

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-209БВ-24

Студент: Крюков Д.М.

Преподаватель: Миронов Е.С.

Оценка: _____

Дата: 13.12.25

Москва, 2025

Постановка задачи

Вариант 24.

Требуется создать динамические библиотеки, которые реализуют заданный вариантом функционал. Далее использовать данные библиотеки 2-мя способами:

1. Во время компиляции (на этапе «линковки»/linking)
2. Во время исполнения программы. Библиотеки загружаются в память с помощью интерфейса ОС для работы с динамическими библиотеками

В конечном итоге, в лабораторной работе необходимо получить следующие части:

- Динамические библиотеки, реализующие контракты, которые заданы вариантом;
- Тестовая программа (*программа №1*), которая используют одну из библиотек, используя информацию полученные на этапе компиляции;
- Тестовая программа (*программа №2*), которая загружает библиотеки, используя только их относительные пути и контракты.

№	Описание	Сигнатура	Реализация 1	Реализация 2
1	Подсчёт наибольшего общего делителя для двух натуральных чисел	Int GCF(int A, int B)	Алгоритм Евклида	Наивный алгоритм. Пытаться разделить числа на все числа, что меньше A и B.

2	Подсчет площади плоской геометрической фигуры по двум сторонам	Float Square(float A, float B)	Фигура прямоугольник	Фигура прямоугольный треугольник
---	---	--------------------------------------	-------------------------	--

Общий метод и алгоритм решения

Использованные системные вызовы

- `void* dlopen(const char* filename, int flag);` - загрузка библиотеки в память
- `int dlclose(void* handle);` - выгрузка библиотеки из памяти
- `void* dlsym(void* handle, const char* symbol);` - получение адреса функции
- `char* dlerror(void);` - получение строки ошибки

Алгоритм работы:

Программа является интерактивным консольным приложением, поддерживающим два режима работы:

- **Статический режим:** функции `GCF` и `Square` линкуются на этапе сборки.
- **Динамический режим:** библиотеки загружаются во время выполнения, возможна смена реализации без перекомпиляции.

Программа работает в бесконечном цикле, обрабатывая команды:

- 0 – переключение между библиотеками (только в динамическом режиме)
- 1 – вычисление НОД для двух целых чисел
- 2 – вычисление площади для двух вещественных чисел

Код программы

`calc.h`

```
#ifndef CALC_H
```

```
#define CALC_H
```

```
int GCF(int A, int B);
```

```
float Square(float A, float B);
```

```
#endif
```

lib1.c

```
#include "calc.h"
```

```
int GCF(int A, int B) { // алгоритм Евклида
```

```
    while (B != 0) {
```

```
        int tmp = A % B;
```

```
        A = B;
```

```
        B = tmp;
```

```
    }
```

```
    return A;
```

```
}
```

```
float Square(float A, float B) { // площадь прямоугольника
```

```
    return A * B;
```

```
}
```

lib2.c

```
#include "calc.h"
```

```
int GCF(int A, int B) { // наивный алгоритм
```

```
    int min = A < B ? A : B;
```

```
    for (int i = min; i >= 1; --i) {
```

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

```
            return i;
```

```
        }
```

```
    }
```

```
    return 1;
```

```
}
```

```
float Square(float A, float B) { // площадь прямоугольного треугольника  
    return (A * B) / 2.0f;  
}
```

prog_dynamic.c

```
#include <stdio.h>
```

```
#include <dlfcn.h>
```

```
#include "calc.h"
```

```
typedef int (*gcf_func)(int, int);
```

```
typedef float (*square_func)(float, float);
```

```
int main() {
```

```
    void* handle = NULL;
```

```
    gcf_func GCF_dyn = NULL;
```

```
    square_func Square_dyn = NULL;
```

```
    int current = 1;
```

```
    handle = dlopen("./libcalc1.so", RTLD_LAZY);
```

```
    GCF_dyn = (gcf_func)dlsym(handle, "GCF");
```

```
    Square_dyn = (square_func)dlsym(handle, "Square");
```

```
    int cmd;
```

```
    while (1) {
```

```
        if (scanf("%d", &cmd) != 1)
```

```
            break;
```

```
        if (cmd == 0) {
```

```
            dlclose(handle);
```

```
            if (current == 1) {
```

```
                handle = dlopen("./libcalc2.so", RTLD_LAZY);
```

```
                current = 2;
```

```

        printf("переключено на 2ую реализацию\n");

    } else {

        handle = dlopen("./libcalc1.so", RTLD_LAZY);

        current = 1;

        printf("переключено на 1ую реализацию\n");

    }

    GCF_dyn = (gcf_func)dlsym(handle, "GCF");

    Square_dyn = (square_func)dlsym(handle, "Square");

}

else if (cmd == 1) {

    int a, b;

    scanf("%d %d", &a, &b);

    printf("НОД = %d\n", GCF_dyn(a, b));

}

else if (cmd == 2) {

    float a, b;

    scanf("%f %f", &a, &b);

    printf("площадь = %.2f\n", Square_dyn(a, b));

}

}

dlclose(handle);

return 0;

}

```

prog_static.c

```

#include <stdio.h>

#include "calc.h"

```

```

int main() {

```

```

int cmd;

while (1) {
    if (scanf("%d", &cmd) != 1)
        break;
    if (cmd == 1) {
        int a, b;
        scanf("%d %d", &a, &b);
        printf("НОД = %d\n", GCF(a, b));
    }
    else if (cmd == 2) {
        float a, b;
        scanf("%f %f", &a, &b);
        printf("площадь = %.2f\n", Square(a, b));
    }
}
return 0;
}

```

Протокол работы программы

Тестирование:

./prog_static

1 48 18

НОД = 6

2 4 5

площадь = 10.00

./prog_dynamic

1 48 18

НОД = 6

2 4 5

площадь = 20.00

0

переключено на 2ую реализацию

1 48 18

НОД = 6

2 4 5

площадь = 10.00

Strace:

prog_static.c

execve("./prog_static", ["/prog_static"], 0x7ffed8ccad28 /* 30 vars */) = 0

brk(NULL) = 0x3b59d000

**mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x72f7b2c73000**

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

**openat(AT_FDCWD, "/glibc-hwcaps/x86-64-v3/libcalc2.so",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)**

**openat(AT_FDCWD, "/glibc-hwcaps/x86-64-v2/libcalc2.so",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)**

openat(AT_FDCWD, "./libcalc2.so", O_RDONLY|O_CLOEXEC) = 3

**read(3, "\177ELF\2\1\1\0\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|0755, st_size=15184, ...}) = 0

getcwd("/workspace/lab4/src", 128) = 20

**mmap(NULL, 16400, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE,
3, 0) = 0x72f7b2c6e000**

**mmap(0x72f7b2c6f000, 4096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x72f7b2c6f000**

**mmap(0x72f7b2c70000, 4096, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x72f7b2c70000**

**mmap(0x72f7b2c71000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x72f7b2c71000**

close(3) = 0

**openat(AT_FDCWD, "/glibc-hwcaps/x86-64-v3/libc.so.6",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)**


```

    openat(AT_FDCWD, "./glibc-hwcaps/x86-64-v2/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

    fstat(3, {st_mode=S_IFREG|0644, st_size=29003, ...}) = 0

    mmap(NULL, 29003, PROT_READ, MAP_PRIVATE, 3, 0) = 0x72f7b2c66000

    close(3) = 0

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

    read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0p\236\2\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"... ,
840, 64) = 840

    fstat(3, {st_mode=S_IFREG|0755, st_size=2003408, ...}) = 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"... ,
840, 64) = 840

    mmap(NULL, 2055800, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3,
0) = 0x72f7b2a70000

    mmap(0x72f7b2a98000, 1462272, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x72f7b2a98000

    mmap(0x72f7b2bfd000, 352256, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18d000) = 0x72f7b2bfd000

    mmap(0x72f7b2c53000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e2000) = 0x72f7b2c53000

    mmap(0x72f7b2c59000, 52856, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x72f7b2c59000

    close(3) = 0

    mmap(NULL, 12288, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x72f7b2a6d000

    arch_prctl(ARCH_SET_FS, 0x72f7b2a6d740) = 0

    set_tid_address(0x72f7b2a6da10) = 5922

    set_robust_list(0x72f7b2a6da20, 24) = 0

```

```

rseq(0x72f7b2a6d680, 0x20, 0, 0x53053053) = 0
mprotect(0x72f7b2c53000, 16384, PROT_READ) = 0
mprotect(0x72f7b2c71000, 4096, PROT_READ) = 0
mprotect(0x403000, 4096, PROT_READ) = 0
mprotect(0x72f7b2ca9000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
munmap(0x72f7b2c66000, 29003) = 0
fstat(0, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
getrandom("\xa1\x07\x1f\x45\x15\xca\x3b\xe1", 8, GRND_NONBLOCK) = 8
brk(NULL) = 0x3b59d000
brk(0x3b5be000) = 0x3b5be000
read(0, 1 48 18
"1 48 18\n", 1024) = 8
fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
write(1, "\320\235\320\236\320\224 = 6\n", 11НОД = 6
) = 11
read(0, 2 4 5
"2 4 5\n", 1024) = 6
write(1, "\320\277\320\273\320\276\321\211\320\260\320\264\321\214 =
10.00\n", 23площадь = 10.00
) = 23
read(0, "", 1024) = 0
exit_group(0) = ?
+++ exited with 0 +++
prog_dynamic.c
execve("./prog_dynamic", ["/prog_dynamic"], 0x7ffedd7138f8 /* 30 vars */) = 0
brk(NULL) = 0x20e4a000
mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f87d51bb000
access("/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)

```

```

    openat(AT_FDCWD,                                "./glibc-hwcaps/x86-64-v3/libc.so.6",
O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

    openat(AT_FDCWD,                                "./glibc-hwcaps/x86-64-v2/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

    fstat(3, {st_mode=S_IFREG|0644, st_size=29003, ...}) = 0

    mmap(NULL, 29003, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f87d51b3000

    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\0p\236\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"..., 840,
64) = 840

    fstat(3, {st_mode=S_IFREG|0755, st_size=2003408, ...}) = 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"..., 840,
64) = 840

    mmap(NULL, 2055800, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3,
0) = 0x7f87d4fbd000

    mmap(0x7f87d4fe5000,                            1462272,                PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x7f87d4fe5000

    mmap(0x7f87d514a000,                            352256,                    PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18d000) = 0x7f87d514a000

    mmap(0x7f87d51a0000,                            24576,                      PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e2000) = 0x7f87d51a0000

    mmap(0x7f87d51a6000,                            52856,                      PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f87d51a6000

    close(3)                                         = 0

    mmap(NULL,                                        12288,                      PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f87d4fba000

    arch_prctl(ARCH_SET_FS, 0x7f87d4fba740) = 0

    set_tid_address(0x7f87d4fbaa10)                = 10807

```

```

set_robust_list(0x7f87d4fbbaa20, 24)    = 0
rseq(0x7f87d4fba680, 0x20, 0, 0x53053053) = 0
mprotect(0x7f87d51a0000, 16384, PROT_READ) = 0
mprotect(0x403000, 4096, PROT_READ)    = 0
mprotect(0x7f87d51f1000, 8192, PROT_READ) = 0
prlimit64(0,          RLIMIT_STACK,          NULL,          {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7f87d51b3000, 29003)            = 0
getrandom("\x03\x59\x00\xaa\xec\x55\x2f\xde", 8, GRND_NONBLOCK) = 8
brk(NULL)                                = 0x20e4a000
brk(0x20e6b000)                          = 0x20e6b000
openat(AT_FDCWD, "./libcalc1.so", 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|0755, st_size=15112, ...}) = 0
getcwd("/workspace/lab4/src", 128)      = 20
mmap(NULL, 16400, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3,
0) = 0x7f87d51b6000
mmap(0x7f87d51b7000,          4096,          PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f87d51b7000
mmap(0x7f87d51b8000,          4096,          PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f87d51b8000
mmap(0x7f87d51b9000,          8192,          PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f87d51b9000
close(3)                                = 0
mprotect(0x7f87d51b9000, 4096, PROT_READ) = 0
fstat(0, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
read(0, 1 48 18
"1 48 18\n", 1024)                    = 8
fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0), ...}) = 0
write(1, "\320\235\320\236\320\224 = 6\n", 11HOД = 6
) = 11

```

```

read(0, 2 4 5
"2 4 5\n", 1024)          = 6
write(1,      "\320\277\320\273\320\276\321\211\320\260\320\264\321\214      =
20.00\n", 23площадь = 20.00
) = 23
read(0, 0
"0\n", 1024)              = 2
munmap(0x7f87d51b6000, 16400)      = 0
openat(AT_FDCWD, "./libcalc2.so", 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|0755, st_size=15184, ...}) = 0
getcwd("/workspace/lab4/src", 128)    = 20
mmap(NULL, 16400, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3,
0) = 0x7f87d51b6000
mmap(0x7f87d51b7000,      4096,      PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f87d51b7000
mmap(0x7f87d51b8000,      4096,      PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f87d51b8000
mmap(0x7f87d51b9000,      8192,      PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f87d51b9000
close(3)                  = 0
mprotect(0x7f87d51b9000, 4096, PROT_READ) = 0
write(1,
"\320\277\320\265\321\200\320\265\320\272\320\273\321\216\321\207\320\265\320\27
5\320\276 \320\275\320\260 2\321\203\321"..., 55переключено на 2ую реализацию
) = 55
read(0, 1 48 18
"1 48 18\n", 1024)        = 8
write(1, "\320\235\320\236\320\224 = 6\n", 11НОД = 6
) = 11
read(0, 2 4 5

```

```
"2 4 5\n", 1024)          = 6
```

```
write(1,      "\320\277\320\273\320\276\321\211\320\260\320\264\321\214  
10.00\n", 23)площадь = 10.00
```

```
) = 23
```

```
read(0, "", 1024)          = 0
```

```
munmap(0x7f87d51b6000, 16400)      = 0
```

```
exit_group(0)                = ?
```

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

Вывод

В ходе выполнения лабораторной работы были созданы две динамические библиотеки, реализующие различные алгоритмы вычисления НОД и площади фигуры. Были разработаны две тестовые программы:

1. prog_static – использует статическую линковку библиотек.
2. prog_dynamic – загружает библиотеки во время выполнения с возможностью их переключения.

Программа с динамической загрузкой демонстрирует гибкость и возможность обновления функционала без перекомпиляции основного приложения. В процессе работы были изучены системные вызовы dlopen, dlclose, dlsym, а также различия между статической и динамической линковкой.