**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ  
ФЕДЕРАЦИИ МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ  
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)**

**ЛАБОРАТОРНАЯ РАБОТА №5**

**по курсу операционные системы I семестр, 2021/22  
уч. год**

Студент: Белоусов Егор Владимирович

Группа: М8О-207Б-20

Преподаватель: Миронов Е. С.

Вариант: 16

Дата:

Оценка:

**Тема**

Динамические библиотеки

Цель работы

Целью является приобретение практических навыков в:

· Создание динамических библиотек

· Создание программ, которые используют функции динамических библиотек

**Задание**

Требуется создать динамические библиотеки, которые реализуют определенный 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 знания полученные на этапе компиляции;
* Тестовая программа (программа №2), которая загружает библиотеки, используя только их местоположение и контракты.

Провести анализ двух типов использования библиотек.

Пользовательский ввод для обоих программ должен быть организован следующим образом:

1. Если пользователь вводит команду «0», то программа переключает одну реализацию контрактов на другую (необходимо только для программы №2). Можно реализовать лабораторную работу без данной функции, но максимальная оценка в этом случае будет «хорошо»;

2. «1 arg1 arg2 … argN», где после «1» идут аргументы для первой функции, предусмотренной контрактами. После ввода команды происходит вызов первой функции, и на экране появляется результат её выполнения;

3. «2 arg1 arg2 … argM», где после «2» идут аргументы для второй функции, предусмотренной контрактами. После ввода команды происходит вызов второй функции, и на экране появляется результат её выполнения.

**Вариант 16:**

3. Подсчет количества простых чисел на отрезке [A, B] (A, B – натуральные).

4. Подсчет наибольшего общего делителя для двух натуральных чисел.

Для решения поставленной задачи используются следующие функции:

**void \*dlclose(const char\* filename, int flag) –** загружает динамическую библиотеку, имя которой указанно в строке filename и возвращает прямой указатель на начало загруженной библиотеки.

**void \*dlsym(void\* handle, char\* symbol) –** возвращает указатель на область памяти, в которой содержится необходимый **symbol** в динамической библиотеке по указателю **handle**.

**int dlclose(void\* handle) –** уменьшает счетчик ссылок на указатель **handle** и если он равен 0, то освобождает библиотеку.

**Пример работы**

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ gcc -fPIC -shared lib1.c -o lib1.so*

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ gcc -fPIC -shared lib2.c -o lib2.so*

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ gcc static.c -o static.o -ldl*

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ gcc dynamic.c -o dynamic -ldl*

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ ./static*

*Enter a command: 1 - PrimeCount*

*2 - GCF*

*1*

*Enter values for PrimeCount: 1 1000*

*168*

*Enter a command: 1 - PrimeCount*

*2 - GCF*

*2*

*Enter values for GCF: 200 500*

*100*

*Enter a command: 1 - PrimeCount*

*2 - GCF*

*2*

*Enter values for GCF: 123*

*321*

*3*

*Enter a command: 1 - PrimeCount*

*2 - GCF*

*2*

*Enter values for GCF: 560 561*

*1*

*Enter a command: 1 - PrimeCount*

*2 - GCF*

*1*

*Enter values for PrimeCount: 100 250*

*28*

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ ./dynamic*

*Enter a command: 0 - swap lib*

*1 - PrimeCount*

*2 - GCF*

*1*

*Enter values for PrimeCount: 1 1000*

*168*

*Enter a command: 0 - swap lib*

*1 - PrimeCount*

*2 - GCF*

*0*

*Swapped! Current lib is: ./lib2.so*

*Enter a command: 0 - swap lib*

*1 - PrimeCount*

*2 - GCF*

*1*

*Enter values for PrimeCount: 1 10000000*

*664579*

*Enter a command: 0 - swap lib*

*1 - PrimeCount*

*2 - GCF*

*2*

*Enter values for GCF: 5 100*

*5*

*Enter a command: 0 - swap lib*

*1 - PrimeCount*

*2 - GCF*

*0*

*Swapped! Current lib is: ./lib1.so*

*Enter a command: 0 - swap lib*

*1 - PrimeCount*

*2 – GCF*

**Исходный код:**

**lib.h**

#pragma once

int PrimeCount(int A,int B);

int GCF(int A, int B);

**lib1.c**

#include "lib.h"

int PrimeCount(int A, int B){

    int result = 0;

    for(int i = A;i <= B; ++i){

        if(i == 1)continue;

        int ok = 1;

        for(int j = 2;j < i; ++j){

            if(i % j == 0){

                ok = 0;

                break;

            }

        }

        if(ok)++result;

    }

    return result;

}

int GCF(int A, int B){

    while(A != 0){

        B %= A;

        int temp = A;

        A = B;

        B = temp;

    }

    return B;

}

**lib2.c**

#include "lib.h"

#include "stdlib.h"

int PrimeCount(int A, int B){

    int \* sieve = (int \*)(calloc(B + 1, sizeof(int)));

    sieve[1] = sieve[0] = 1;

    for(int i = 2; i \* i <= B; ++i){

        if(sieve[i])continue;

        for(int j = i \* i;j <= B; j += i){

            sieve[j] = 1;

        }

    }

    int result = 0;

    for(int i = A; i <= B; ++i){

        if(sieve[i] == 0){

            ++result;

        }

    }

    free(sieve);

    return result;

}

int GCF(int A, int B){

    if(A > B){

        int temp = A;

        A = B;

        B = temp;

    }

    int result = 0;

    for(int i = 1; i <= B;++i){

        if(B % i == 0 && A % i == 0){

            result = i;

        }

    }

    return result;

}

**static.c**

#include "lib1.c"

#include "stdio.h"

void menu(){

    printf("Enter a command: 1 - PrimeCount\n2 - GCF\n");

}

int main(){

    int c;

    menu();

    while(scanf("%d",&c) != EOF){

        if(c == 1){

            int A, B;

            printf("Enter values for PrimeCount: ");

            scanf("%d %d", &A, &B);

            printf("%d\n", PrimeCount(A, B));

        }

        else if(c == 2){

            int A, B;

            printf("Enter values for GCF: ");

            scanf("%d %d", &A, &B);

            printf("%d\n", GCF(A, B));

        }

        menu();

    }

    return 0;

}

**dynamic.c**

#include "dlfcn.h"

#include "stdio.h"

#include "stdlib.h"

int current\_lib = 0;

const char\* lib1 = "./lib1.so";

const char\* lib2 = "./lib2.so";

void \*handle = NULL;

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

int (\*PrimeCount)(int, int);

void load(){

    if(current\_lib == 0){

        handle = dlopen(lib1, RTLD\_LAZY);

    }

    else {

        handle = dlopen(lib2, RTLD\_LAZY);

    }

    if(handle == 0){

        printf("Error with loading library!\n");

        exit(EXIT\_FAILURE);

    }

    PrimeCount = dlsym(handle, "PrimeCount");

    GCF = dlsym(handle, "GCF");

}

void unload(){

    dlclose(handle);

}

void swap\_lib(){

    unload();

    if(current\_lib == 0){

        current\_lib = 1;

    }

    else current\_lib = 0;

    load();

}

void menu(){

    printf("Enter a command: 0 - swap lib\n1 - PrimeCount\n2 - GCF\n");

}

int main(){

    load();

    int c;

    menu();

    while(scanf("%d", &c) != EOF){

        if(c == 0){

            swap\_lib();

            const char \* name = current\_lib? lib2 : lib1;

            printf("Swapped! Current lib is: %s\n", name);

        }

        else if(c == 1){

            int A, B;

            printf("Enter values for PrimeCount: ");

            scanf("%d %d", &A, &B);

            printf("%d\n", PrimeCount(A, B));

        }

        else{

            int A, B;

            printf("Enter values for GCF: ");

            scanf("%d %d", &A, &B);

            printf("%d\n", GCF(A, B));

        }

        menu();

    }

}

**Strace**

*egorb@LAPTOP-TIFMO0IA:/mnt/d/os/lab5$ strace ./dynamic*

*execve("./dynamic", ["./dynamic"], 0x7ffcf72cb610 /\* 29 vars \*/) = 0*

*brk(NULL) = 0x555b4dbaa000*

*arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7fffc378b500) = -1 EINVAL (Invalid argument)*

*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=36953, ...}) = 0*

*mmap(NULL, 36953, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f5dc6b0f000*

*close(3) = 0*

*openat(AT\_FDCWD, "/lib/x86\_64-linux-gnu/libdl.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 \22\0\0\0\0\0\0"..., 832) = 832*

*fstat(3, {st\_mode=S\_IFREG|0644, st\_size=18816, ...}) = 0*

*mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f5dc6b0d000*

*mmap(NULL, 20752, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f5dc6b07000*

*mmap(0x7f5dc6b08000, 8192, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f5dc6b08000*

*mmap(0x7f5dc6b0a000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x3000) = 0x7f5dc6b0a000*

*mmap(0x7f5dc6b0b000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x3000) = 0x7f5dc6b0b000*

*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\360q\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*

*pread64(3, "\4\0\0\0\20\0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0", 32, 848) = 32*

*pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\t\233\222%\274\260\320\31\331\326\10\204\276X>\263"..., 68, 880) = 68*

*fstat(3, {st\_mode=S\_IFREG|0755, st\_size=2029224, ...}) = 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*

*pread64(3, "\4\0\0\0\20\0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0", 32, 848) = 32*

*pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\t\233\222%\274\260\320\31\331\326\10\204\276X>\263"..., 68, 880) = 68*

*mmap(NULL, 2036952, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f5dc6915000*

*mprotect(0x7f5dc693a000, 1847296, PROT\_NONE) = 0*

*mmap(0x7f5dc693a000, 1540096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x25000) = 0x7f5dc693a000*

*mmap(0x7f5dc6ab2000, 303104, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x19d000) = 0x7f5dc6ab2000*

*mmap(0x7f5dc6afd000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e7000) = 0x7f5dc6afd000*

*mmap(0x7f5dc6b03000, 13528, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f5dc6b03000*

*close(3) = 0*

*mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f5dc6912000*

*arch\_prctl(ARCH\_SET\_FS, 0x7f5dc6912740) = 0*

*mprotect(0x7f5dc6afd000, 12288, PROT\_READ) = 0*

*mprotect(0x7f5dc6b0b000, 4096, PROT\_READ) = 0*

*mprotect(0x555b4cfeb000, 4096, PROT\_READ) = 0*

*mprotect(0x7f5dc6b46000, 4096, PROT\_READ) = 0*

*munmap(0x7f5dc6b0f000, 36953) = 0*

*brk(NULL) = 0x555b4dbaa000*

*brk(0x555b4dbcb000) = 0x555b4dbcb000*

*openat(AT\_FDCWD, "./lib1.so", 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@\20\0\0\0\0\0\0"..., 832) = 832*

*fstat(3, {st\_mode=S\_IFREG|0777, st\_size=15672, ...}) = 0*

*getcwd("/mnt/d/os/lab5", 128) = 15*

*mmap(NULL, 16424, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f5dc6b14000*

*mmap(0x7f5dc6b15000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f5dc6b15000*

*mmap(0x7f5dc6b16000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b16000*

*mmap(0x7f5dc6b17000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b17000*

*close(3) = 0*

*mprotect(0x7f5dc6b17000, 4096, PROT\_READ) = 0*

*fstat(1, {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0x4), ...}) = 0*

*write(1, "Enter a command: 0 - swap lib\n", 30Enter a command: 0 - swap lib*

*) = 30*

*write(1, "1 - PrimeCount\n", 151 - PrimeCount*

*) = 15*

*write(1, "2 - GCF\n", 82 - GCF*

*) = 8*

*fstat(0, {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0x4), ...}) = 0*

*read(0, 0x555b4dbaac50, 1024) = ? ERESTARTSYS (To be restarted if SA\_RESTART is set)*

*--- SIGWINCH {si\_signo=SIGWINCH, si\_code=SI\_KERNEL} ---*

*read(0, 0*

*"0\n", 1024) = 2*

*munmap(0x7f5dc6b14000, 16424) = 0*

*openat(AT\_FDCWD, "./lib2.so", 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\200\20\0\0\0\0\0\0"..., 832) = 832*

*fstat(3, {st\_mode=S\_IFREG|0777, st\_size=16184, ...}) = 0*

*getcwd("/mnt/d/os/lab5", 128) = 15*

*mmap(NULL, 16440, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f5dc6b14000*

*mmap(0x7f5dc6b15000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f5dc6b15000*

*mmap(0x7f5dc6b16000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b16000*

*mmap(0x7f5dc6b17000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b17000*

*close(3) = 0*

*mprotect(0x7f5dc6b17000, 4096, PROT\_READ) = 0*

*write(1, "Swapped! Current lib is: ./lib2."..., 35Swapped! Current lib is: ./lib2.so*

*) = 35*

*write(1, "Enter a command: 0 - swap lib\n1 "..., 45Enter a command: 0 - swap lib*

*1 - PrimeCount*

*) = 45*

*write(1, "2 - GCF\n", 82 - GCF*

*) = 8*

*read(0, 0*

*"0\n", 1024) = 2*

*munmap(0x7f5dc6b14000, 16440) = 0*

*openat(AT\_FDCWD, "./lib1.so", 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@\20\0\0\0\0\0\0"..., 832) = 832*

*fstat(3, {st\_mode=S\_IFREG|0777, st\_size=15672, ...}) = 0*

*getcwd("/mnt/d/os/lab5", 128) = 15*

*mmap(NULL, 16424, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f5dc6b14000*

*mmap(0x7f5dc6b15000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f5dc6b15000*

*mmap(0x7f5dc6b16000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b16000*

*mmap(0x7f5dc6b17000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b17000*

*close(3) = 0*

*mprotect(0x7f5dc6b17000, 4096, PROT\_READ) = 0*

*write(1, "Swapped! Current lib is: ./lib1."..., 35Swapped! Current lib is: ./lib1.so*

*) = 35*

*write(1, "Enter a command: 0 - swap lib\n1 "..., 45Enter a command: 0 - swap lib*

*1 - PrimeCount*

*) = 45*

*write(1, "2 - GCF\n", 82 - GCF*

*) = 8*

*read(0, 1*

*"1\n", 1024) = 2*

*write(1, "Enter values for PrimeCount: ", 29Enter values for PrimeCount: ) = 29*

*read(0, 1 1000*

*"1 1000\n", 1024) = 7*

*write(1, "168\n", 4168*

*) = 4*

*write(1, "Enter a command: 0 - swap lib\n1 "..., 45Enter a command: 0 - swap lib*

*1 - PrimeCount*

*) = 45*

*write(1, "2 - GCF\n", 82 - GCF*

*) = 8*

*read(0, 0*

*"0\n", 1024) = 2*

*munmap(0x7f5dc6b14000, 16424) = 0*

*openat(AT\_FDCWD, "./lib2.so", 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\200\20\0\0\0\0\0\0"..., 832) = 832*

*fstat(3, {st\_mode=S\_IFREG|0777, st\_size=16184, ...}) = 0*

*getcwd("/mnt/d/os/lab5", 128) = 15*

*mmap(NULL, 16440, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f5dc6b14000*

*mmap(0x7f5dc6b15000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f5dc6b15000*

*mmap(0x7f5dc6b16000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b16000*

*mmap(0x7f5dc6b17000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f5dc6b17000*

*close(3) = 0*

*mprotect(0x7f5dc6b17000, 4096, PROT\_READ) = 0*

*write(1, "Swapped! Current lib is: ./lib2."..., 35Swapped! Current lib is: ./lib2.so*

*) = 35*

*write(1, "Enter a command: 0 - swap lib\n1 "..., 45Enter a command: 0 - swap lib*

*1 - PrimeCount*

*) = 45*

*write(1, "2 - GCF\n", 82 - GCF*

*) = 8*

*read(0, 2*

*"2\n", 1024) = 2*

*write(1, "Enter values for GCF: ", 22Enter values for GCF: ) = 22*

*read(0, 1 50*

*"1 50\n", 1024) = 5*

*write(1, "1\n", 21*

*) = 2*

*write(1, "Enter a command: 0 - swap lib\n1 "..., 45Enter a command: 0 - swap lib*

*1 - PrimeCount*

*) = 45*

*write(1, "2 - GCF\n", 82 - GCF*

*) = 8*

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

*exit\_group(0) = ?*

*+++ exited with 0 +++*

**Вывод**

В данной лабораторной работе я освежил свои знания по работе с статическими библиотеками, а также познакомился с таким новым понятием, как динамические библиотеки. Я реализовал две динамические библиотеки с вспомогательными функциями и использовал их в своей программе на ОС Linux. Использование библиотек разбивает весь код на множество модулей, что сильно упрощает работу с хоть немного большими проектами.