

**Московский авиационный институт  
(Национальный исследовательский университет)**

**Факультет информационных технологий и прикладной математики**

**Кафедра вычислительной математики и программирования**

**Лабораторная работа № 5 по курсу  
«Операционные системы»  
Вариант 14**

Студент: Аминов Степан  
Сергеевич  
Группа: 8О-308  
Преподаватель: Е. С. Миронов  
Вариант: 14  
Дата:  
Оценка:

Москва, 2021

# Лабораторная работа №5

## 1 Описание

Данная лабораторная работа будет выполняться в ОС Unix.

**Тема:** Динамические библиотеки

### Цель работы:

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 No1), которая использует одну из библиотек, используя знания полученные на этапе компиляции;
- Тестовая программа (программа No2), 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 No2). 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 № 14:** контракты: 2 – расчёт производной функции  $\cos(x)$  в точке А с приращением  $\delta x$ ; 8 – перевод числа  $x$  из десятичной системы счисления в другую.

## 2 Метод решения

Используемые библиотечные вызовы:

|  |   |
|--|---|
| <code>void *dlopen(const char *filename, int flag);</code> | Загружает динамическую библиотеку, имя которой указано в строке filename и возвращает прямой указатель на начало загруженной библиотеки.  |
| <code>const char *dlerror(void);</code>                    | Возвращает указатель на начало строки, описывающей ошибку, полученную на предыдущем вызове.   |
| <code>void *dlsym(void *handle, char *symbol);</code>      | Получает параметр handle, который является выходом вызова dlopen и параметр symbol, который является строкой, в которой содержится название символа, который необходимо загрузить из библиотеки. Возвращает указатель на область памяти, в которой содержится необходимый символ. |
| <code>int dlclose(void *handle);</code>                    | Уменьшает счетчик ссылок на указатель handle и если он равен нулю, то освобождает библиотеку.   |

Лабораторная работа построена на системе сборки stake.

Для обычной компиляции программ следует написать в директории с файлом *CMakeLists.txt* «stake .». Для выбора того, какую реализацию стоит использовать при компиляции со статической библиотекой, следует написать «stake . -Dlab\_link\_type=first», где first показывает, что нужно использовать первую реализацию; «second» означало бы, что нужно использовать вторую реализацию.

В *CMakeLists.txt* происходит создание двух динамических библиотек с одним и тем же общим заголовком и с библиотекой m, что означает math. После идет обработка данных, переданных пользователем и затем компиляция выбранной пользователем библиотеки (для статической версии). Для динамической версии происходит установление зависимости главной программы от двух динамических библиотек.

При обычной компиляции динамических библиотек для запуска динамической версии программы необходимо для начала экспортировать переменную окружения **LD\_LIBRARY\_PATH**, но при использовании stake данная необходимость отпадает.

## 3 Консоль

```
magic@magical:~/CLionProjects/os5$ LD_LIBRARY_PATH=. ./main1
1
Enter x: 5
Enter dx: 7
0.080027
magic@magical:~/CLionProjects/os5$ LD_LIBRARY_PATH=. ./main1
2
Enter a: 151
Translate integer(151) from 10 base to 2 base10010111
```

```
magic@magical:~/CLionProjects/os5$ LD_LIBRARY_PATH=. ./main2
```

```
1
```

```
Enter x: 5
```

```
Enter dx: 7
```

```
0.080027
```

```
2
```

```
151
```

```
Translate integer(151) from 10 base to 2 base
```

```
10010111
```

## 4 strace

```
magic@magical:~/CLionProjects/os5$ strace ./main1
execve("./main1", ["/main1"], 0x7ffc921b7660 /* 66 vars */) = 0
brk(NULL)                               = 0x5575074f9000
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffcbbcf3100) = -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=102741, ...}) = 0
mmap(NULL, 102741, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7fcd6fa87000
close(3)                                 = 0
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/haswell/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/tls/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/tls", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/haswell/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/lib/x86_64-linux-gnu", {st_mode=S_IFDIR|0755, st_size=12288, ...}) = 0
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/tls/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such
```

```

file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/haswell/libfirst.so", O_RDONLY|
O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/tls/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/x86_64/libfirst.so", O_RDONLY|
O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/tls/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/tls/libfirst.so", O_RDONLY|O_CLOEXEC) = -
1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/tls", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/haswell/x86_64/libfirst.so", O_RDONLY|
O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file
or directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/haswell/libfirst.so", O_RDONLY|
O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/x86_64/libfirst.so", O_RDONLY|
O_CLOEXEC) = -1 ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or
directory)
openat(AT_FDCWD, "/usr/lib/x86_64-linux-gnu/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/usr/lib/x86_64-linux-gnu", {st_mode=S_IFDIR|0755, st_size=86016, ...}) = 0
openat(AT_FDCWD, "/lib/tls/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/lib/tls/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/tls/haswell/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT
(No such file or directory)
stat("/lib/tls/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/tls/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT
(No such file or directory)
stat("/lib/tls/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/tls/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such
file or directory)
stat("/lib/tls", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/lib/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/haswell/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)
stat("/lib/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)

```

```

openat(AT_FDCWD, "/lib/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)
stat("/lib/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/lib/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file
or directory)
stat("/lib", {st_mode=S_IFDIR|0755, st_size=4096, ...}) = 0
openat(AT_FDCWD, "/usr/lib/tls/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/usr/lib/tls/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/tls/haswell/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/usr/lib/tls/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/tls/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/usr/lib/tls/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/tls/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No
such file or directory)
stat("/usr/lib/tls", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/haswell/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1
ENOENT (No such file or directory)
stat("/usr/lib/haswell/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/haswell/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT
(No such file or directory)
stat("/usr/lib/haswell", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/x86_64/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT
(No such file or directory)
stat("/usr/lib/x86_64", 0x7ffcbbcf2350) = -1 ENOENT (No such file or directory)
openat(AT_FDCWD, "/usr/lib/libfirst.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such
file or directory)
stat("/usr/lib", {st_mode=S_IFDIR|0755, st_size=4096, ...}) = 0
writev(2, [{iov_base="/main1", iov_len=7}, {iov_base=": ", iov_len=2}, {iov_base="error while
loading shared libra...", iov_len=36}, {iov_base=": ", iov_len=2}, {iov_base="libfirst.so",
iov_len=11}, {iov_base=": ", iov_len=2}, {iov_base="cannot open shared object file",
iov_len=30}, {iov_base=": ", iov_len=2}, {iov_base="No such file or directory", iov_len=25},
{iov_base="\n", iov_len=1}], 10./main1: error while loading shared libraries: libfirst.so: cannot
open shared object file: No such file or directory
) = 118
exit_group(127) = ?
+++ exited with 127 +++

```

## 5 Исходный код

### realization2.c:

```
#include "realization.h"

#include <stdlib.h>
#include <string.h>
#include <math.h>

float SinIntegral(float A, float B, float e){
    int n = (int)((B - A) / e);

    float h = (B-A)/(float)n;

    float s = sinf(A)+sinf(B);

    for (int i = 1; i < n; i++){
        s += 2*sinf(A+(float)i*h);
    }

    return h*s/2;
}

#pragma clang diagnostic push
#pragma ide diagnostic ignored "DanglingPointers"
char* translation(long x){
    char *res = calloc(1, sizeof(char));
    res[0] = '\0';
    if(x == 0){
        char *tmp = calloc(strlen(res) + 2, sizeof(char));
        strcpy(tmp+1, res);
        free(res);
        res = tmp;
        res[0] = '0';
    }
    while(x > 0){
        char ch = (x % 3) + '0';
        x /= 3;
        char *tmp = calloc(strlen(res) + 2, sizeof(char));
        strcpy(tmp+1, res);
        free(res);
        res = tmp;
        res[0] = ch;
    }
    return res;
}
```

### realization1.c:

```
#include <math.h>
#include <stdlib.h>
#include <string.h>

#include "realization.h"
```



```

float SinIntegral(float A, float B, float e){
    float step = e;
    int count = (int)((B - A) / step);

    float ans = 0;
    for (int i = 0; i <= count; i++) {
        ans += step * sinf(A + (float)(i - 1) * step);
    }
    return ans;
}

#pragma clang diagnostic push
#pragma ide diagnostic ignored "DanglingPointers"
char* translation(long x){
    char *res = calloc(1, sizeof(char));
    res[0] = '\0';
    if(x == 0){
        char *tmp = calloc(strlen(res) + 2, sizeof(char));
        strcpy(tmp+1, res);
        free(res);
        res = tmp;
        res[0] = '0';
    }
    while(x > 0){
        char ch = (x & 1u) + '0';
        x >>= 1u;
        char *tmp = calloc(strlen(res) + 2, sizeof(char));
        strcpy(tmp+1, res);
        free(res);
        res = tmp;
        res[0] = ch;
    }
    return res;
}
#pragma clang diagnostic pop

```

### **main1.c:**

```

#include <stdio.h>
#include <stdlib.h>
#include <dlfcn.h>

```

```

typedef enum {
    FIRST,
    SECOND,
} CONTEXT;

```

```

CONTEXT r = FIRST;

```

```

const char* libName1 = "libfirst.so";
const char* libName2 = "libsecond.so";

```

```

float (*sinInt)(float, float, float) = NULL;

```

```

char* (*translation)(long x) = NULL;
char *err;

void *libHandle = NULL;

void loadDLibs(CONTEXT context){
    const char *name;
    if(context == FIRST){
        name = libName1;
    } else{
        name = libName2;
    }

    libHandle = dlopen(name, RTLD_LAZY);
    if(!libHandle){
        fprintf(stderr, "%s\n", dlerror());
        exit(EXIT_FAILURE);
    }
}

void unloadDLibs(){
    dlclose(libHandle);
}

void loadContext(){
    loadDLibs(r);
    sinInt = dlsym(libHandle, "SinIntegral");
    translation = dlsym(libHandle, "translation");
    if((err = dlerror())) {
        fprintf(stderr, "%s\n", err);
        exit(EXIT_FAILURE);
    }
}

void changeContext(){
    unloadDLibs();
    if(r == FIRST){
        r = SECOND;
    } else {
        r = FIRST;
    }

    loadContext();
}

int main(){
    r = FIRST;
    loadContext();

    int cmd = 0;

    while (scanf("%d", &cmd) != EOF){

        if(cmd == 0){
            changeContext();

```

```

    puts("Ok. Contract was changed");
    if(r == FIRST){
        puts("Now context is first");
    } else{
        puts("Now context is second");
    }
    continue;
}
if(cmd == 1) {
    float a, b, c;
    if (scanf("%f %f %f", &a, &b, &c) == EOF) {
        break;
    }
    printf("%f\n", sinInt(a, b, c));
}
else{
    long x;
    if(scanf("%ld", &x)== EOF){
        break;
    }

    char *string;
    printf("Translate integer(%ld) from 10 base to ", x);
    if(r == FIRST) {
        printf("2");
    }else{
        printf("3");
    }
    puts(" base");
    string = translation(x);
    printf("%s\n", string);
    free(string);
}
}
}
unloadDLibs();
}

```

### realization.h:

```

#ifndef INC_5_LAB_REALIZATION1_H
#define INC_5_LAB_REALIZATION_H

extern float SinIntegral(float A, float B, float e);
extern char* translation(long x);

#endif //INC_5_LAB_REALIZATION1_H

```

### main2.cpp:

```

#include "realization.h"

#include <stdio.h>

int main() {
    int cmd = 0;

```

```

while (scanf("%d", &cmd) != EOF) {

    float a, b, c;
    if (scanf("%f %f %f", &a, &b, &c) == EOF) {
        break;
    }

    if (cmd == 1) {
        puts("Calculate sin(x)");
        printf("%f\n", SinIntegral(a, b, c));
    }
}

```

### **CMakeLists.txt:**

```

cmake_minimum_required(VERSION 3.17)
project(os5 LANGUAGES C VERSION 0.1.0)

set(CMAKE_C_STANDARD 99)

add_executable(main2 src/main1.c)
add_executable(main1 src/main2.c)

add_library(first SHARED
    src/realization1.c
)

set_target_properties(first PROPERTIES
    VERSION ${PROJECT_VERSION}
    PUBLIC_HEADER src/realization.h
)

add_library(
    second SHARED
    src/realization2.c
)

set_target_properties(second PROPERTIES
    VERSION ${PROJECT_VERSION}
    PUBLIC_HEADER src/realization.h
)

target_link_libraries(first m)
target_link_libraries(second m)

target_include_directories(first PRIVATE src/)
target_include_directories(second PRIVATE src/)

set(custom_targets)

```

```

list(APPEND custom_targets first)
list(APPEND custom_targets second)

if(NOT lab_link_type)
    set(lab_link_type first)
endif()

function(invalid_target)
    message("Unknown target selected")
    message("All available targets:")
    foreach(target IN ITEMS ${custom_targets})
        message("  ${target}")
    endforeach()
endfunction()

if (${lab_link_type} IN_LIST custom_targets)
else()
    invalid_target()
    message( FATAL_ERROR "Invalid target selected" )
endif()

message("Link library for statMain will be now " ${lab_link_type})

target_link_libraries(statMain ${lab_link_type})
target_link_libraries(dynMain ${CMAKE_DL_LIBS})
add_dependencies(dynMain first second)

```

## 6 Вывод

При выполнении данной лабораторной работы я научился создавать свои библиотеки, а также использовать их. Также я попрактиковался в применении математических формул в программировании. В ходе данной лабораторной работы я узнал что такое динамические библиотеки и чем они отличаются от статических. К плюсам использования динамических библиотек я могу отнести то, что они загружаются и линкуются только по необходимости и при их ненужности не будут замедлять программу.