



Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Лабораторная работа №5 по курсу

«Операционные системы»

Группа: М80-207Б-20

Студент: Мерц С.П.

Вариант: 16

Преподаватель: Миронов Е.С.

Оценка: _____

Дата: 20.12.21

Москва, 2021.

Содержание

- 1 Постановка задачи.
- 2 Общие сведения о программе.
- 3 Общий метод и алгоритм решения.
- 4 Код программы.
- 5 Демонстрация работы программы.
- 6 Вывод.

Постановка задачи

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 использует 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 обеих 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:

Составить и отладить программу на языке C++ с использованием дин. библиотек, со следующими контактами и функциями.

3	Подсчёт количества простых чисел на отрезке [A, B] (A, B - натуральные)	Int PrimeCount(int A, int B)	Наивный алгоритм. Проверить делимость текущего числа на все предыдущие числа.	Решето Эратосфена
4	Подсчёт наибольшего общего делителя для двух натуральных чисел	Int GCF(int A, int B)	Алгоритм Евклида	Наивный алгоритм. Попытаться разделить числа на все числа, что меньше A и B.

Общие сведения о программе

Программа состоит из четырех файлов compilation.cpp, launch.cpp, realization1.cpp, realization2.cpp, где realization1.cpp и realization2.cpp – реализация функций первого и второго типа соответственно, а compilation.cpp – программа, которая использует одну из библиотек, используя знания полученные на этапе компиляции, а launch.cpp – программа, которая загружает библиотеки, используя только их местоположение и контракты

Программы используют следующие системные вызовы:

1 dlopen - загружает динамическую библиотеку, имя которой мы передаем, и возвращает прямой указатель на начало динамической библиотеки.

2 dlclose уменьшает на единицу счетчик ссылок на указатель динамической библиотеки. Если нет других загруженных библиотек, использующих ее символы и если счетчик ссылок принимает нулевое значение, то динамическая библиотека выгружается.

3 dlsym использует указатель на динамическую библиотеку, возвращаемую dlopen, и оканчивающееся нулем символьное имя, а затем возвращает адрес, указывающий, откуда загружается этот символ.

Общий метод и алгоритм решения

Реализуем заданные варианты функций в файлах. Затем скомпилируем файлы с данными функциями в объектные файлы:

```
g++ -fPIC -c realization1.cpp -o d1.o
```

```
g++ -fPIC -c realization2.cpp -o d2.o
```

флаг -fPIC код означает, чтобы сгенерированный машинный код не зависит от того, чтобы он работал по определенному адресу. Затем, преобразуем объектные файлы в динамические библиотеки с помощью следующего кода:

```
g++ -shared -o libd1.so d1.o
```

```
g++ -shared -o libd2.so d2.o
```

флаг -shared для создания динамическую (т.е. "разделяемой") библиотеки.

И для компиляции самих файлов используем следующие команды:

```
g++ launch.cpp -L. -ld1 -o main2.out -Wl,-rpath -Wl,.
```

```
g++ compilation.cpp -L. -ld1 -o main1.out -Wl,-rpath -Wl,.
```

Флаг -L. для указания пути к каталогу, содержащему наши библиотеки, «.» так как используется файл, который лежит в данной папке. Флаг -Wl передает 'опцию' в качестве опции линкеру, если 'опция' содержит запятые, она расщепляется запятыми на многочисленные опции.

Код программы

compilation.cpp:

```
#include <iostream>
```

```
extern "C" int PrimeCount (int A, int B);
```

```
extern "C" int GCF (int A, int B);
```

```
int main () {
```

```
    int command;
```

```
    while (printf("Please enter your command: ") && (scanf("%d", &command)) != EOF) {
```

```
        switch (command) {
```

```

case 1: {
    int A, B;

    std::cout << "enter A and B\n";

    std::cin >> A >> B;

    int prime_count = PrimeCount (A, B);

    printf("PrimeCount = %d\n", prime_count);

    break;
}
case 2: {
    int A, B;

    std::cout << "enter A and B\n";

    std::cin >> A >> B;

    int gcf = GCF (A, B);

    printf("GCF = %d\n", gcf);

    break;
}
default:

    std::cout << "You must enter 1 or 2!\n" << std::endl;

    break;
}
}

return 0;
}

```

launch.cpp:

```

#include <stdio.h>

#include <iostream>

#include <stdlib.h>

#include <dlfcn.h>

```

```

int main () {

    void* handle = NULL; //адрес, в будущем нужный нам для получения доступа к библиотеке

    int (*GCF)(int A, int B); //объявление указателей на функции

    int (*PrimeCount)(int A, int B); // объявление указателей на функции

    const char* lib_array[] = {"libd1.so", "libd2.so"};

    int curlib;

    int start_library;

    std::cout << "Enter start library: " << std::endl;

    std::cout << "\t" << "1 for using first library" << std::endl;

    std::cout << "\t" << "2 for using second library" << std::endl;

    std::cin >> start_library;

    bool flag = true;

    while (flag) {

        if (start_library == 1) {

            curlib = 0;

            flag = false;

        }

        else if (start_library == 2) {

            curlib = 1;

            flag = false;

        }

        else {

            std::cout << "You should enter only 1 or 2!" << std::endl;

            std::cin >> start_library;

        }

    }

    handle = dlopen(lib_array[curlib], RTLD_LAZY); //rtld lazy выполняется поиск только тех
символов, на которые есть ссылки из кода

    if (!handle) {

```

```

std:: cout << "An error while opening library has been detected" << std:: endl;

exit(EXIT_FAILURE);

}

GCF = (int (*)(int, int))dlsym(handle, "GCF"); //возвращаем адрес функции из памяти библиотеки

PrimeCount = (int (*)(int, int))dlsym(handle, "PrimeCount"); //dlsym присваивает указателю на
функцию, объявленному в начале, ее адрес в библиотеке

int command;

std:: cout << "Hello there! Please enter your command according to next rules: " << std:: endl;

std:: cout << '\t' << "0 for changing the contract;" << std:: endl;

std:: cout << '\t' << "1 for calculating the PrimeCount; " << std:: endl;

std:: cout << '\t' << "2 for calculating the GCF; " << std:: endl;

while (printf("Please enter your command: ") && (scanf("%d", &command)) != EOF) {

    if (command == 0) {

        dlclose(handle); //освобождает указатель на библиотеку и программа перестает ей
пользоваться

        if (curlib == 0) {

            curlib = 1 - curlib;

            handle = dlopen(lib_array[curlib], RTLD_LAZY);

            if (!handle) {

                std:: cout << "An error while opening library has been detected" << std:: endl;

                exit(EXIT_FAILURE);

            }

            GCF = (int (*)(int, int))dlsym(handle, "GCF");

            PrimeCount = (int (*)(int, int))dlsym(handle, "PrimeCount");

        }

    }

    else if (curlib == 1) {

        curlib = 1 - curlib;

        handle = dlopen(lib_array[curlib], RTLD_LAZY);

        if (!handle) {

            std:: cout << "An error while opening library has been detected" << std:: endl;

```



```

        exit(EXIT_FAILURE);
    }

    GCF = (int (*)(int, int))dlsym(handle, "GCF");

    PrimeCount = (int (*)(int, int))dlsym(handle, "PrimeCount");
}

std::cout << "You have changed contracts!" << std::endl;
}

else if (command == 1) {
    int A, B;

    std::cout << "enter A and B\n";

    std::cin >> A >> B;

    int prime_count = PrimeCount (A, B);

    printf("PrimeCount = %d\n", prime_count);
}

else if (command == 2) {
    int A, B;

    std::cout << "enter A and B\n";

    std::cin >> A >> B;

    int gcf = GCF (A, B);

    printf("GCF = %d\n", gcf);
}

else {
    std::cout << "You had to enter only 0, 1 or 2!" << std::endl;
}
}

dlclose(handle);

return 0;
}

```

realization1.cpp:

```
extern "C" int PrimeCount(int A, int B);
```

```
extern "C" int GCF(int A, int B);
```

```
int PrimeCount(int A, int B) {
```

```
    int count = 0;
```

```
    for (int i = A; i <= B; i++){
```

```
        int j;
```

```
        for(j = 2; j <= i; j++){
```

```
            if(i % j == 0) break;
```

```
            if(j == i) count++;
```

```
        }
```

```
    return count;
```

```
}
```

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

```
    while (A!=0 && B!=0)
```

```
        if (A > B)
```

```
            A = A % B;
```

```
        else
```

```
            B = B % A;
```

```
    return A+B;
```

```
}
```

realization2.cpp:

```
extern "C" int PrimeCount(int A, int B);
```

```
extern "C" int GCF(int A, int B);
```

```
int PrimeCount(int A, int B) {
```

```
    int count = 0;
```

```
    int n = B + 1;
```

```

int *arr = new int[n];

for (int i = 0; i < n; i++)

    arr[i] = i;

for (int i = 2; i < n; i++)

    if (arr[i] != 0 ) {

        if (arr[i] >= A && arr[i] <= B) count++;

        for (int j = i*i; j < n; j += i)

            arr[j] = 0;

    }

return count;
}

```

```

int GCF(int A, int B) {

    A = A < B ? A : B;

    B = A > B ? A : B;

    int c = A;

    while (c) {

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

            break;

        c--;

    }

    return c;

}

```

STRACE

```

execve("./main2", ["/main2"], 0x7ffcdabc60 /* 20 vars */) = 0
brk(NULL)                = 0x7fff4e30000
arch_prctl(0x3001 /* ARCH_???, 0x7fffd0bc360) = -1 EINVAL (Invalid argument)
access("/etc/ld.so.preload", R_OK)    = -1 ENOENT (No such file or directory)

```

```

openat(AT_FDCWD, "/tls/haswell/x86_64/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)

openat(AT_FDCWD, "/tls/haswell/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file
or directory)

openat(AT_FDCWD, "/tls/x86_64/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file
or directory)

openat(AT_FDCWD, "/tls/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/haswell/x86_64/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such
file or directory)

openat(AT_FDCWD, "/haswell/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/x86_64/libdl.so.2", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "/libdl.so.2", 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=41063, ...}) = 0

mmap(NULL, 41063, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f31dc165000

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\3\0>\0\1\0\0\0 \22\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) =
0x7f31dc1a0000

mmap(NULL, 20752, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f31dc150000

mmap(0x7f31dc151000, 8192, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f31dc151000

mmap(0x7f31dc153000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x3000) = 0x7f31dc153000

mmap(0x7f31dc154000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7f31dc154000

close(3) = 0

openat(AT_FDCWD, "/tls/haswell/x86_64/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT
(No such file or directory)

openat(AT_FDCWD, "/tls/haswell/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such
file or directory)

```

openat(AT_FDCWD, "/tls/x86_64/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/tls/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/haswell/x86_64/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/haswell/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/x86_64/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.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\0\341\t\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=1956992, ...}) = 0

mmap(NULL, 1972224, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f31dbf60000

mprotect(0x7f31dbff6000, 1290240, PROT_NONE) = 0

mmap(0x7f31dbff6000, 987136, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x96000) = 0x7f31dbff6000

mmap(0x7f31dc0e7000, 299008, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x187000) = 0x7f31dc0e7000

mmap(0x7f31dc131000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1d0000) = 0x7f31dc131000

mmap(0x7f31dc13f000, 10240, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f31dc13f000

close(3) = 0

openat(AT_FDCWD, "/tls/haswell/x86_64/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/tls/haswell/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/tls/x86_64/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, "/haswell/x86_64/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

```

openat(AT_FDCWD, "./haswell/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "./x86_64/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, "/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\0\360q\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"..., 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", 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", 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) = 0x7f31dbd60000

mprotect(0x7f31dbd85000, 1847296, PROT_NONE) = 0

mmap(0x7f31dbd85000, 1540096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x25000) = 0x7f31dbd85000

mmap(0x7f31dbefd000, 303104, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x19d000) = 0x7f31dbefd000

mmap(0x7f31dbf48000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1e7000) = 0x7f31dbf48000

mmap(0x7f31dbf4e000, 13528, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f31dbf4e000

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\3\0>\0\1\0\0\0\300\363\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=1369352, ...}) = 0

mmap(NULL, 1368336, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f31dbc11000

mmap(0x7f31dbc20000, 684032, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xf000) = 0x7f31dbc20000

mmap(0x7f31dbcc7000, 618496, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0xb6000) = 0x7f31dbcc7000

```

```

mmap(0x7f31dbd5e000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x14c000) = 0x7f31dbd5e000

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\3\0>\0\1\0\0\0\3405\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=104984, ...}) = 0

mmap(NULL, 107592, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f31dbbf0000

mmap(0x7f31dbbf3000, 73728, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x3000) = 0x7f31dbbf3000

mmap(0x7f31dbc05000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x15000) = 0x7f31dbc05000

mmap(0x7f31dbc09000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x18000) = 0x7f31dbc09000

close(3)                = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f31dc160000

mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x7f31dbbe0000

arch_prctl(ARCH_SET_FS, 0x7f31dbbe0740) = 0

mprotect(0x7f31dbf48000, 12288, PROT_READ) = 0

mprotect(0x7f31dbc09000, 4096, PROT_READ) = 0

mprotect(0x7f31dbd5e000, 4096, PROT_READ) = 0

mprotect(0x7f31dc131000, 45056, PROT_READ) = 0

mprotect(0x7f31dc154000, 4096, PROT_READ) = 0

mprotect(0x7f31dc1a8000, 4096, PROT_READ) = 0

mprotect(0x7f31dc19d000, 4096, PROT_READ) = 0

munmap(0x7f31dc165000, 41063)      = 0

brk(NULL)                = 0x7fff4e30000

brk(0x7fff4e51000)       = 0x7fff4e51000

fstat(1, {st_mode=S_IFCHR|0660, st_rdev=makedev(0x4, 0x1), ...}) = 0

ioctl(1, TCGETS, {B38400 opost isig icanon echo ...}) = 0

write(1, "Enter start library: \n", 22Enter start library:

) = 22

```

```

write(1, "\t1 for using first library\n", 27 1 for using first library
) = 27
write(1, "\t2 for using second library\n", 28 2 for using second library
) = 28
fstat(0, {st_mode=S_IFCHR|0660, st_rdev=makedev(0x4, 0x1), ...}) = 0
ioctl(0, TCGETS, {B38400 opost isig icanon echo ...}) = 0
read(0, 1
"1\n", 4096) = 2
openat(AT_FDCWD, "./tls/haswell/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)
openat(AT_FDCWD, "./tls/haswell/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "./tls/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "./tls/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "./haswell/x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such
file or directory)
openat(AT_FDCWD, "./haswell/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "./x86_64/libd1.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "./libd1.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@\20\0\0\0\0\0"... , 832) = 832
fstat(3, {st_mode=S_IFREG|0777, st_size=15680, ...}) = 0
getcwd("/mnt/c/Users/\320\241\320\260\320\262\320\265\320\273\320\270\320\271/Documents/GitHub/
OS/lab5", 128) = 53
mmap(NULL, 16424, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f31dbbd0000
mmap(0x7f31dbbd1000, 4096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f31dbbd1000
mmap(0x7f31dbbd2000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x2000) = 0x7f31dbbd2000
mmap(0x7f31dbbd3000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f31dbbd3000
close(3) = 0

```



```

mprotect(0x7f31dbbd3000, 4096, PROT_READ) = 0

write(1, "Hello there! Please enter your c"..., 65Hello there! Please enter your command according to next
rules:

) = 65

write(1, "\t0 for changing the contract;\n", 30 0 for changing the contract;

) = 30

write(1, "\t1 for calculating the PrimeCoun"..., 36 1 for calculating the PrimeCount;

) = 36

write(1, "\t2 for calculating the GCF; \n", 29 2 for calculating the GCF;

) = 29

write(1, "Please enter your command: ", 27Please enter your command: ) = 27

read(0, 0

"0\n", 4096) = 2

munmap(0x7f31dbbd0000, 16424) = 0

openat(AT_FDCWD, "./tls/haswell/x86_64/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)

openat(AT_FDCWD, "./tls/haswell/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "./tls/x86_64/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "./tls/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "./haswell/x86_64/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such
file or directory)

openat(AT_FDCWD, "./haswell/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "./x86_64/libd2.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or
directory)

openat(AT_FDCWD, "./libd2.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\200\20\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0777, st_size=16208, ...}) = 0

getcwd("/mnt/c/Users/\320\241\320\260\320\262\320\265\320\273\320\270\320\271/Documents/GitHub/
OS/lab5", 128) = 53

mmap(NULL, 16440, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x7f31dbbd0000

```

```

mmap(0x7f31dbbd1000, 4096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f31dbbd1000

mmap(0x7f31dbbd2000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x2000) = 0x7f31dbbd2000

mmap(0x7f31dbbd3000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f31dbbd3000

close(3)                = 0

mprotect(0x7f31dbbd3000, 4096, PROT_READ) = 0

write(1, "You have changed contracts!\n", 28You have changed contracts!
) = 28

write(1, "Please enter your command: ", 27Please enter your command: ) = 27

read(0, "", 4096)        = 0

munmap(0x7f31dbbd0000, 16440) = 0

exit_group(0)            = ?

+++ exited with 0 +++

```

Вывод

В процессе выполнения данной лабораторной работы я описал динамические библиотеки функций подсчета количества простых чисел на отрезке и поиска НОД в двух реализациях, написал две программы, использующие эти библиотеки. Я познакомился с динамическими и статическими библиотеками, научился использовать их в своих программах, ведь это ускоряет компиляцию программ, а динамические библиотеки экономят память: не нужно копировать файлы библиотеки, она подсоединяется в моменте работы программы.