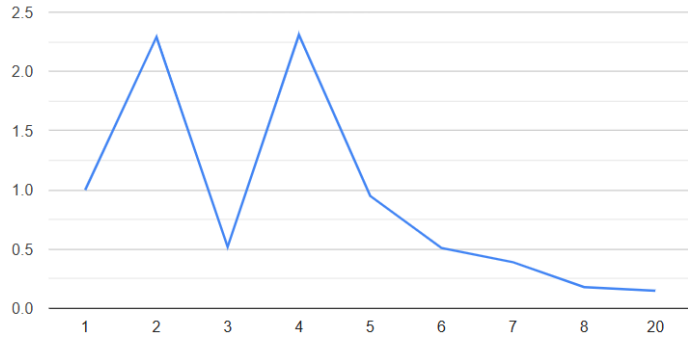
Битоническая сортировка – это алгоритм сортировки, суть которого заключается в разделении массива пополам и сортировке отдельных частей по убыванию и возрастанию. Рассчитываем максимальную глубину рекурсии, на основе которой определяем кол-во созданных потоков. Для сортировки массива используем рекурсивный подход: если текущая глубина рекурсии меньше максимальной, создаются два потока для выполнения сортировки, иначе сортировка в текущем потоке. Затем объединяем две отсортированные половинки.

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

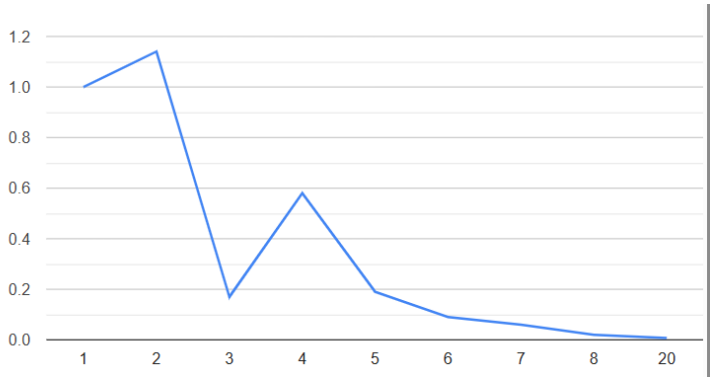
Число потоков	Время выполнения	Ускорение	Эффективность
1	2.248	1,00	1.00
2	0.981	2,29	1.14
3	4.365	0.52	0.17
4	0.974	2.31	0.58
5	2.355	0.95	0.19
6	4.383	0.51	0.09
7	5.699	0.39	0.06
8	12.681	0.18	0.02
20	15.331	0.15	0.007

Размер массива	Время выполнения(мс)
2	1.214
4	0.854
8	1.058
32	0.328
64	0.299
1024	0.322
4096	6.621
8192	8.540

Ускорение:



Эффективность:



Код программы

```c
#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>
#include <time.h>

int thread_count = 0;

pthread_mutex_t count_mutex;

typedef struct {
    int *arr;
    int low;
    int count;
    int dir;
    int depth;
    int max_depth;
} thread_data_t;

void compare_and_swap(int *arr, int i, int j, int dir) {
    if ((dir == 1 && arr[i] > arr[j]) || (dir == 0 && arr[i] < arr[j])) {
        int temp = arr[i];
        arr[i] = arr[j];
        arr[j] = temp;
    }
}

void bitonic_merge(int *arr, int low, int count, int dir) {
    if (count > 1) {
        int k = count / 2;
        for (int i = low; i < low + k; i++) {
            compare_and_swap(arr, i, i + k, dir);
        }
        bitonic_merge(arr, low, k, dir);
        bitonic_merge(arr, low + k, k, dir);
    }
}
```

```c
void *bitonic_sort(void *arg) {
    thread_data_t *data = (thread_data_t *)arg;
    int *arr = data->arr;
    int low = data->low;
    int count = data->count;
    int dir = data->dir;
    int depth = data->depth;
    int max_depth = data->max_depth;

    if (count > 1) {
        int k = count / 2;

        if (depth < max_depth) {
            pthread_t thread1, thread2;

            thread_data_t *left_data = malloc(sizeof(thread_data_t));
            *left_data = (thread_data_t){arr, low, k, 1, depth + 1, max_depth};

            thread_data_t *right_data = malloc(sizeof(thread_data_t));
            *right_data = (thread_data_t){arr, low + k, k, 0, depth + 1, max_depth};

            pthread_mutex_lock(&count_mutex);
            thread_count += 2;
            pthread_mutex_unlock(&count_mutex);

            pthread_create(&thread1, NULL, bitonic_sort, left_data);
            pthread_create(&thread2, NULL, bitonic_sort, right_data);

            pthread_join(thread1, NULL);
            pthread_join(thread2, NULL);

            free(left_data);
            free(right_data);
        } else {
            thread_data_t left_data = {arr, low, k, 1, depth + 1, max_depth};
            thread_data_t right_data = {arr, low + k, k, 0, depth + 1, max_depth};

            bitonic_sort(&left_data);
```

```
            bitonic_sort(&right_data);
        }

        bitonic_merge(arr, low, count, dir);
    }

    return NULL;
}
int compute_max_depth(int num_threads) {
    int max_depth = 0;
    int total_threads = 0;
    while (1) {
        total_threads = (1 << (max_depth + 1)) - 1;
        if (total_threads > num_threads) {
            return max_depth - 1;
        }
        max_depth++;
    }
}

void sort(int *arr, int n, int num_threads) {
    pthread_t thread;

    int max_depth = compute_max_depth(num_threads);

    pthread_mutex_lock(&count_mutex);
    thread_count++;
    pthread_mutex_unlock(&count_mutex);

    thread_data_t *data = malloc(sizeof(thread_data_t));
    *data = (thread_data_t){arr, 0, n, 1, 0, max_depth};

    pthread_create(&thread, NULL, bitonic_sort, data);
    pthread_join(thread, NULL);

    free(data);
}

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

```c
    if (argc != 3) {
        printf("Usage: %s <array_size> <num_threads>\n", argv[0]);
        return 1;
    }

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

    if (n & (n - 1)) {
        printf("The array size is set incorrectly, it must be a power of 2\n");
        return 1;
    }

    int *arr = (int *)malloc(n * sizeof(int));

    for (int i = 0; i < n; i++) {
        arr[i] = rand() % 100;
    }

    pthread_mutex_init(&count_mutex, NULL);

    clock_t start_time = clock();

    sort(arr, n, num_threads);

    clock_t end_time = clock();
    double time_spent = (double)(end_time - start_time) / CLOCKS_PER_SEC;

    printf("The value of the requested threads: %d\n", num_threads);
    printf("The total number of threads created: %d\n", thread_count);
    printf("Elapsed time: %f seconds\n", time_spent);

    free(arr);

    pthread_mutex_destroy(&count_mutex);

    return 0;
}
```

Протокол работы программы

```
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 1024 8
Number of threads requested: 8
Total threads created: 7
Time taken: 0.002196 seconds
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 1024 20
Number of threads requested: 20
Total threads created: 15
Time taken: 0.013289 seconds
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 1024 7
Number of threads requested: 7
Total threads created: 7
Time taken: 0.005926 seconds
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 1024 1
Number of threads requested: 1
Total threads created: 1
Time taken: 0.001070 seconds
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 16 5
Number of threads requested: 5
Total threads created: 3
Time taken: 0.000874 seconds
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 16 3
Number of threads requested: 3
Total threads created: 3
Time taken: 0.001546 seconds
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ ./main 20 4
Array size must be a power of 2
leoelena@DESKTOP-HJEL67G:/mnt/c/Users/Елена/Desktop/OS/OS/LAB_2/src$ strace -f ./main 1024 4
execve("./main", ["./main", "1024", "4"], 0x7ffd83a382b8 /* 36 vars */) = 0
brk(NULL)                               = 0x5580da4d4000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97868d000
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
fstat(3, {st_mode=S_IFREG|0644, st_size=20115, ...}) = 0
mmap(NULL, 20115, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fa978688000
close(3)                                = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libtsan.so.2", 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
fstat(3, {st_mode=S_IFREG|0644, st_size=8980352, ...}) = 0
mmap(NULL, 17393288, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fa9775f1000
mmap(0x7fa97762b000, 856064, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3a000) =
0x7fa97762b000
mmap(0x7fa9776fc000, 212992, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x10b000) =
0x7fa9776fc000
mmap(0x7fa977730000, 49152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x13e000)
= 0x7fa977730000
mmap(0x7fa97773c000, 16037512, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x7fa97773c000
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\0\220\243\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
fstat(3, {st_mode=S_IFREG|0755, st_size=2125328, ...}) = 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, 2170256, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fa9773df000
mmap(0x7fa977407000, 1605632, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000)
= 0x7fa977407000
mmap(0x7fa97758f000, 323584, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1b0000) =
0x7fa97758f000
mmap(0x7fa9775de000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1fe000)
= 0x7fa9775de000
mmap(0x7fa9775e4000, 52624, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x7fa9775e4000
```

```
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"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=952616, ...}) = 0
mmap(NULL, 950296, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fa9772f6000
mmap(0x7fa977306000, 520192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x10000) =
0x7fa977306000
mmap(0x7fa977385000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x8f000) =
0x7fa977385000
mmap(0x7fa9773dd000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe7000) =
0x7fa9773dd000
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\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832
fstat(3, {st_mode=S_IFREG|0644, st_size=183024, ...}) = 0
mmap(NULL, 185256, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7fa9772c8000
mmap(0x7fa9772cc000, 147456, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x4000) =
0x7fa9772cc000
mmap(0x7fa9772f0000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) =
0x7fa9772f0000
mmap(0x7fa9772f4000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2b000) =
0x7fa9772f4000
close(3)                                = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9772c6000
mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9772c3000
arch_prctl(ARCH_SET_FS, 0x7fa9772c3f00) = 0
set_tid_address(0x7fa9772c41d0)         = 63073
set_robust_list(0x7fa9772c41e0, 24)     = 0
rseq(0x7fa9772c4820, 0x20, 0, 0x53053053) = 0
mprotect(0x7fa9775de000, 16384, PROT_READ) = 0
mprotect(0x7fa9772f4000, 4096, PROT_READ) = 0
mprotect(0x7fa9773dd000, 4096, PROT_READ) = 0
mprotect(0x7fa977730000, 16384, PROT_READ) = 0
mprotect(0x5580d9cfc000, 4096, PROT_READ) = 0
mprotect(0x7fa9786c5000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7fa978688000, 20115)           = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97868c000
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97868b000
openat(AT_FDCWD, "/proc/self/environ", O_RDONLY) = 3
read(3, "SHELL=/bin/bash\0COLORTERM=trueco"..., 4096) = 4096
close(3)                                = 0
munmap(0x7fa97868b000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97868a000
openat(AT_FDCWD, "/proc/self/environ", O_RDONLY) = 3
read(3, "SHELL=/bin/bash\0COLORTERM=trueco"..., 8192) = 4725
read(3, "", 3467)                       = 0
close(3)                                = 0
readlinkat(AT_FDCWD, "/proc/self/exe",
"/mnt/c/Users/\320\225\320\273\320\265\320\275\320\260/Desktop/"..., 4096) = 52
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa978689000
openat(AT_FDCWD, "/proc/self/cmdline", O_RDONLY) = 3
read(3, "./main\0001024\0004\0", 4096)  = 14
read(3, "", 4082)                       = 0
close(3)                                = 0
munmap(0x7fa978689000, 4096)            = 0
mmap(NULL, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9772b3000
mmap(0x720000000000, 1099511635968, PROT_NONE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -
1, 0) = 0x720000000000
mmap(0x730000000000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x730000000000
mmap(NULL, 8388608, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) = 0x7fa976ab3000
mmap(NULL, 4263936, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9766a2000
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9764a2000
munmap(0x7fa9764a2000, 385024)          = 0
munmap(0x7fa976600000, 663552)          = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa978689000
```

```
mmap(NULL, 3727360, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976172000
clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=102249214}) = 0
mmap(0x721400000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x721400000000
mmap(0x721780000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x721780000000
clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=102688516}) = 0
mmap(0x720800000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x720800000000
mmap(0x720b80000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x720b80000000
clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=103168105}) = 0
mmap(0x721000000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x721000000000
mmap(0x721380000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x721380000000
clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=103611967}) = 0
mmap(0x721800000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x721800000000
mmap(0x721b80000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x721b80000000
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa978688000
prlimit64(0, RLIMIT_CORE, NULL, {rlim_cur=0, rlim_max=RLIM64_INFINITY}) = 0
prlimit64(0, RLIMIT_CORE, {rlim_cur=0, rlim_max=RLIM64_INFINITY}, NULL) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
prlimit64(0, RLIMIT_AS, NULL, {rlim_cur=RLIM64_INFINITY, rlim_max=RLIM64_INFINITY}) = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9766a1000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "5580d9cf9000-5580d9cfa000 r--p 0"..., 4096) = 4090
read(3, "7fa978", 6)                    = 6
close(3)                                = 0
munmap(0x7fa9766a1000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9766a0000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "5580d9cf9000-5580d9cfa000 r--p 0"..., 8192) = 4090
read(3, "7fa97868f000-7fa978690000 r--p 0"..., 4102) = 841
read(3, "", 3261)                       = 0
close(3)                                = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97669f000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "5580d9cf9000-5580d9cfa000 r--p 0"..., 4096) = 4090
read(3, "7fa978", 6)                    = 6
close(3)                                = 0
munmap(0x7fa97669f000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97669e000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "5580d9cf9000-5580d9cfa000 r--p 0"..., 8192) = 4090
read(3, "7fa97868f000-7fa978690000 r--p 0"..., 4102) = 841
read(3, "", 3261)                       = 0
close(3)                                = 0
mmap(0x20000000000, 15393162788864, PROT_NONE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -
1, 0) = 0x20000000000
mmap(0x380000000000, 31885837205504, PROT_NONE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -
1, 0) = 0x380000000000
mmap(0x5a0000000000, 26388279066624, PROT_NONE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -
1, 0) = 0x5a0000000000
mmap(0x730000002000, 7696581386240, PROT_NONE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -
1, 0) = 0x730000002000
munmap(0x7fa97669e000, 8192)            = 0
mmap(0x100000000000, 35184372088832, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) = 0x100000000000
madvise(0x100000000000, 35184372088832, MADV_NOHUGEPAGE) = 0
madvise(0x100000000000, 35184372088832, MADV_DONTDUMP) = 0
mmap(0x300000000000, 8796093022208, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) = 0x300000000000
madvise(0x300000000000, 8796093022208, MADV_NOHUGEPAGE) = 0
madvise(0x300000000000, 8796093022208, MADV_DONTDUMP) = 0
```

```
getpid()                                = 63073
openat(AT_FDCWD, "/tmp/tsan.rodata.63073", O_RDWR|O_CREAT|O_EXCL, 0600) = 3
unlinkat(AT_FDCWD, "/tmp/tsan.rodata.63073", 0) = 0
mmap(NULL, 524288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976620000
write(3, "\0\0\0@\0\0\0@\0\0\0@\0\0\0@\0\0\0@\0\0\0@\0\0\0@\0\0\0@"..., 524288) = 524288
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, 3, 0) = 0x7fa97661f000
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97661e000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 4
read(4, "20000000000-100000000000 ---p 00"..., 4096) = 4071
read(4, "7fa977734000-7fa97773c000", 25) = 25
close(4)                                = 0
munmap(0x7fa97661e000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97661d000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 4
read(4, "20000000000-100000000000 ---p 00"..., 8192) = 4071
read(4, "7fa977734000-7fa97773c000 rw-p 0"..., 4121) = 1055
read(4, "", 3066)                       = 0
close(4)                                = 0
munmap(0x7fa9766a0000, 8192)            = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9766a1000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 4
read(4, "20000000000-100000000000 ---p 00"..., 4096) = 4004
read(4, "7fa977730000-7fa977734000 r--p 0"..., 92) = 92
close(4)                                = 0
munmap(0x7fa9766a1000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa9766a0000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 4
read(4, "20000000000-100000000000 ---p 00"..., 8192) = 4071
read(4, "7fa977734000-7fa97773c000 rw-p 0"..., 4121) = 1055
read(4, "", 3066)                       = 0
close(4)                                = 0
mmap(0x1b01b39f4000, 8192, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x1b01b39f4000
mmap(0x2f52ee598000, 294912, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee598000
mmap(0x2f52ee60c000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee60c000
mmap(0x2f52ee68c000, 516096, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee68c000
mmap(0x2f52ee80e000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee80e000
mmap(0x2f52ee88e000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee88e000
mmap(0x2f52ee90e000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee90e000
mmap(0x2f52ee98e000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52ee98e000
mmap(0x2f52eea0e000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eea0e000
mmap(0x2f52eea8e000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eea8e000
mmap(0x2f52eeb0e000, 65536, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eeb0e000
mmap(0x2f52eec56000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eec56000
mmap(0x2f52eecd6000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eecd6000
mmap(0x2f52eed56000, 524288, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eed56000
mmap(0x2f52eedd6000, 139264, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52eedd6000
mmap(0x2f52f0d20000, 352256, PROT_READ, MAP_PRIVATE|MAP_FIXED, 3, 0) = 0x2f52f0d20000
close(3)                                = 0
munmap(0x7fa9766a0000, 8192)            = 0
munmap(0x7fa976620000, 524288)          = 0
sigaltstack(NULL, {ss_sp=NULL, ss_flags=SS_DISABLE, ss_size=0}) = 0
mmap(NULL, 32768, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa97669a000
sigaltstack({ss_sp=0x7fa97669a000, ss_flags=0, ss_size=32768}, NULL) = 0
rt_sigaction(SIGSEGV, {sa_handler=0x7fa97769af20, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_NODEFER|SA_SIGINFO, sa_restorer=0x7fa977424320}, NULL, 8) = 0
rt_sigaction(SIGBUS, {sa_handler=0x7fa97769af20, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_NODEFER|SA_SIGINFO, sa_restorer=0x7fa977424320}, NULL, 8) = 0
rt_sigaction(SIGFPE, {sa_handler=0x7fa97769af20, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_NODEFER|SA_SIGINFO, sa_restorer=0x7fa977424320}, NULL, 8) = 0
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975f72000
munmap(0x7fa975f72000, 581632)          = 0
munmap(0x7fa976100000, 466944)          = 0
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975e00000
munmap(0x7fa975f00000, 1048576)         = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976699000
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975c00000
munmap(0x7fa975d00000, 1048576)         = 0
```

```
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976698000
mmap(NULL, 1703936, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975a60000
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976697000
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975860000
munmap(0x7fa975860000, 655360)          = 0
munmap(0x7fa975a00000, 393216)          = 0
munmap(0x7fa976697000, 4096)            = 0
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975700000
munmap(0x7fa975800000, 1048576)         = 0
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975500000
munmap(0x7fa975600000, 1048576)         = 0
munmap(0x7fa976698000, 4096)            = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976698000
mmap(NULL, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976688000
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976686000
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975300000
munmap(0x7fa975400000, 1048576)         = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976685000
gettid()                                = 63073
mmap(NULL, 524288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) =
0x7fa975f80000
madvise(0x7fa975f80000, 524288, MADV_NOHUGEPAGE) = 0
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975100000
munmap(0x7fa975200000, 1048576)         = 0
mmap(NULL, 2097152, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa974f00000
munmap(0x7fa975000000, 1048576)         = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976684000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "20000000000-100000000000 ---p 00"..., 4096) = 4089
read(3, "7fa975f", 7)                   = 7
close(3)                                = 0
munmap(0x7fa976684000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976683000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "20000000000-100000000000 ---p 00"..., 8192) = 4089
read(3, "7fa975f80000-7fa976000000 rw-p 0"..., 4103) = 3566
read(3, "", 537)                        = 0
close(3)                                = 0
munmap(0x7fa97661d000, 8192)            = 0
mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976682000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "20000000000-100000000000 ---p 00"..., 4096) = 4089
read(3, "7fa975f", 7)                   = 7
close(3)                                = 0
munmap(0x7fa976682000, 4096)            = 0
mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976681000
openat(AT_FDCWD, "/proc/self/maps", O_RDONLY) = 3
read(3, "20000000000-100000000000 ---p 00"..., 8192) = 4089
read(3, "7fa975f80000-7fa976000000 rw-p 0"..., 4103) = 3566
read(3, "", 537)                        = 0
close(3)                                = 0
munmap(0x7fa976681000, 8192)            = 0
mmap(NULL, 262144, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976643000
mmap(0x728000000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x728000000000
mmap(0x728380000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x728380000000
mmap(NULL, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976633000
mmap(NULL, 1048576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa975d00000
mmap(NULL, 8388608, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) =
0x7fa974700000
mmap(NULL, 90112, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7fa976609000
clock_gettime(CLOCK_PROCESS_CPUTIME_ID, {tv_sec=0, tv_nsec=24321400}) = 0
rt_sigprocmask(SIG_SETMASK, ~[SYS RT_1], [], 8) = 0
rt_sigaction(SIGRT_1, {sa_handler=0x7fa977478520, sa_mask=[],
sa_flags=SA_RESTORER|SA_ONSTACK|SA_RESTART|SA_SIGINFO, sa_restorer=0x7fa977424320}, 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) = 0x7fa973eff000
mprotect(0x7fa973f00000, 8388608, PROT_READ|PROT_WRITE) = 0
mmap(0x724400000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x724400000000
mmap(0x724780000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) =
0x724780000000
rt_sigprocmask(SIG_BLOCK, ~[], ~[KILL STOP SYS RTMIN RT_1], 8) = 0
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7fa9746ff990, parent_tid=0x7fa9746ff990,
exit_signal=0, stack=0x7fa973eff000, stack_size=0x7ff7c0, tls=0x7fa9746ff6c0}strace: Process 63074
attached
 => {parent_tid=[63074]}, 88) = 63074
[pid 63074] rseq(0x7fa9746fffe0, 0x20, 0, 0x53053053 <unfinished ...>
[pid 63073] rt_sigprocmask(SIG_SETMASK, ~[KILL STOP SYS RTMIN RT_1],  <unfinished ...>
[pid 63074] <... rseq resumed>)         = 0
[pid 63073] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63074] set_robust_list(0x7fa9746ff9a0, 24 <unfinished ...>
[pid 63073] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63074] <... set_robust_list resumed>) = 0
[pid 63073] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63074] rt_sigprocmask(SIG_SETMASK, ~[KILL STOP SYS RTMIN RT_1],  <unfinished ...>
[pid 63073] mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0 <unfinished
...>
[pid 63074] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63073] <... mmap resumed>)         = 0x7fa9736fe000
[pid 63074] gettimeofday( <unfinished ...>
[pid 63073] mprotect(0x7fa9736ff000, 8388608, PROT_READ|PROT_WRITE <unfinished ...>
[pid 63074] <... gettimeofday resumed>{tv_sec=1733221983, tv_usec=145509}, NULL) = 0
[pid 63073] <... mprotect resumed>)     = 0
[pid 63074] nanosleep({tv_sec=0, tv_nsec=100000000},  <unfinished ...>
[pid 63073] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0
[pid 63073]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7fa973efe990, parent_tid=0x7fa973efe990,
exit_signal=0, stack=0x7fa9736fe000, stack_size=0x7ff7c0, tls=0x7fa973efe6c0}strace: Process 63075
attached
 => {parent_tid=[63075]}, 88) = 63075
[pid 63075] rseq(0x7fa973efefe0, 0x20, 0, 0x53053053 <unfinished ...>
[pid 63073] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63075] <... rseq resumed>)         = 0
[pid 63073] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63075] set_robust_list(0x7fa973efe9a0, 24 <unfinished ...>
[pid 63073] mmap(NULL, 4096, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0 <unfinished ...>
[pid 63075] <... set_robust_list resumed>) = 0
[pid 63073] <... mmap resumed>)         = 0x7fa976632000
[pid 63075] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63073] futex(0x7ffefb57c0a4, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63075] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63073] <... futex resumed>)        = 0
[pid 63073] futex(0x7ffefb57c0a8, FUTEX_WAIT_PRIVATE, 0, NULL <unfinished ...>
[pid 63075] gettid()                    = 63075
[pid 63075] mmap(NULL, 524288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0)
= 0x7fa975f00000
[pid 63075] madvise(0x7fa975f00000, 524288, MADV_NOHUGEPAGE) = 0
[pid 63075] sched_getaffinity(63075, 32, [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]) = 32
[pid 63075] mmap(0x722800000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -
1, 0) = 0x722800000000
[pid 63075] mmap(0x722b80000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -
1, 0) = 0x722b80000000
[pid 63075] mmap(NULL, 262144, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7fa976132000
[pid 63075] mmap(0x2f52e6dff000, 16756736, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) = 0x2f52e6dff000
[pid 63075] madvise(0x2f52e6dff000, 16756736, MADV_NOHUGEPAGE) = 0
[pid 63075] futex(0x7ffefb57c0a8, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63073] <... futex resumed>)        = 0
```

```
[pid 63075] <... futex resumed>)        = 1
[pid 63073] futex(0x7fa973efe990, FUTEX_WAIT_BITSET|FUTEX_CLOCK_REALTIME, 63075, NULL,
FUTEX_BITSET_MATCH_ANY <unfinished ...>
[pid 63075] mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7fa972efd000
[pid 63075] mprotect(0x7fa972efe000, 8388608, PROT_READ|PROT_WRITE) = 0
[pid 63075] mmap(0x724400010000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -
1, 0) = 0x724400010000
[pid 63075] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0
[pid 63075]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7fa9736fd990, parent_tid=0x7fa9736fd990,
exit_signal=0, stack=0x7fa972efd000, stack_size=0x7ff7c0, tls=0x7fa9736fd6c0}strace: Process 63076
attached
 <unfinished ...>
[pid 63076] rseq(0x7fa9736fdfe0, 0x20, 0, 0x53053053 <unfinished ...>
[pid 63075] <... clone3 resumed> => {parent_tid=[63076]}, 88) = 63076
[pid 63076] <... rseq resumed>)         = 0
[pid 63075] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63076] set_robust_list(0x7fa9736fd9a0, 24 <unfinished ...>
[pid 63075] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63076] <... set_robust_list resumed>) = 0
[pid 63075] futex(0x7fa973efd524, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63076] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63075] <... futex resumed>)        = 0
[pid 63076] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63075] futex(0x7fa973efd528, FUTEX_WAIT_PRIVATE, 0, NULL <unfinished ...>
[pid 63076] gettid()                    = 63076
[pid 63076] mmap(NULL, 524288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0)
= 0x7fa975880000
[pid 63076] madvise(0x7fa975880000, 524288, MADV_NOHUGEPAGE) = 0
[pid 63076] sched_getaffinity(63076, 32, [0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]) = 32
[pid 63076] mmap(NULL, 262144, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7fa975a20000
[pid 63076] mmap(0x2f52e5dfd000, 16756736, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0) = 0x2f52e5dfd000
[pid 63076] madvise(0x2f52e5dfd000, 16756736, MADV_NOHUGEPAGE) = 0
[pid 63076] futex(0x7fa973efd528, FUTEX_WAKE_PRIVATE, 1) = 1
[pid 63075] <... futex resumed>)        = 0
[pid 63075] mmap(NULL, 8392704, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_STACK, -1, 0) =
0x7fa9726fc000
[pid 63075] mprotect(0x7fa9726fd000, 8388608, PROT_READ|PROT_WRITE) = 0
[pid 63075] rt_sigprocmask(SIG_BLOCK, ~[], [], 8) = 0
[pid 63076] mmap(NULL, 262144, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0 <unfinished
...>
[pid 63075]
clone3({flags=CLONE_VM|CLONE_FS|CLONE_FILES|CLONE_SIGHAND|CLONE_THREAD|CLONE_SYSVSEM|CLONE_SETTLS|CL
ONE_PARENT_SETTID|CLONE_CHILD_CLEARTID, child_tid=0x7fa972efc990, parent_tid=0x7fa972efc990,
exit_signal=0, stack=0x7fa9726fc000, stack_size=0x7ff7c0, tls=0x7fa972efc6c0} <unfinished ...>
[pid 63076] <... mmap resumed>)         = 0x7fa975840000
strace: Process 63077 attached
[pid 63077] rseq(0x7fa972efcfe0, 0x20, 0, 0x53053053 <unfinished ...>
[pid 63075] <... clone3 resumed> => {parent_tid=[63077]}, 88) = 63077
[pid 63077] <... rseq resumed>)         = 0
[pid 63076] madvise(0x2f52e5dfc000, 16760832, MADV_DONTNEED <unfinished ...>
[pid 63077] set_robust_list(0x7fa972efc9a0, 24 <unfinished ...>
[pid 63075] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63077] <... set_robust_list resumed>) = 0
[pid 63076] <... madvise resumed>)      = 0
[pid 63077] rt_sigprocmask(SIG_SETMASK, [],  <unfinished ...>
[pid 63075] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63077] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63076] madvise(0x2f52e6df9000, 12288, MADV_DONTNEED <unfinished ...>
[pid 63075] futex(0x7fa973efd524, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63077] gettid( <unfinished ...>
[pid 63076] <... madvise resumed>)      = 0
[pid 63077] <... gettid resumed>)       = 63077
```

```
[pid 63075] <... futex resumed>)        = 0
[pid 63077] mmap(NULL, 524288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0
<unfinished ...>
[pid 63076] munmap(0x7fa975880000, 524288 <unfinished ...>
[pid 63077] <... mmap resumed>)         = 0x7fa975680000
[pid 63075] futex(0x7fa973efd528, FUTEX_WAIT_PRIVATE, 0, NULL <unfinished ...>
[pid 63077] madvise(0x7fa975680000, 524288, MADV_NOHUGEPAGE <unfinished ...>
[pid 63076] <... munmap resumed>)       = 0
[pid 63077] <... madvise resumed>)      = 0
[pid 63076] futex(0x7fa9781505a8, FUTEX_WAIT_PRIVATE, 0, NULL <unfinished ...>
[pid 63077] futex(0x7fa9781505a8, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63076] <... futex resumed>)        = -1 EAGAIN (Resource temporarily unavailable)
[pid 63077] <... futex resumed>)        = 0
[pid 63076] clock_gettime(CLOCK_MONOTONIC,  <unfinished ...>
[pid 63077] futex(0x730000000108, FUTEX_WAIT_PRIVATE, 0, NULL <unfinished ...>
[pid 63076] <... clock_gettime resumed>{tv_sec=14139, tv_nsec=143820917}) = 0
[pid 63076] futex(0x730000000108, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63077] <... futex resumed>)        = 0
[pid 63076] <... futex resumed>)        = 1
[pid 63077] sched_getaffinity(63077, 32,  <unfinished ...>
[pid 63076] clock_gettime(CLOCK_MONOTONIC,  <unfinished ...>
[pid 63077] <... sched_getaffinity resumed>[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15]) = 32
[pid 63076] <... clock_gettime resumed>{tv_sec=14139, tv_nsec=144244974}) = 0
[pid 63077] futex(0x730000000508, FUTEX_WAIT_PRIVATE, 0, NULL <unfinished ...>
[pid 63076] futex(0x730000000508, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63077] <... futex resumed>)        = -1 EAGAIN (Resource temporarily unavailable)
[pid 63076] <... futex resumed>)        = 0
[pid 63077] mmap(NULL, 262144, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0 <unfinished
...>
[pid 63076] rt_sigprocmask(SIG_BLOCK, ~[RT_1],  <unfinished ...>
[pid 63077] <... mmap resumed>)         = 0x7fa9758c0000
[pid 63076] <... rt_sigprocmask resumed>NULL, 8) = 0
[pid 63077] mmap(0x2f52e4dfb000, 16756736, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS|MAP_NORESERVE, -1, 0 <unfinished ...>
[pid 63076] madvise(0x7fa972efd000, 8368128, MADV_DONTNEED <unfinished ...>
[pid 63077] <... mmap resumed>)         = 0x2f52e4dfb000
[pid 63076] <... madvise resumed>)      = 0
[pid 63077] madvise(0x2f52e4dfb000, 16756736, MADV_NOHUGEPAGE <unfinished ...>
[pid 63076] exit(0 <unfinished ...>
[pid 63077] <... madvise resumed>)      = 0
[pid 63076] <... exit resumed>)         = ?
[pid 63077] futex(0x7fa973efd528, FUTEX_WAKE_PRIVATE, 1 <unfinished ...>
[pid 63076] +++ exited with 0 +++
[pid 63077] <... futex resumed>)        = 1
[pid 63075] <... futex resumed>)        = 0
[pid 63075] futex(0x7fa972efc990, FUTEX_WAIT_BITSET|FUTEX_CLOCK_REALTIME, 63077, NULL,
FUTEX_BITSET_MATCH_ANY <unfinished ...>
[pid 63077] mmap(NULL, 262144, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7fa975880000
[pid 63077] madvise(0x2f52e4dfa000, 16760832, MADV_DONTNEED) = 0
[pid 63077] madvise(0x2f52e5df7000, 12288, MADV_DONTNEED) = 0
[pid 63077] munmap(0x7fa975680000, 524288) = 0
[pid 63077] clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=146871807}) = 0
[pid 63077] clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=147016816}) = 0
[pid 63077] rt_sigprocmask(SIG_BLOCK, ~[RT_1], NULL, 8) = 0
[pid 63077] madvise(0x7fa9726fc000, 8368128, MADV_DONTNEED) = 0
[pid 63077] exit(0)                     = ?
[pid 63077] +++ exited with 0 +++
[pid 63075] <... futex resumed>)        = 0
[pid 63075] madvise(0x2f52e6dfe000, 16760832, MADV_DONTNEED) = 0
[pid 63075] madvise(0x2f52e7dfb000, 12288, MADV_DONTNEED) = 0
[pid 63075] munmap(0x7fa975f00000, 524288) = 0
[pid 63075] clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=148361079}) = 0
[pid 63075] clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=148480782}) = 0
[pid 63075] clock_gettime(CLOCK_MONOTONIC, {tv_sec=14139, tv_nsec=148643437}) = 0
[pid 63075] rt_sigprocmask(SIG_BLOCK, ~[RT_1], NULL, 8) = 0
[pid 63075] madvise(0x7fa9736fe000, 8368128, MADV_DONTNEED) = 0
```

```
[pid 63075] exit(0)                     = ?
[pid 63073] <... futex resumed>)        = 0
[pid 63075] +++ exited with 0 +++
[pid 63073] clock_gettime(CLOCK_PROCESS_CPUTIME_ID, {tv_sec=0, tv_nsec=31190600}) = 0
[pid 63073] fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x5), ...}) = 0
[pid 63073] mmap(0x726000000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -
1, 0) = 0x726000000000
[pid 63073] mmap(0x726380000000, 65536, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -
1, 0) = 0x726380000000
[pid 63073] write(1, "Number of threads requested: 4\n", 31Number of threads requested: 4
) = 31
[pid 63073] write(1, "Total threads created: 3\n", 25Total threads created: 3
) = 25
[pid 63073] write(1, "Time taken: 0.006869 seconds\n", 29Time taken: 0.006869 seconds
) = 29
[pid 63073] exit_group(0)               = ?
[pid 63074] <... nanosleep resumed> <unfinished ...>) = ?
[pid 63074] +++ exited with 0 +++
+++ exited with 0 +++
```

Вывод

В ходе выполнения данной лабораторной работы я научилась создавать программы на языке C, которые обрабатывают данные в многопоточном режиме, а также применять стандартные средства операционной системы для управления потоками и их синхронизации.

В результате экспериментов и тестирования программы, реализующей битоническую сортировку, я проанализировала, как входной размер данных и количество потоков влияют на ускорение и эффективность выполнения алгоритма.

Оказалось, что увеличение числа потоков приводит к заметному ускорению выполнения программы при обработке больших массивов данных. Однако эффективность использования вычислительных ресурсов начинает снижаться при превышении количества потоков, соответствующего числу логических ядер процессора. Это связано с дополнительными затратами на управление потоками и синхронизацию между ними.

Также в ходе работы я приобрела ценные навыки в области работы с потоками и их синхронизацией. Лабораторная работа была интересной и позволила мне глубже понять многопоточность и оптимизацию вычислений.