

Московский авиационный институт  
(Национальный исследовательский университет)  
Факультет "Информационные технологии и прикладная математика"  
Кафедра "Вычислительная математика и программирование"

**Лабораторная работа №4 по курсу  
“Операционные системы”**

*Студент:* Иванов Андрей Кириллович

*Группа:* М8О-208Б-22

*Преподаватель:* Миронов Евгений Сергеевич

*Вариант:* 16

*Оценка:* \_\_\_\_\_

*Дата:* \_\_\_\_\_

*Подпись:* \_\_\_\_\_

Москва, 2023

# Содержание

# 1 Репозиторий

<https://github.com/kumaroid/osLABs>

## 2 Цель работы

Приобретение практических навыков в:

- Создании динамических библиотек
- Создании программ, которые используют функции динамических библиотек

## 3 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 использует одну из библиотек, используя знания полученные на этапе компиляции;
- Тестовая программа (программа №2), которая загружает библиотеки, используя только их местоположение и контракты.

## 4 Описание работы программы

Функции, написанные в результате выполнения лабораторной работы:

- Подсчёт количества простых чисел на отрезке  $[A, B]$  ( $A, B$  - натуральные)
- Подсчёт наибольшего общего делителя для двух натуральных чисел

В ходе выполнения лабораторной работы я использовал следующие системные вызовы:

- `dlopen` - открытие динамического объекта
- `dlclose` - закрытие динамического объекта

## 5 Исходный код

lib.hpp

```
1 #pragma once
2
3 #ifdef __cplusplus
4 extern "C" {
5 #endif
6
7 int PrimeCount(int a, int b);
8 int GCF(int a, int b);
9
10 #ifdef __cplusplus
11 }
12 #endif
```

lib1.cpp

```
1 #include "lib.hpp"
2 #include <iostream>
3 extern "C" bool IsPrime(int a) {
4     bool res = 1;
5     for (int i = 2; i < a; ++i) {
6         if (a % i == 0) {
7             res = 0;
8         }
9     }
10    return res;
11 }
12
13 extern "C" int PrimeCount(int a, int b) {
14     int count = 0;
15     for (int i = a; i <= b; ++i) {
16         if (IsPrime(i)) {
17             ++count;
18         }
19     }
20    return count;
21 }
22 extern "C" int GCF(int a, int b) {
23     while (a != 0 && b != 0) {
24         if (a > b) {
25             a = a % b;
26         } else {
27             b = b % a;
28         }
29     }
30    return a + b;
31 }
```

lib2.cpp

```
1 #include "lib.hpp"
2 #include <iostream>
3 #include <vector>
4
5 extern "C" int PrimeCount(int a, int b) {
6     std::vector<bool> prime(b + 1, true);
7     prime[0] = prime[1] = false;
8     for (int i = 2; i * i <= b + 1; ++i) {
```

```

9         for (int j = i * i; j <= b + 1; j += i) {
10             prime[j] = false;
11         }
12     }
13     int count = 0;
14     for (int i = a; i <= b; ++i) {
15         if (prime[i]) {
16             ++count;
17         }
18     }
19     return count;
20 }
21 extern "C" int GCF(int a, int b) {
22     int res = 0;
23     for (int i = std::min(a, b); i > 0; --i) {
24         if (a % i == 0 && b % i == 0) {
25             res = i;
26         }
27     }
28     return res;
29 }

```

main\_dynamic.cpp

```

1 #include <iostream>
2 #include <cstdlib>
3 #include <dlfcn.h>
4
5 using PrimeCountFunc = int (*)(int, int);
6 using GCFFunc = int (*)(int, int);
7
8 void* loadLibrary(const std::string& libraryName) {
9     void* handle = dlopen(libraryName.c_str(), RTLD_LAZY);
10    if (!handle) {
11        std::cerr << "Cannot load library: " << dlerror() << std::
endl;
12        exit(EXIT_FAILURE);
13    }
14    return handle;
15 }
16
17 void unloadLibrary(void* handle) {
18     if (dlclose(handle) != 0) {
19        std::cerr << "Cannot unload library: " << dlerror() << std
::endl;
20        exit(EXIT_FAILURE);
21    }
22 }
23
24 int main() {
25     int whichLib = 1;
26
27     const char* PathToLib1 = getenv("PATH_TO_LIB1");
28     const char* PathToLib2 = getenv("PATH_TO_LIB2");
29
30     if (PathToLib1 == nullptr || PathToLib2 == nullptr) {
31        std::cout << "PATH_TO_LIB1 or PATH_TO_LIB2 is not
specified\n";
32        exit(1);
33    }

```

```

34
35     void* libraryHandle = loadLibrary(PathToLib1);
36
37     PrimeCountFunc PrimeCount = (PrimeCountFunc)dlsym(
libraryHandle, "PrimeCount");
38     GCFFunc GCF = (GCFFunc)dlsym(libraryHandle, "GCF");
39     std::string command;
40
41     while (true) {
42         std::cin >> command;
43         int a, b;
44         if (command == "0") {
45             unloadLibrary(libraryHandle);
46             if (whichLib == 2) {
47                 libraryHandle = loadLibrary(PathToLib1);
48                 whichLib = 1;
49             } else if (whichLib == 1) {
50                 libraryHandle = loadLibrary(PathToLib2);
51                 whichLib = 2;
52             }
53             PrimeCount = reinterpret_cast<PrimeCountFunc>(dlsym(
libraryHandle, "PrimeCount"));
54             GCF = reinterpret_cast<GCFFunc>(dlsym(libraryHandle, "
GCF"));
55
56             } else if (command == "1") {
57                 std::cin >> a >> b;
58                 std::cout << "OK\n";
59                 std::cout << "PrimeCount = " << PrimeCount(a, b) <<
std::endl;
60             } else if (command == "2") {
61                 std::cin >> a >> b;
62                 std::cout << "OK\n";
63                 std::cout << "GCF = " << GCF(a, b) << std::endl;
64             } else {
65                 std::cout << "Wrong Argument" << std::endl;
66             }
67         }
68     }
69 }

```

#### main\_static.cpp

```

1  #include <iostream>
2  #include "lib.hpp"
3
4  int main() {
5      std::string command;
6      int a, b;
7
8      while (true) {
9          std::cin >> command;
10         if (command == "0") {
11             break;
12         } else if (command == "1") {
13             std::cin >> a >> b;
14             std::cout << "PrimeCount = " << PrimeCount(a, b) <<
std::endl;
15         } else if (command == "2") {
16             std::cin >> a >> b;

```

```
17         std::cout << "GCF = " << GCF(a, b) << std::endl;
18     } else {
19         std::cout << "Wrong Argument" << std::endl;
20     }
21 }
22 }
```

## 6 Тесты



## 7 Запуск тестов

## 8 Выводы

В результате выполнения данной лабораторной работы были созданы динамические библиотеки, которые реализуют функционал в соответствие с вариантом задания на C++. Были получены практические навыки в создании программ, которые используют функции динамических библиотек.