Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

**Лабораторная работа №4 по курсу**

**«Операционные системы»**

Группа: М8О-215Б-23

Студент: Павлов О.В.

Преподаватель: Миронов Е.С.

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

Дата: 13.03.25

Москва, 2025

**Постановка задачи**

**Вариант 23.**

**Контракт 4:**

1. Описание. Подсчёт наибольшего общего делителя для двух натуральных чисел.
2. Сигнатура: Int GCF(int A, int B)
3. Реализация 1: Алгоритм Евклида
4. Реализация 2: Наивный алгоритм. Пытаться разделить числа на все числа, что меньше A и B.

**Контракт 6:**

1. Описание: Рассчет значения числа е(основание натурального логарифма)
2. Сигнатура: Float E(int x)
3. Реализация 1: (1 + 1/x) ^ x
4. Реализация 2: Сумма ряда по n от 0 до x, где элементы ряда равны: (1/(n!))

**Общий метод и алгоритм решения**

**Использованные системные вызовы:**

* void \*dlopen(const char \*filename, int flags) – функция загружает динамически разделенный объект (разделенную библиотеку) с именем filename, заканчивающимся нулевым детерменантом, и возвращает при успехе не NULL “handle” (дескриптор) для загруженного объект, при неудаче – NULL;
* int dlclose(void \*handle) – функция уменьшает число ссылок на динамически загружаемый разделенных объект, на который ссылается handle. При успехе возвращает 0, иначе возвращает ненулевое значение;
* void \*dlsym(void \*restrict handle, const char \*restrict symbol) – функция принимает дескриптор динамически загруженного разделенного объекта, возвращаемый dlopen() вместе с символическим именем, заканчивающимся ‘\0’, и возвращает адрес, где этот символ загружен в памяти. Если символ не найден в указанном объекте или в любых разделенных объектах, которые были автоматически загружены с помощь dlopen() когда эти объекта были загружены, то dlsym() возвращает NULL.

**Использованные флаги Makefile:**

* -c – компиляция исходного кода в объектный файл без линковки;
* -fPIC (Position Independent Code) – генерация позиционно-независимого кода, необходимого для создания динамических библиотек .so;
* -shared – создание динамической библиотеки .so из объектных файлов;
* -ldl - подключение библиотеки libdl, которая предоставляет функции для динамической загрузки кода (например, dlopen, dlsym). Используется в program2;
* -lm - подключение математической библиотеки libm, необходимой для функций вроде log, pow и т.д;
* -L - указание линковщику искать библиотеки в текущей директории (.). Используется в program1.

**Код программы**

**libgcf.h**

#ifndef LIBGCF\_H

#define LIBGCF\_H

int GCF\_EUCLID(int A, int B);

int GCF\_NATIVE(int A, int B);

#endif

**liblog.h**

#ifndef LIBLOG\_H

#define LIBLOG\_H

float E\_FORMULA(int x);

float E\_SUMM(int x);

#endif

**libgcf\_euclid.c**

#include "libgcf.h"

int GCF\_EUCLID(int A, int B)

{

while (B != 0)

{

int temp = B;

B = A % B;

A = temp;

}

return A;

}

**libgcf\_native.c**

#include "libgcf.h"

int GCF\_NATIVE(int A, int B)

{

int gcd = 1;

for (int i = 1; i <= A && i <= B; i++)

if (A % i == 0 && B % i == 0)

gcd = i;

return gcd;

}

**liblog\_formula.c**

#include "liblog.h"

#include <math.h>

// (1 + 1/x)^x

float E\_FORMULA(int x)

{

return powf((1.0f + 1.0f / x), x);

}

**liblog\_summ.c**

#include "liblog.h"

// Сумма ряда (1/n!)

float E\_SUMM(int x)

{

float e = 1.0f;

float fact = 1.0f;

for (int i = 1; i <= x; i++)

{

fact \*= i;

e += 1.0f / fact;

}

return e;

}

**program1.c**

#include <stdio.h>

#include "libgcf.h"

#include "liblog.h"

int main()

{

char command[256];

printf("Команды:\n");

printf("1 <int> <int> для подсчета НОД.\n");

printf("2 <int> для подсчета E.\n");

printf("q для выхода.\n");

printf("\n");

while (1)

{

printf("Введите команду: ");

fgets(command, sizeof(command), stdin);

if (command[0] == '1') // Подсчет НОД

{

int a, b;

if (sscanf(command, "1 %d %d", &a, &b) != 2)

{

printf("Некорректный ввод. Введите: 1 <int> <int>\n\n");

continue;

}

printf("НОД (%d, %d) - Евклид: %d\n", a, b, GCF\_EUCLID(a, b));

printf("НОД (%d, %d) - Наивный: %d\n", a, b, GCF\_NATIVE(a, b));

printf("\n");

}

else if (command[0] == '2') // Расчет числа e

{

int x;

if (sscanf(command, "%d", &x) != 1)

{

printf("Некорректный ввод. Введите: 2 <int>\n\n");

continue;

}

printf("E(%d) - По формуле: %f\n", x, E\_FORMULA(x));

printf("E(%d) - По сумме рядов: %f\n", x, E\_SUMM(x));

printf("\n");

}

else if (command[0] == 'q') // Выход

break;

else

printf("Неизвестная команда. Используйте 1 для НОД и 2 для E.\n\n");

}

printf("\n");

return 0;

}

**program2.c**

#include <stdio.h>

#include <stdlib.h>

#include <dlfcn.h>

#include <string.h>

int main()

{

void \*gcf\_lib, \*log\_lib;

int (\*GCF\_EUCLID)(int, int), (\*GCF\_NATIVE)(int, int);

float (\*E\_FORMULA)(int), (\*E\_SUMM)(int);

gcf\_lib = dlopen("./libgcf.so", RTLD\_LAZY);

if (!gcf\_lib)

{

fprintf(stderr, "Ошибка при загрузке libgcf.so: %s\n", dlerror());

exit(EXIT\_FAILURE);

}

log\_lib = dlopen("./liblog.so", RTLD\_LAZY);

if (!log\_lib)

{

fprintf(stderr, "Ошибка при загрузке liblog.so: %s\n", dlerror());

dlclose(gcf\_lib);

exit(EXIT\_FAILURE);

}

GCF\_EUCLID = dlsym(gcf\_lib, "GCF\_EUCLID");

if (!GCF\_EUCLID)

{

fprintf(stderr, "Ошибка загрузки GCF\_EUCLID: %s\n", dlerror());

dlclose(gcf\_lib);

dlclose(log\_lib);

exit(EXIT\_FAILURE);

}

GCF\_NATIVE = dlsym(gcf\_lib, "GCF\_NATIVE");

if (!GCF\_NATIVE)

{

fprintf(stderr, "Ошибка загрузки GCF\_NATIVE: %s\n", dlerror());

dlclose(gcf\_lib);

dlclose(log\_lib);

exit(EXIT\_FAILURE);

}

E\_FORMULA = dlsym(log\_lib, "E\_FORMULA");

if (!E\_FORMULA)

{

fprintf(stderr, "Ошибка загрузки E\_FORMULA: %s\n", dlerror());

dlclose(gcf\_lib);

dlclose(log\_lib);

exit(EXIT\_FAILURE);

}

E\_SUMM = dlsym(log\_lib, "E\_SUMM");

if (!E\_SUMM)

{

fprintf(stderr, "Ошибка загрузки E\_SUMM: %s\n", dlerror());

dlclose(gcf\_lib);

dlclose(log\_lib);

exit(EXIT\_FAILURE);

}

char command[256];

printf("Команды:\n");

printf("1 <int> <int> для подсчета НОД.\n");

printf("2 <int> для подсчета E.\n");

printf("q для выхода.\n");

printf("\n");

while (1)

{

printf("Введите команду: ");

fgets(command, sizeof(command), stdin);

if (command[0] == '1') // Подсчет НОД

{

int a, b;

if (sscanf(command, "1 %d %d", &a, &b) != 2)

{

printf("Некорректный ввод. Введите: 1 <int> <int>\n\n");

continue;

}

printf("НОД (%d, %d) - Евклид: %d\n", a, b, GCF\_EUCLID(a, b));

printf("НОД (%d, %d) - Наивный: %d\n", a, b, GCF\_NATIVE(a, b));

printf("\n");

}

else if (command[0] == '2') //Подсчет числа e

{

int x;

if (sscanf(command, "2 %d", &x) != 1)

{

printf("Некорректный ввод. Введите: 2 <int>\n\n");

continue;

}

printf("Число e (%d) - по формуле: %.6f\n", x, E\_FORMULA(x));

printf("Число e (%d) - по сумме рядов: %.6f\n", x, E\_SUMM(x));

printf("\n");

}

else if (command[0] == 'q') //Выход

break;

else

printf("Неизвестная команда. Используйте 1 для НОД и 2 для E.\n\n");

}

printf("\n");

dlclose(gcf\_lib);

dlclose(log\_lib);

return 0;

}

**Makefile**

# Компилятор и флаги

CC = gcc

CFLAGS = -c -fPIC -Wall

LDFLAGS = -shared

DLFLAGS = -ldl

MATH\_LIB = -lm

RM = rm -f

# Исходные файлы и программы

GCF\_SRCS = libgcf\_euclid.c libgcf\_native.c

LOG\_SRCS = liblog\_formula.c liblog\_summ.c

GCF\_OBJS = libgcf\_euclid.o libgcf\_native.o

LOG\_OBJS = liblog\_formula.o liblog\_summ.o

PROGRAMS = program1 program2

# Цели

all: $(PROGRAMS)

# Динамические библиотеки

libgcf.so: $(GCF\_OBJS)

$(CC) $(LDFLAGS) -o $@ $^

liblog.so: $(LOG\_OBJS)

$(CC) $(LDFLAGS) -o $@ $^ $(MATH\_LIB)

# Компиляция исходников

libgcf\_euclid.o: libgcf\_euclid.c

$(CC) $(CFLAGS) $< -o $@

libgcf\_native.o: libgcf\_native.c

$(CC) $(CFLAGS) $< -o $@

liblog\_formula.o: liblog\_formula.c

$(CC) $(CFLAGS) $< -o $@

liblog\_summ.o: liblog\_summ.c

$(CC) $(CFLAGS) $< -o $@

# Сборка программ

program1: program1.c libgcf.so liblog.so

$(CC) -o $@ $< -L. -lgcf -llog $(MATH\_LIB)

program2: program2.c

$(CC) -o $@ $< $(DLFLAGS) $(MATH\_LIB)

# Запуск программ

run1: program1

LD\_LIBRARY\_PATH=$(PWD):$$LD\_LIBRARY\_PATH ./program1

run2: program2

LD\_LIBRARY\_PATH=$(PWD):$$LD\_LIBRARY\_PATH ./program2

# Очистка

clean:

echo "Removing object files and binaries..."

$(RM) -v \*.o \*.so $(PROGRAMS)

**Протокол работы программы**

**Тестирование:**

$ make

gcc -c -fPIC -Wall libgcf\_euclid.c -o libgcf\_euclid.o

gcc -c -fPIC -Wall libgcf\_native.c -o libgcf\_native.o

gcc -shared -o libgcf.so libgcf\_euclid.o libgcf\_native.o

gcc -c -fPIC -Wall liblog\_formula.c -o liblog\_formula.o

gcc -c -fPIC -Wall liblog\_summ.c -o liblog\_summ.o

gcc -shared -o liblog.so liblog\_formula.o liblog\_summ.o -lm

gcc -o program1 program1.c -L. -lgcf -llog -lm

gcc -o program2 program2.c -ldl -lm

$ ./program1

Команды:

1 <int> <int> для подсчета НОД.

2 <int> для подсчета E.

q для выхода.

Введите команду: 1 2 6

НОД (2, 6) - Евклид: 2

НОД (2, 6) - Наивный: 2

Введите команду: 2 999

E(2) - По формуле: 2.250000

E(2) - По сумме рядов: 2.500000

Введите команду: q

$ ./program2

Команды:

1 <int> <int> для подсчета НОД.

2 <int> для подсчета E.

q для выхода.

Введите команду: 1 2

Некорректный ввод. Введите: 1 <int> <int>

Введите команду: 2 q

Некорректный ввод. Введите: 2 <int>

Введите команду: q

**Strace:**

**$ strace -f ./program1**

execve("./program1", ["./program1"], 0xffffc393b400 /\* 30 vars \*/) = 0

brk(NULL) = 0xaaaae7234000

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffffb2b7b000

faccessat(AT\_FDCWD, "/etc/ld.so.preload", R\_OK) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/aarch64/atomics/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls/aarch64/atomics", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/aarch64/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls/aarch64", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/atomics/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls/atomics", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/aarch64/atomics/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/aarch64/atomics", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/aarch64/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/aarch64", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/atomics/libgcf.so", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/atomics", 0xffffedd1c660, 0) = -1 ENOENT (No such file or directory)

**openat(AT\_FDCWD, "/workspaces/src/libgcf.so", O\_RDONLY|O\_CLOEXEC) = 3**

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=7936, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 135200, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffffb2b24000

mmap(0xffffb2b30000, 69664, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffffb2b30000

munmap(0xffffb2b24000, 49152) = 0

munmap(0xffffb2b42000, 12320) = 0

mprotect(0xffffb2b31000, 61440, PROT\_NONE) = 0

mmap(0xffffb2b40000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffffb2b40000

close(3) = 0

**openat(AT\_FDCWD, "/workspaces/src/liblog.so", O\_RDONLY|O\_CLOEXEC) = 3**

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=8184, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 135208, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffffb2b0e000

mmap(0xffffb2b10000, 69672, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffffb2b10000

munmap(0xffffb2b0e000, 8192) = 0

munmap(0xffffb2b22000, 53288) = 0

mprotect(0xffffb2b11000, 61440, PROT\_NONE) = 0

mmap(0xffffb2b20000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffffb2b20000

close(3) = 0

openat(AT\_FDCWD, "/workspaces/src/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "libc.so.6", O\_RDONLY|O\_CLOEXEC) = -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=12079, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 12079, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0xffffb2b78000

close(3) = 0

**openat(AT\_FDCWD, "/lib/aarch64-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\267\0\1\0\0\0\340u\2\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=1637400, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 1805928, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffffb2957000

mmap(0xffffb2960000, 1740392, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffffb2960000

munmap(0xffffb2957000, 36864) = 0

munmap(0xffffb2b09000, 28264) = 0

mprotect(0xffffb2ae8000, 61440, PROT\_NONE) = 0

mmap(0xffffb2af7000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x187000) = 0xffffb2af7000

mmap(0xffffb2afd000, 48744, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0xffffb2afd000

close(3) = 0

openat(AT\_FDCWD, "/workspaces/src/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

**openat(AT\_FDCWD, "/lib/aarch64-linux-gnu/libm.so.6", O\_RDONLY|O\_CLOEXEC) = 3**

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=551064, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 680048, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffffb28b9000

mmap(0xffffb28c0000, 614512, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffffb28c0000

munmap(0xffffb28b9000, 28672) = 0

munmap(0xffffb2957000, 32880) = 0

mprotect(0xffffb2946000, 61440, PROT\_NONE) = 0

mmap(0xffffb2955000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x85000) = 0xffffb2955000

close(3) = 0

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffffb2b76000

set\_tid\_address(0xffffb2b764f0) = 3555

set\_robust\_list(0xffffb2b76500, 24) = 0

rseq(0xffffb2b76bc0, 0x20, 0, 0xd428bc00) = 0

mprotect(0xffffb2af7000, 16384, PROT\_READ) = 0

mprotect(0xffffb2955000, 4096, PROT\_READ) = 0

mprotect(0xffffb2b20000, 4096, PROT\_READ) = 0

mprotect(0xffffb2b40000, 4096, PROT\_READ) = 0

mprotect(0xaaaab1b51000, 4096, PROT\_READ) = 0

mprotect(0xffffb2b80000, 8192, PROT\_READ) = 0

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

munmap(0xffffb2b78000, 12079) = 0

newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0

getrandom("\xfe\xf0\x64\x8c\x0a\xd1\x14\xb5", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0xaaaae7234000

brk(0xaaaae7255000) = 0xaaaae7255000

write(1, "\320\232\320\276\320\274\320\260\320\275\320\264\321\213:\n", 16) = 16

write(1, "1 <int> <int> \320\264\320\273\321\217 \320\277\320\276\320\264\321\201\321\207\320"..., 46) = 46

write(1, "2 <int> \320\264\320\273\321\217 \320\277\320\276\320\264\321\201\321\207\320\265\321\202\320\260 "..., 35) = 35

write(1, "q \320\264\320\273\321\217 \320\262\321\213\321\205\320\276\320\264\320\260.\n", 23) = 23

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

newfstatat(0, "", {st\_mode=S\_IFREG|0644, st\_size=12, ...}, AT\_EMPTY\_PATH) = 0

read(0, "1 2 6\n2 4\nq\n", 4096) = 12

write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265 \320\272\320\276\320\274\320\260\320\275\320\264\321\203: \320"..., 63) = 63

write(1, "\320\235\320\236\320\224 (2, 6) - \320\235\320\260\320\270\320\262\320\275\321\213\320\271: "..., 34) = 34

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

write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265 \320\272\320\276\320\274\320\260\320\275\320\264\321\203: E"..., 68) = 68

write(1, "E(2) - \320\237\320\276 \321\201\321\203\320\274\320\274\320\265 \321\200\321\217\320\264\320\276\320"..., 44) = 44

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

write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265 \320\272\320\276\320\274\320\260\320\275\320\264\321\203: \n", 32) = 32

exit\_group(0) = ?

+++ exited with 0 +++

**$ strace -f ./program2**

execve("./program2", ["./program2"], 0xffffd22ea9b0 /\* 30 vars \*/) = 0

brk(NULL) = 0xaaaadecd0000

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffff86cdb000

faccessat(AT\_FDCWD, "/etc/ld.so.preload", R\_OK) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/aarch64/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls/aarch64/atomics", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/aarch64/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls/aarch64", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls/atomics", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/tls/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/tls", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/aarch64/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/aarch64/atomics", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/aarch64/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/aarch64", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src/atomics", 0xffffcd71a2c0, 0) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "/workspaces/src/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT\_FDCWD, "/workspaces/src", {st\_mode=S\_IFDIR|0755, st\_size=768, ...}, 0) = 0

openat(AT\_FDCWD, "tls/aarch64/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "atomics/libc.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "libc.so.6", O\_RDONLY|O\_CLOEXEC) = -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=12079, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 12079, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0xffff86cd8000

close(3) = 0

**openat(AT\_FDCWD, "/lib/aarch64-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\267\0\1\0\0\0\340u\2\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=1637400, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 1805928, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffff86aed000

mmap(0xffff86af0000, 1740392, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffff86af0000

munmap(0xffff86aed000, 12288) = 0

munmap(0xffff86c99000, 52840) = 0

mprotect(0xffff86c78000, 61440, PROT\_NONE) = 0

mmap(0xffff86c87000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x187000) = 0xffff86c87000

mmap(0xffff86c8d000, 48744, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0xffff86c8d000

close(3) = 0

set\_tid\_address(0xffff86cdc030) = 3740

set\_robust\_list(0xffff86cdc040, 24) = 0

rseq(0xffff86cdc700, 0x20, 0, 0xd428bc00) = 0

mprotect(0xffff86c87000, 16384, PROT\_READ) = 0

mprotect(0xaaaab1391000, 4096, PROT\_READ) = 0

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffff86cd6000

mprotect(0xffff86ce0000, 8192, PROT\_READ) = 0

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

munmap(0xffff86cd8000, 12079) = 0

getrandom("\xc5\x82\xb7\x70\xfb\x45\x86\xc8", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0xaaaadecd0000

brk(0xaaaadecf1000) = 0xaaaadecf1000

**openat(AT\_FDCWD, "./libgcf.so", O\_RDONLY|O\_CLOEXEC) = 3**

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=7936, ...}, AT\_EMPTY\_PATH) = 0

getcwd("/workspaces/src", 128) = 16

mmap(NULL, 135200, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffff86ace000

mmap(0xffff86ad0000, 69664, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffff86ad0000

munmap(0xffff86ace000, 8192) = 0

munmap(0xffff86ae2000, 53280) = 0

mprotect(0xffff86ad1000, 61440, PROT\_NONE) = 0

mmap(0xffff86ae0000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffff86ae0000

close(3) = 0

mprotect(0xffff86ae0000, 4096, PROT\_READ) = 0

**openat(AT\_FDCWD, "./liblog.so", O\_RDONLY|O\_CLOEXEC) = 3**

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=8184, ...}, AT\_EMPTY\_PATH) = 0

getcwd("/workspaces/src", 128) = 16

mmap(NULL, 135208, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffff86aae000

mmap(0xffff86ab0000, 69672, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffff86ab0000

munmap(0xffff86aae000, 8192) = 0

munmap(0xffff86ac2000, 53288) = 0

mprotect(0xffff86ab1000, 61440, PROT\_NONE) = 0

mmap(0xffff86ac0000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffff86ac0000

close(3) = 0

openat(AT\_FDCWD, "/workspaces/src/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/aarch64/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "tls/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "aarch64/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "atomics/libm.so.6", O\_RDONLY|O\_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT\_FDCWD, "libm.so.6", O\_RDONLY|O\_CLOEXEC) = -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=12079, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 12079, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0xffff86cd8000

close(3) = 0

**openat(AT\_FDCWD, "/lib/aarch64-linux-gnu/libm.so.6", O\_RDONLY|O\_CLOEXEC) = 3**

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=551064, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 680048, PROT\_NONE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0xffff86a09000

mmap(0xffff86a10000, 614512, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0) = 0xffff86a10000

munmap(0xffff86a09000, 28672) = 0

munmap(0xffff86aa7000, 32880) = 0

mprotect(0xffff86a96000, 61440, PROT\_NONE) = 0

mmap(0xffff86aa5000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x85000) = 0xffff86aa5000

close(3) = 0

mprotect(0xffff86aa5000, 4096, PROT\_READ) = 0

mprotect(0xffff86ac0000, 4096, PROT\_READ) = 0

munmap(0xffff86cd8000, 12079) = 0

newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0

write(1, "\320\232\320\276\320\274\320\260\320\275\320\264\321\213:\n", 16) = 16

write(1, "1 <int> <int> \320\264\320\273\321\217 \320\277\320\276\320\264\321\201\321\207\320"..., 46) = 46

write(1, "2 <int> \320\264\320\273\321\217 \320\277\320\276\320\264\321\201\321\207\320\265\321\202\320\260 "..., 35) = 35

write(1, "q \320\264\320\273\321\217 \320\262\321\213\321\205\320\276\320\264\320\260.\n", 23) = 23

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

newfstatat(0, "", {st\_mode=S\_IFREG|0644, st\_size=18, ...}, AT\_EMPTY\_PATH) = 0

read(0, "1 123 456\n2 9999\nq", 4096) = 18

write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265 \320\272\320\276\320\274\320\260\320\275\320\264\321\203: \320"..., 67) = 67

write(1, "\320\235\320\236\320\224 (123, 456) - \320\235\320\260\320\270\320\262\320\275\321\213"..., 38) = 38

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

write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265 \320\272\320\276\320\274\320\260\320\275\320\264\321\203: \320"..., 83) = 83

write(1, "\320\247\320\270\321\201\320\273\320\276 e (9999) - \320\277\320\276 \321\201\321\203\320"..., 59) = 59

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

read(0, "", 4096) = 0

write(1, "\320\222\320\262\320\265\320\264\320\270\321\202\320\265 \320\272\320\276\320\274\320\260\320\275\320\264\321\203: \n", 32) = 32

munmap(0xffff86ad0000, 69664) = 0

munmap(0xffff86ab0000, 69672) = 0

munmap(0xffff86a10000, 614512) = 0

exit\_group(0) = ?

+++ exited with 0 +++

**Вывод**

Оба примера используют динамические библиотеки (\*.so), которые загружаются в память во время выполнения, что снижает размер программ и позволяет обновлять библиотеки без перекомпиляции. В program2 реализована явная динамическая загрузка через dlopen, что обеспечивает гибкость (например, подгрузку библиотек по требованию), тогда как program1 линкуется с библиотеками на этапе компиляции. Динамические библиотеки экономят память и дисковое пространство, так как разделяются между программами, но могут добавлять накладные расходы на загрузку и разрешение зависимостей. Статические библиотеки увеличивают размер исполняемого файла, но обеспечивают быстрый старт и независимость от окружения. Выбор зависит от приоритетов: экономия ресурсов → динамические, скорость запуска → статические.