

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: Кочев Д.О.

Преподаватель: Миронов Е.С.

Оценка: _____

Дата: 16.12.2023

Москва, 2023

Постановка задачи

Необходимо реализовать две динамические библиотеки:

2	Расчет производной функции $\cos(x)$ в точке A с приращением δx	Float Derivative(float A, float δx)	$f'(x) = (f(A + \delta x) - f(A))/\delta x$	$f'(x) = (f(A + \delta x) - f(A - \delta x))/(2 * \delta x)$
7	Подсчет площади плоской геометрической фигуры по двум сторонам	Float Square(float A, float B)	Фигура прямоугольник	Фигура прямоугольный треугольник

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

program1.c

```
#include <stdio.h>
```

```
#include "lib1/lib1.h"
```

```

int main(){
    int check;

    while(scanf("%d", &check) != EOF){
        if (check != 0 && check != 1 && check != 2){
            printf("You can only enter 1, 2 or 0\n");
            return 1;
        }
        else if(check == 1){
            float arg1, arg2;
            scanf("%f %f", &arg1, &arg2);
            float result = derivative(arg1, arg2);
            printf("Result: %f\n", result);
        }
        else if(check == 2){
            float arg1, arg2;
            scanf("%f %f", &arg1, &arg2);
            float result = square(arg1, arg2);
            printf("Result: %f\n", result);
        }
        else{
            printf("Sorry, adiós\n");
            return 0;
        }
    }
    return 0;
}

```

program2.c

```
#include <stdio.h>
```

```
#include <dlfcn.h>
```

```
int main(){
```

```
    int check;
```

```
    int key = 2;
```

```
    float result;
```

```
    float arg1, arg2;
```

```
    void *current_lib;
```

```
    current_lib = dlopen("./lib2/lib2.so", RTLD_LAZY); // Загрузка библиотеки
```

```
    if (!current_lib) {
```

```
        printf("Library loading error\n");
```

```
        return 1;
```

```
    }
```

```
    float (*square)(float, float) = dlsym(current_lib, "square"); // Получение  
указателя на функцию
```

```
    float (*derivative)(float, float) = dlsym(current_lib, "derivative");
```

```
    if (!square) {
```

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

```
        return 1;
```

```
    }
```

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

```
if (check != 0 && check != 1 && check != 2){  
    printf("You can only enter 1, 2 or 0\n");  
    dlclose(current_lib); // Закрытие библиотеки  
    return 1;  
}  
else if(check == 1){  
    scanf("%f %f", &arg1, &arg2);  
    result = derivative(arg1, arg2);  
    printf("Result: %f\n", result);  
}  
else if(check == 2){  
    scanf("%f %f", &arg1, &arg2);  
    result = square(arg1, arg2);  
    printf("Result: %f\n", result);  
}  
else if(check == 0){  
    if (key == 2){  
        current_lib = dlopen("./lib1/lib1.so", RTLD_LAZY);  
        key = 1;  
    }  
    else{  
        current_lib = dlopen("./lib2/lib2.so", RTLD_LAZY);  
        key = 2;  
    }  
  
    if (!current_lib) {  
        printf("Library loading error\n");  
    }  
}
```

```

        return 1;
    }

    square = dlsym(current_lib, "square");
    derivative = dlsym(current_lib, "derivative");
    // if (!square) {
    //     printf("Could not find function\n");
    //     return 1;
    // }
}

}

dlclose(current_lib);

return 0;

}

```

lib2.c

```

#include "lib2.h"

float derivative(float a, float deltax){
    float result;

    result = sinf(a + deltax) - sinf(a - deltax);

    result /= (2 * deltax);

    return result;
}

float square(float a, float b){
    float result = a * b;

    result /= 2;
}

```

```
    return result;
}
```

lib2.h

```
#ifndef MYLIBRARY2
#define MYLIBRARY2

#include <stdio.h>
#include <math.h>

float derivative(float a, float deltax);
float square(float a, float b);

#endif
```

lib1.c

```
#include "lib1.h"

float derivative(float a, float deltax){
    float result;

    result = sinf(a + deltax) - sinf(a);
    result /= deltax;
    return result;
}

float square(float a, float b){
    float result = a * b;
    return result;
}
```

lib1.h

```
#ifndef MYLIBRARY1
```

```
#define MYLIBRARY1
```

```
#include <stdio.h>
```

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

```
float derivative(float a, float deltax);
```

```
float square(float a, float b);
```

```
#endif
```

Протокол работы программы

Тестирование:

```
$ ./prog1
```

```
1 2 2.5
```

```
Result: -0.754731
```

```
2 2 2.5
```

```
Result: 5.000000
```

```
$ ./prog2
```

```
1 2 2.5
```

```
Result: -0.099621
```

```
2 2 2.5
```

```
Result: 2.500000
```

```
0
```

```
1 2 2.5
```

```
Result: -0.754731
```


2 2 2.5

Result: 5.000000

Strace:

```
$ strace -f ./prog2
execve("./prog2", ["/prog2"], 0x7fff103ee6b8 /* 36 vars */) = 0
brk(NULL)                                = 0x565108878000
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffe4f254490) = -1 EINVAL (Invalid
argument)
mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f829e9fd000
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=19839, ...},
AT_EMPTY_PATH) = 0
mmap(NULL, 19839, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f829e9f8000
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"..., 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"..., 48,
848) = 48
pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0
=\340\256\3\265?\356\25x\261\27\313A#350"..., 68, 896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2216304, ...},
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, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3,
0) = 0x7f829e7d0000
mmap(0x7f829e7f8000, 1658880, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) =
0x7f829e7f8000
mmap(0x7f829e98d000, 360448, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) =
0x7f829e98d000
mmap(0x7f829e9e5000, 24576, PROT_READ|PROT_WRITE,
```

```

MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) =
0x7f829e9e5000
mmap(0x7f829e9eb000, 52816, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f829e9eb000
close(3) = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f829e7cd000
arch_prctl(ARCH_SET_FS, 0x7f829e7cd740) = 0
set_tid_address(0x7f829e7cda10) = 108265
set_robust_list(0x7f829e7cda20, 24) = 0
rseq(0x7f829e7ce0e0, 0x20, 0, 0x53053053) = 0
mprotect(0x7f829e9e5000, 16384, PROT_READ) = 0
mprotect(0x565107daa000, 4096, PROT_READ) = 0
mprotect(0x7f829ea37000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7f829e9f8000, 19839) = 0
getrandom("\x40\x08\xe2\x92\x3d\x67\x14\x0e", 8, GRND_NONBLOCK) = 8
brk(NULL) = 0x565108878000
brk(0x565108899000) = 0x565108899000
openat(AT_FDCWD, "/lib2/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|0777, st_size=15592, ...},
AT_EMPTY_PATH) = 0
getcwd("/mnt/c/Users/koche/mai/osi/laba4", 128) = 33
mmap(NULL, 16432, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =
0x7f829e9f8000
mmap(0x7f829e9f9000, 4096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) =
0x7f829e9f9000
mmap(0x7f829e9fa000, 4096, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f829e9fa000
mmap(0x7f829e9fb000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f829e9fb000
close(3) = 0
openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=19839, ...},
AT_EMPTY_PATH) = 0
mmap(NULL, 19839, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f829e7c8000
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\0\0\0\0\0\0"..., 832) =
832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=940560, ...},
AT_EMPTY_PATH) = 0
mmap(NULL, 942344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0)
= 0x7f829e6e1000
mmap(0x7f829e6ef000, 507904, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe000) = 0x7f829e6ef000
mmap(0x7f829e76b000, 372736, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x8a000) =
0x7f829e76b000
mmap(0x7f829e7c6000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe4000) =
0x7f829e7c6000
close(3) = 0
mprotect(0x7f829e7c6000, 4096, PROT_READ) = 0
mprotect(0x7f829e9fb000, 4096, PROT_READ) = 0
munmap(0x7f829e7c8000, 19839) = 0
newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x4), ...},
AT_EMPTY_PATH) = 0
read(0, 1 2 2.5
"1 2 2.5\n", 1024) = 8
newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x4), ...},
AT_EMPTY_PATH) = 0
write(1, "Result: -0.099621\n", 18Result: -0.099621
) = 18
read(0, 2 2 2.5
"2 2 2.5\n", 1024) = 8
write(1, "Result: 2.500000\n", 17Result: 2.500000
) = 17
read(0, 0
"0\n", 1024) = 2
openat(AT_FDCWD, "/lib1/lib1.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\0\0\0\0\0\0"..., 832) =
832
newfstatat(3, "", {st_mode=S_IFREG|0777, st_size=15520, ...},
AT_EMPTY_PATH) = 0
getcwd("/mnt/c/Users/koche/mai/osi/laba4", 128) = 33
mmap(NULL, 16432, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) =

```

```

0x7f829e7c8000
mmap(0x7f829e7c9000, 4096, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x7f829e7c9000
mmap(0x7f829e7ca000, 4096, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f829e7ca000
mmap(0x7f829e7cb000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x7f829e7cb000
close(3) = 0
mprotect(0x7f829e7cb000, 4096, PROT_READ) = 0
read(0, 1 2 2.5
"1 2 2.5\n", 1024) = 8
write(1, "Result: -0.754731\n", 18Result: -0.754731
) = 18
read(0, 2 2 2.5
"2 2 2.5\n", 1024) = 8
write(1, "Result: 5.000000\n", 17Result: 5.000000
) = 17
read(0, "", 1024) = 0
munmap(0x7f829e7c8000, 16432) = 0
exit_group(0) = ?
+++ exited with 0 +++

```

```
$ strace -f ./prog1
execve("./prog1", ["/prog1"], 0x7ffe6af64358 /* 36 vars */) = 0
brk(NULL)                               = 0x564212edf000
arch_prctl(0x3001 /* ARCH_??? */, 0x7ffd14a10420) = -1 EINVAL (Invalid
argument)
mmap(NULL, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f92bab92000
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=19839, ...},
AT_EMPTY_PATH) = 0
mmap(NULL, 19839, PROT_READ, MAP_PRIVATE, 3, 0) = 0x7f92bab8d000
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\3\0>\0\1\0\0\0\0\0\0\0\0\0"..., 832) =
832
newfstatat(3, "", {st_mode=S_IFREG|0644, st_size=940560, ...},
```

```

AT_EMPTY_PATH) = 0
mmap(NULL, 942344, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0)
= 0x7f92baaa6000
mmap(0x7f92baab4000, 507904, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe000) = 0x7f92baab4000
mmap(0x7f92bab30000, 372736, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x8a000) =
0x7f92bab30000
mmap(0x7f92bab8b000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe4000) =
0x7f92bab8b000
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"..., 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
=340\2563\265?\356\25x\261\27\313A#\350"..., 68, 896) = 68
newfstatat(3, "", {st_mode=S_IFREG|0755, st_size=2216304, ...},
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, 2260560, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3,
0) = 0x7f92ba87e000
mmap(0x7f92ba8a6000, 1658880, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) =
0x7f92ba8a6000
mmap(0x7f92baa3b000, 360448, PROT_READ,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1bd000) =
0x7f92baa3b000
mmap(0x7f92baa93000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x214000) =
0x7f92baa93000
mmap(0x7f92baa99000, 52816, PROT_READ|PROT_WRITE
, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x7f92baa99000
close(3) = 0
mmap(NULL, 12288, PROT_READ|PROT_WRITE,

```

```

MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x7f92ba87b000
arch_prctl(ARCH_SET_FS, 0x7f92ba87b740) = 0
set_tid_address(0x7f92ba87ba10)      = 108406
set_robust_list(0x7f92ba87ba20, 24)  = 0
rseq(0x7f92ba87c0e0, 0x20, 0, 0x53053053) = 0
mprotect(0x7f92baa93000, 16384, PROT_READ) = 0
mprotect(0x7f92bab8b000, 4096, PROT_READ) = 0
mprotect(0x56421163c000, 4096, PROT_READ) = 0
mprotect(0x7f92babcc000, 8192, PROT_READ) = 0
prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024,
rlim_max=RLIM64_INFINITY}) = 0
munmap(0x7f92bab8d000, 19839)      = 0
newfstatat(0, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x4), ...},
AT_EMPTY_PATH) = 0
getrandom("\xd5\xec\x99\x01\xfe\xdd\xdb\x75", 8, GRND_NONBLOCK) = 8
brk(NULL)      = 0x564212edf000
brk(0x564212f00000)      = 0x564212f00000
read(0, 1 2 2.5
"1 2 2.5\n", 1024)      = 8
newfstatat(1, "", {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x4), ...},
AT_EMPTY_PATH) = 0
write(1, "Result: -0.754731\n", 18Result: -0.754731
)      = 18
read(0, 2 2 2.5
"2 2 2.5\n", 1024)      = 8
write(1, "Result: 5.000000\n", 17Result: 5.000000
)      = 17
read(0, "", 1024)      = 0
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.