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: М80-206Б-20

Студент: Носов А.К.

Преподаватель: Миронов Е.С.

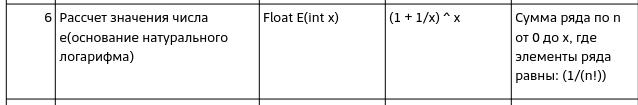
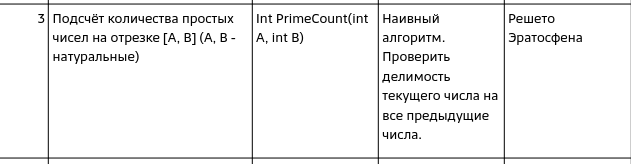
Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: 01.03.2024

Москва, 2024

**Постановка задачи**

Необходимо реализовать две динамические библиотеки:



**Общий метод и алгоритм решения**

Использованные системные вызовы:

* void \*dlopen(const char \*filename, int flag) -загружает динамическую библиотеку, имя которой указано в строке filename, и возвращает прямой указатель на начало динамической библиотеки.
* void \*dlsym(void \*handle, char \*symbol) - использует указатель на динамическую библиотеку, возвращаемую dlopen, и оканчивающееся нулем символьное имя, а затем возвращает адрес, указывающий, откуда загружается этот символ.
* int dlclose(void \*handle) - уменьшает на единицу счетчик ссылок на указатель динамической библиотеки handle. Если нет других загруженных библиотек, использующих ее символы и если счетчик ссылок принимает нулевое значение, то динамическая библиотека выгружается.

В начале создадим две библиотеки, каждая из которых содержит две функции: расчет производной функции и подсчет площади плоской фигуры. После этого создадим две программы. В первой будем использовать библиотеку, которую получим на этапе компиляции. Во второй программе с помощью системных вызовов откроем библиотеку и вызовем функции поочередно. Также сделаем возможность замены библиотеки с одной на другую.

**Код программы**

**main1.c**

#include <stdio.h>

#include <stdlib.h>

#include "lib1.c"

#include "lib2.c"

int main(){

int check;

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

if (check != 1 && check != 2){

printf("You can only enter 1, 2 or 0\n");

return 1;

}

else if(check == 1){

printf("Your choice: prime count\n Please, enter 2 numbers: ");

float arg1, arg2;

scanf("%f %f", &arg1, &arg2);

if (arg2<arg1) {

printf("Second number should be bigger!\n");

return 1;

}

printf("Naive implement: %d\n", naive\_prime\_count(arg1, arg2));

printf("Eratosthenes method: %d\n", eratosphene\_prime\_count(arg1, arg2));

}

else if(check == 2){

printf("Your choice: calculate Euler's number\n Please, enter 1 number: ");

float arg;

scanf("%f", &arg);

if (arg <= 0) {

printf("Number should be bigger than 0!\n");

return 1;

}

printf("(1 + 1 / x) ^ x method: %f\n", E(arg));

printf("Sum of series method: %f\n", sum\_of\_series(arg));

}

}

return 0;

}

**main2.c**

#include <stdio.h>

#include <stdlib.h>

#include <dlfcn.h>

int main(){

int check;

void \*current\_lib;

int arg, arg1, arg2;

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

if (check != 1 && check != 2){

printf("You can only enter 1, 2 or 0\n");

return 1;

}

else if(check == 1){

printf("Your choice: prime count\n Please, enter 2 numbers: ");

current\_lib = dlopen("./lib1.so", RTLD\_LAZY);

if (!current\_lib) {

printf("Library loading error\n");

return 1;

}

float (\*naive\_prime\_count)(float, float) = dlsym(current\_lib, "naive\_prime\_count");

if (!naive\_prime\_count) {

printf("Could not find function\n");

return 1;

}

float (\*eratosphene\_prime\_count)(float, float) = dlsym(current\_lib, "eratosphene\_prime\_count");

if (!eratosphene\_prime\_count) {

printf("Could not find function\n");

return 1;

}

scanf("%d %d", &arg1, &arg2);

if (arg2<arg1) {

printf("Second number should be bigger!\n");

return 1;

}

printf("Naive implement: %f\n", naive\_prime\_count(arg1, arg2));

printf("Eratosthenes method: %f\n", eratosphene\_prime\_count(arg1, arg2));

}

else if(check == 2){

printf("Your choice: calculate Euler's number\n Please, enter 1 number: ");

current\_lib = dlopen("./lib2.so", RTLD\_LAZY);

if (!current\_lib) {

printf("Library loading error\n");

return 1;

}

float (\*E)(float) = dlsym(current\_lib, "E");

if (!E) {

printf("Could not find function\n");

return 1;

}

float (\*sum\_of\_series)(float) = dlsym(current\_lib, "sum\_of\_series");

if (!sum\_of\_series) {

printf("Could not find function\n");

return 1;

}

scanf("%d", &arg);

if (arg <= 0) {

printf("Number should be bigger than 0!\n");

return 1;

}

printf("(1 + 1 / x) ^ x method: %f\n", E(arg));

printf("Sum of series method: %f\n", sum\_of\_series(arg));

}

dlclose(current\_lib);

return 0;

}

}

**lib1.c**

#ifndef MYLIBRARY2

#define MYLIBRARY2

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

int naive\_prime\_count(int, int);

int eratosphene\_prime\_count(int, int);

#endif

int naive\_prime\_count(int A, int B)

{

int counter = 0;

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

if (i == 0 || i == 1) {

continue;

}

int count\_divider = 0;

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

if (i % j == 0) {

++count\_divider;

}

}

if (count\_divider <= 1) {

++counter;

}

}

return counter;

}

int eratosphene\_prime\_count(int A, int B)

{

bool is\_prime[B + 1];

for (int i = 0; i <= B; ++i) {

is\_prime[i] = true;

}

is\_prime[0] = is\_prime[1] = false;

for (int i = 2; i \* i <= B; ++i) {

if (is\_prime[i]) {

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

is\_prime[j] = false;

}

}

}

int counter = 0;

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

if (is\_prime[i]) {

++counter;

}

}

return counter;

}

**lib2.c**

#ifndef MYLIBRARY1

#define MYLIBRARY1

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

float E(int);

float sum\_of\_series(int x);

#endif

double power(double base, int exponent) {

double result = 1.0;

for (int i = 0; i < exponent; i++) {

result \*= base;

}

return result;

}

float E(int x)

{

return power(1 + 1.0 / x, x);

}

float sum\_of\_series(int x)

{

float sum = 1.0 ;

float factorial = 1.0;

for (int n = 1; n <= x; ++n) {

factorial \*= n;

sum += 1.0 / factorial;

}

return sum;

}

**Протокол работы программы**

**Тестирование:**

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ g++ -c -o lib1.o lib1.c

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ gcc -shared -o lib1.so lib1.o

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ g++ -c -o lib2.o lib2.c

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ gcc -shared -o lib2.so lib2.o

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ gcc main1.c -o main1

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ gcc main2.c -o main2

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ ./main1

1

Your choice: prime count

Please, enter 2 numbers: 1

5

Naive implement: 3

Eratosthenes method: 3

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ ./main2

2

Your choice: calculate Euler's number

Please, enter 1 number: 4

(1 + 1 / x) ^ x method: 2.718282

Sum of series method: 2.718282

alex@alex-HP-ENVY-x360-Convertible-1

**Strace:**

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ strace

./main1

execve("./main1", ["./main1"], 0x7ffe17d74ac0 /\* 87 vars \*/) = 0

brk(NULL) = 0x5621f5960000

arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7ffedf45dd10) = -1 EINVAL (Invalid argument)

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_ANONYMOUS, -1, 0) = 0x7ffade009000

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

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=70887, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 70887, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7ffaddff7000

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\237\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 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) =

48

pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\302\211\332Pq\

2439\235\350\223\322\257\201\326\243\f"..., 68, 896) = 68

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=2220400, ...}, AT\_EMPTY\_PATH) = 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

mmap(NULL, 2264656, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7ffaddc00000

mprotect(0x7ffaddc28000, 2023424, PROT\_NONE) = 0

mmap(0x7ffaddc28000, 1658880, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|

MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7ffaddc28000

mmap(0x7ffadddbd000, 360448, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|

MAP\_DENYWRITE, 3, 0x1bd000) = 0x7ffadddbd000

mmap(0x7ffadde16000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_FIXED|MAP\_DENYWRITE, 3, 0x215000) = 0x7ffadde16000

mmap(0x7ffadde1c000, 52816, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7ffadde1c000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_ANONYMOUS, -1, 0) = 0x7ffaddff4000

arch\_prctl(ARCH\_SET\_FS, 0x7ffaddff4740) = 0

set\_tid\_address(0x7ffaddff4a10) = 6521

set\_robust\_list(0x7ffaddff4a20, 24) = 0

rseq(0x7ffaddff50e0, 0x20, 0, 0x53053053) = 0

mprotect(0x7ffadde16000, 16384, PROT\_READ) = 0

mprotect(0x5621f47c0000, 4096, PROT\_READ) = 0

mprotect(0x7ffade043000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024,

rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7ffaddff7000, 70887) = 0

newfstatat(0, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...},

AT\_EMPTY\_PATH) = 0

getrandom("\xc1\x6a\xbe\x55\x5d\xa4\xb4\xc8", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x5621f5960000

brk(0x5621f5981000) = 0x5621f5981000

read(0, 1

"1\n", 1024) = 2

newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...},

AT\_EMPTY\_PATH) = 0

write(1, "Your choice: prime count\n", 25Your choice: prime count

) = 25

write(1, " Please, enter 2 numbers: ", 26 Please, enter 2 numbers: ) = 26

read(0, 1

"1\n", 1024) = 2

read(0, 5

"5\n", 1024) = 2

write(1, "Naive implement: 3\n", 19Naive implement: 3

) = 19

write(1, "Eratosthenes method: 3\n", 23Eratosthenes method: 3

) = 23

read(0, "", 1024) = 0

exit\_group(0) = ?

+++ exited with 0 +++

alex@alex-HP-ENVY-x360-Convertible-13-ay1xxx:~/Desktop/OSLabs/lab4$ strace

./main2

execve("./main2", ["./main2"], 0x7ffdbc23fdc0 /\* 87 vars \*/) = 0

brk(NULL) = 0x561cadc45000

arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7ffc1e8f7f90) = -1 EINVAL (Invalid argument)

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_ANONYMOUS, -1, 0) = 0x7fa100041000

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

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=70887, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 70887, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7fa10002f000

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\237\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 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) =

48

pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\302\211\332Pq\

2439\235\350\223\322\257\201\326\243\f"..., 68, 896) = 68

newfstatat(3, "", {st\_mode=S\_IFREG|0755, st\_size=2220400, ...}, AT\_EMPTY\_PATH)

= 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

mmap(NULL, 2264656, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fa0ffe00000

mprotect(0x7fa0ffe28000, 2023424, PROT\_NONE) = 0

mmap(0x7fa0ffe28000, 1658880, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|

MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7fa0ffe28000

mmap(0x7fa0fffbd000, 360448, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|

MAP\_DENYWRITE, 3, 0x1bd000) = 0x7fa0fffbd000

mmap(0x7fa100016000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_FIXED|MAP\_DENYWRITE, 3, 0x215000) = 0x7fa100016000

mmap(0x7fa10001c000, 52816, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fa10001c000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_ANONYMOUS, -1, 0) = 0x7fa10002c000

arch\_prctl(ARCH\_SET\_FS, 0x7fa10002c740) = 0

set\_tid\_address(0x7fa10002ca10) = 7384

set\_robust\_list(0x7fa10002ca20, 24) = 0

rseq(0x7fa10002d0e0, 0x20, 0, 0x53053053) = 0

mprotect(0x7fa100016000, 16384, PROT\_READ) = 0

mprotect(0x561cadb4c000, 4096, PROT\_READ) = 0

mprotect(0x7fa10007b000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024,

rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7fa10002f000, 70887) = 0

newfstatat(0, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...},

AT\_EMPTY\_PATH) = 0

getrandom("\xdc\xe0\xc2\x51\x92\x11\xcb\x45", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x561cadc45000

brk(0x561cadc66000) = 0x561cadc66000

read(0, 2

"2\n", 1024) = 2

newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...},

AT\_EMPTY\_PATH) = 0

write(1, "Your choice: calculate Euler's n"..., 38Your choice: calculate Euler's number

) = 38

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\0\0\0\0\0\0\0\0"..., 832) = 832

newfstatat(3, "", {st\_mode=S\_IFREG|0664, st\_size=15424, ...}, AT\_EMPTY\_PATH) = 0

getcwd("/home/alex/Desktop/OSLabs/lab4", 128) = 31

mmap(NULL, 16432, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fa10003c000

mmap(0x7fa10003d000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|

MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7fa10003d000

mmap(0x7fa10003e000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|

MAP\_DENYWRITE, 3, 0x2000) = 0x7fa10003e000

mmap(0x7fa10003f000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|

MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7fa10003f000

close(3) = 0

mprotect(0x7fa10003f000, 4096, PROT\_READ) = 0

write(1, " Please, enter 1 number: ", 25 Please, enter 1 number: ) = 25

read(0, 2

"2\n", 1024) = 2

write(1, "(1 + 1 / x) ^ x method: 2.718282"..., 33(1 + 1 / x) ^ x method: 2.718282

) = 33

write(1, "Sum of series method: 2.718282\n", 31Sum of series method: 2.718282

) = 31

munmap(0x7fa10003c000, 16432) = 0

lseek(0, -1, SEEK\_CUR) = -1 ESPIPE (Illegal seek)

exit\_group(0) = ?

+++ exited with 0 +++

**Вывод**

В ходе выполнения данной лабораторной работы я научился использовать библиотеки двумя способами. Было интересно узнать некоторые особенности второго способа. Я уверен, что это пригодится мне в будущем.