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-215Б-23

Студент: Тараскаев Д.М.

Преподаватель: Миронов Е.С. (ПМИ)

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: 01.11.24

Москва, 2024

**Постановка задачи**

**Вариант 16.**

**Задаётся радиус окружности. Необходимо с помощью метода Монте-Карло рассчитать её площадь**

**Общий метод и алгоритм решения**

Использованные системные вызовы:

* pthread\_mutex\_init(&mutex, NULL); - создание мутекса
* pthread\_mutex\_unlock(&mutex); - разблокировка мутекса
* pthread\_mutex\_lock(&mutex); - блокировка мутекса
* pthread\_exit(NULL); - завершение работы потока
* pthread\_create(&threads[i], NULL, monte\_carlo, &attrs); - создание потока
* pthread\_join(threads[i], NULL); - ожидание завершения потока
* pthread\_mutex\_destroy(&mutex); - удаление мутекса

**Программа выполняет многопоточное вычисление числа точек, попадающих в окружность, используя метод Монте-Карло. Она создает несколько потоков, каждый из которых выполняет часть вычислений, а затем объединяет результаты.**

**Основные шаги работы программы:**

1. **Программа принимает в качестве аргументов радиус окружности и количество поток**
2. **Общее количество точек равномерно распределяется по потокам**
3. **Создаются потоки, где в каждом происходит генерация точек x и y и идет проверка условия, что точка находится внутри окружности.**
4. **Общее количество точек попадающих круг суммируется с данными от других потоков**
5. **Считается площадь круга**

**Код программы**

**main.c**

#include <stdio.h>

#include <stdlib.h>

#include <unistd.h>

#include <string.h>

#include <pthread.h>

#include <time.h>

#define TOTAL\_PTS 1000000

typedef struct {

double radius;

long pts\_thread;

} thread\_attrs;

pthread\_mutex\_t mutex;

long total\_in\_circle = 0;

void\* monte\_carlo(void\* arg) {

thread\_attrs\* attrs = (thread\_attrs\*)arg;

long in\_circle = 0;

unsigned int seed = rand();

for (long i = 0; i < attrs->pts\_thread; i++) {

double x = (double)rand\_r(&seed) / RAND\_MAX \* attrs->radius;

double y = (double)rand\_r(&seed) / RAND\_MAX \* attrs->radius;

if (x \* x + y \* y <= attrs->radius \* attrs->radius) {

in\_circle++;

}

}

pthread\_mutex\_lock(&mutex);

total\_in\_circle += in\_circle;

pthread\_mutex\_unlock(&mutex);

pthread\_exit(NULL);

}

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

// struct timespec start, end;

// clock\_gettime(CLOCK\_MONOTONIC, &start);

if (argc != 3) {

printf("Using: %s <radius> <thread\_count>\n", argv[0]);

return 1;

}

double radius = atof(argv[1]);

if (radius < 0) {

printf("Negative radius\n");

return -1;

}

int threads\_count = atoi(argv[2]);

srand(time(NULL));

pthread\_t threads[threads\_count];

long long pts\_thread = TOTAL\_PTS / threads\_count;

thread\_attrs attrs;

attrs.radius = radius;

attrs.pts\_thread = pts\_thread;

pthread\_mutex\_init(&mutex, NULL);

for (int i = 0; i < threads\_count; ++i) {

pthread\_create(&threads[i], NULL, monte\_carlo, &attrs);

}

for (int i = 0; i < threads\_count; ++i) {

pthread\_join(threads[i], NULL);

}

pthread\_mutex\_destroy(&mutex);

double area = 4 \* radius \* radius \* ((double)total\_in\_circle / (double)TOTAL\_PTS);

printf("Circle area with radius %.2f is %.5f\n", radius, area);

// clock\_gettime(CLOCK\_MONOTONIC, &end);

// double elapsed = (end.tv\_sec - start.tv\_sec) + (end.tv\_nsec - start.tv\_nsec) / 1e9;

// printf("Execution time: %.10f seconds\n", elapsed);

return 0;

}

**Протокол работы программы**

**Тестирование:**

$ ./main -6 16

Negative radius

$ ./main 10 16

Circle area with radius 10.00 is 314.15112

$ ./main 20 4

Circle area with radius 20.00 is 1256.67829

**Strace:**

$ strace -f ./main 5 4

execve("./main", ["./main", "5", "4"], 0x7ffcf3cb3ca8 /\* 50 vars \*/) = 0

brk(NULL) = 0x5b46adac8000

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=121423, ...}) = 0

mmap(NULL, 121423, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7b82c2c90000

close(3) = 0

openat(AT\_FDCWD, "/usr/lib/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\340\_\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=2014520, ...}) = 0

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7b82c2c8e000

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, 2034616, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7b82c2a9d000

mmap(0x7b82c2ac1000, 1511424, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x24000) = 0x7b82c2ac1000

mmap(0x7b82c2c32000, 319488, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x195000) = 0x7b82c2c32000

mmap(0x7b82c2c80000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e3000) = 0x7b82c2c80000

mmap(0x7b82c2c86000, 31672, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7b82c2c86000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7b82c2a9a000

arch\_prctl(ARCH\_SET\_FS, 0x7b82c2a9a740) = 0

set\_tid\_address(0x7b82c2a9aa10) = 45190

set\_robust\_list(0x7b82c2a9aa20, 24) = 0

rseq(0x7b82c2a9b060, 0x20, 0, 0x53053053) = 0

mprotect(0x7b82c2c80000, 16384, PROT\_READ) = 0

mprotect(0x5b469a694000, 4096, PROT\_READ) = 0

mprotect(0x7b82c2ce8000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7b82c2c90000, 121423) = 0

rt\_sigaction(SIGRT\_1, {sa\_handler=0x7b82c2b2e2b0, sa\_mask=[], sa\_flags=SA\_RESTORER|SA\_ONSTACK|SA\_RESTART|SA\_SIGINFO, sa\_restorer=0x7b82c2ada1d0}, 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) = 0x7b82c2200000

mprotect(0x7b82c2201000, 8388608, PROT\_READ|PROT\_WRITE) = 0

getrandom("\xe1\xab\xcf\x06\xb4\x30\xff\x30", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x5b46adac8000

brk(0x5b46adae9000) = 0x5b46adae9000

rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0

clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7b82c2a00990, parent\_tid=0x7b82c2a00990, exit\_signal=0, stack=0x7b82c2200000, stack\_size=0x7fff80, tls=0x7b82c2a006c0}strace: Process 45191 attached

=> {parent\_tid=[45191]}, 88) = 45191

[pid 45191] rseq(0x7b82c2a00fe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 45190] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45191] <... rseq resumed>) = 0

[pid 45190] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45190] mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0 <unfinished ...>

[pid 45191] set\_robust\_list(0x7b82c2a009a0, 24 <unfinished ...>

[pid 45190] <... mmap resumed>) = 0x7b82c1800000

[pid 45191] <... set\_robust\_list resumed>) = 0

[pid 45190] mprotect(0x7b82c1801000, 8388608, PROT\_READ|PROT\_WRITE <unfinished ...>

[pid 45191] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45190] <... mprotect resumed>) = 0

[pid 45191] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45190] rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0

[pid 45190] clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7b82c2000990, parent\_tid=0x7b82c2000990, exit\_signal=0, stack=0x7b82c1800000, stack\_size=0x7fff80, tls=0x7b82c20006c0}strace: Process 45192 attached

=> {parent\_tid=[45192]}, 88) = 45192

[pid 45192] rseq(0x7b82c2000fe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 45190] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45192] <... rseq resumed>) = 0

[pid 45190] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45192] set\_robust\_list(0x7b82c20009a0, 24 <unfinished ...>

[pid 45190] mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0 <unfinished ...>

[pid 45192] <... set\_robust\_list resumed>) = 0

[pid 45190] <... mmap resumed>) = 0x7b82c0e00000

[pid 45192] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45190] mprotect(0x7b82c0e01000, 8388608, PROT\_READ|PROT\_WRITE <unfinished ...>

[pid 45192] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45190] <... mprotect resumed>) = 0

[pid 45190] rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0

[pid 45190] clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7b82c1600990, parent\_tid=0x7b82c1600990, exit\_signal=0, stack=0x7b82c0e00000, stack\_size=0x7fff80, tls=0x7b82c16006c0}strace: Process 45193 attached

=> {parent\_tid=[45193]}, 88) = 45193

[pid 45193] rseq(0x7b82c1600fe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 45190] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45193] <... rseq resumed>) = 0

[pid 45190] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45193] set\_robust\_list(0x7b82c16009a0, 24 <unfinished ...>

[pid 45190] mmap(NULL, 8392704, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS|MAP\_STACK, -1, 0 <unfinished ...>

[pid 45193] <... set\_robust\_list resumed>) = 0

[pid 45190] <... mmap resumed>) = 0x7b82c0400000

[pid 45193] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45190] mprotect(0x7b82c0401000, 8388608, PROT\_READ|PROT\_WRITE <unfinished ...>

[pid 45193] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45190] <... mprotect resumed>) = 0

[pid 45190] rt\_sigprocmask(SIG\_BLOCK, ~[], [], 8) = 0

[pid 45190] clone3({flags=CLONE\_VM|CLONE\_FS|CLONE\_FILES|CLONE\_SIGHAND|CLONE\_THREAD|CLONE\_SYSVSEM|CLONE\_SETTLS|CLONE\_PARENT\_SETTID|CLONE\_CHILD\_CLEARTID, child\_tid=0x7b82c0c00990, parent\_tid=0x7b82c0c00990, exit\_signal=0, stack=0x7b82c0400000, stack\_size=0x7fff80, tls=0x7b82c0c006c0}strace: Process 45194 attached

=> {parent\_tid=[45194]}, 88) = 45194

[pid 45194] rseq(0x7b82c0c00fe0, 0x20, 0, 0x53053053 <unfinished ...>

[pid 45190] rt\_sigprocmask(SIG\_SETMASK, [], <unfinished ...>

[pid 45194] <... rseq resumed>) = 0

[pid 45190] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45194] set\_robust\_list(0x7b82c0c009a0, 24 <unfinished ...>

[pid 45190] futex(0x7b82c2a00990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 45191, NULL, FUTEX\_BITSET\_MATCH\_ANY <unfinished ...>

[pid 45194] <... set\_robust\_list resumed>) = 0

[pid 45194] rt\_sigprocmask(SIG\_SETMASK, [], NULL, 8) = 0

[pid 45192] openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3

[pid 45192] fstat(3, {st\_mode=S\_IFREG|0644, st\_size=121423, ...}) = 0

[pid 45192] mmap(NULL, 121423, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7b82c2c90000

[pid 45192] close(3) = 0

[pid 45192] mmap(NULL, 134217728, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7b82b8400000

[pid 45192] munmap(0x7b82b8400000, 62914560) = 0

[pid 45192] munmap(0x7b82c0000000, 4194304 <unfinished ...>

[pid 45191] futex(0x7b82c2ceaa08, FUTEX\_WAIT\_PRIVATE, 2, NULL <unfinished ...>

[pid 45192] <... munmap resumed>) = 0

[pid 45192] mprotect(0x7b82bc000000, 135168, PROT\_READ|PROT\_WRITE) = 0

[pid 45192] openat(AT\_FDCWD, "/usr/lib/libgcc\_s.so.1", O\_RDONLY|O\_CLOEXEC) = 3

[pid 45192] 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

[pid 45192] fstat(3, {st\_mode=S\_IFREG|0644, st\_size=915712, ...}) = 0

[pid 45192] mmap(NULL, 184808, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7b82c2a6c000

[pid 45192] mmap(0x7b82c2a70000, 147456, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x4000) = 0x7b82c2a70000

[pid 45192] mmap(0x7b82c2a94000, 16384, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7b82c2a94000

[pid 45192] mmap(0x7b82c2a98000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2b000) = 0x7b82c2a98000

[pid 45192] close(3) = 0

[pid 45192] mprotect(0x7b82c2a98000, 4096, PROT\_READ) = 0

[pid 45192] munmap(0x7b82c2c90000, 121423) = 0

[pid 45192] futex(0x7b82c2ceaa08, FUTEX\_WAKE\_PRIVATE, 1) = 1

[pid 45191] <... futex resumed>) = 0

[pid 45192] futex(0x7b82c2a99070, FUTEX\_WAKE\_PRIVATE, 2147483647 <unfinished ...>

[pid 45191] futex(0x7b82c2ceaa08, FUTEX\_WAKE\_PRIVATE, 1 <unfinished ...>

[pid 45192] <... futex resumed>) = 0

[pid 45191] <... futex resumed>) = 0

[pid 45192] rt\_sigprocmask(SIG\_BLOCK, ~[RT\_1], <unfinished ...>

[pid 45191] rt\_sigprocmask(SIG\_BLOCK, ~[RT\_1], <unfinished ...>

[pid 45192] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45191] <... rt\_sigprocmask resumed>NULL, 8) = 0

[pid 45192] madvise(0x7b82c1800000, 8368128, MADV\_DONTNEED <unfinished ...>

[pid 45191] madvise(0x7b82c2200000, 8368128, MADV\_DONTNEED <unfinished ...>

[pid 45192] <... madvise resumed>) = 0

[pid 45191] <... madvise resumed>) = 0

[pid 45192] exit(0 <unfinished ...>

[pid 45191] exit(0 <unfinished ...>

[pid 45192] <... exit resumed>) = ?

[pid 45191] <... exit resumed>) = ?

[pid 45192] +++ exited with 0 +++

[pid 45190] <... futex resumed>) = 0

[pid 45191] +++ exited with 0 +++

[pid 45190] futex(0x7b82c1600990, FUTEX\_WAIT\_BITSET|FUTEX\_CLOCK\_REALTIME, 45193, NULL, FUTEX\_BITSET\_MATCH\_ANY <unfinished ...>

[pid 45194] rt\_sigprocmask(SIG\_BLOCK, ~[RT\_1], NULL, 8) = 0

[pid 45194] madvise(0x7b82c0400000, 8368128, MADV\_DONTNEED) = 0

[pid 45194] exit(0) = ?

[pid 45194] +++ exited with 0 +++

[pid 45193] rt\_sigprocmask(SIG\_BLOCK, ~[RT\_1], NULL, 8) = 0

[pid 45193] madvise(0x7b82c0e00000, 8368128, MADV\_DONTNEED) = 0

[pid 45193] exit(0) = ?

[pid 45193] +++ exited with 0 +++

<... futex resumed>) = 0

fstat(1, {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}) = 0

write(1, "Circle area with radius 5.00 is "..., 41Circle area with radius 5.00 is 78.54487

) = 41

exit\_group(0) = ?

+++ exited with 0 +++

**Сравнение скорости выполнения программы от количества потоков**

| **Число потоков** | **Время исполнения (мс)** | **Ускорение** | **Эффективность** |
| --- | --- | --- | --- |
| 1 | 0.0145880630 | 1 | 1 |
| 2 | 0.0073788710 | 1.98 | 0.99 |
| 3 | 0.0058537980 | 2.5 | 0.83 |
| 4 | 0.0043141160 | 3.38 | 0.85 |
| 5 | 0.0033514660 | 4.35 | 0.87 |
| 6 | 0.0033016780 | 4.42 | 0.74 |

**Вывод**

**Программа успешно демонстрирует использование многопоточности для параллельных вычислений методом Монте-Карло. По таблице видно что эффективнее всего использовать 5 потоков. При использовании 6 и более потоков время выполнения программы практически не меняется.**