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

**Факультет информационных технологий и прикладной математики Кафедра вычислительной математики и программирования**

# Лабораторная работа № 5 по курсу

**«Операционные системы»**

Студент: Степанов Данила Михайлович

Группа: 8О-207Б Преподаватель: Е. С. Миронов Вариант: 35

Дата:

Оценка:

Москва, 2021

# Лабораторная работа №5

## 2. 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

Вариант задания: 35. Функции: подсчет площади плоской геометрической фигуры по двум сторонам исортировка целочисленного массива.

## Описание

Статическая библиотека состоит из подпрограмм, которые непосредственно компилируются и линкуются с вашей программой. При компиляции программы, которая использует статическую библиотеку, весь функционал статической библиотеки (тот, что использует ваша программа) становится частью вашего исполняемого файла. Динамическая же библиотека состоит из подпрограмм, которые подгружаются в вашу программу во время её выполнения. При компиляции программы, которая использует динамическую библиотеку, эта библиотека не становится частью вашего исполняемого файла — она так и остается отдельным модулем. Для написания лабораторной работы я буду использовать Makefile: для первой программы для получения объектных файлов и их дальнейшей линковки(использование динамической библиотеке со знаниями,полученными на этапе линковки), а во втором для загрузки динамической библиотеки во время исполнения программы.

## Исходный код

### Makefile

all:

gcc -shared -fPIC -o lib1.so lib1.c gcc -shared -fPIC -o lib2.so lib2.c

Prog1: main.o lib1.o

gcc main.o lib1.o -o Prog1

Prog2: din\_main.c

gcc din\_main.c -o Prog2 -ldl

main.o: lib1.o

gcc -c main.c lib1.o: lib1.c

gcc -c lib1.c

clean:

rm -f \*.o

### lib1.c

#include "lib.h" #include <stdbool.h>

float Square(float A, float B)

{

return A \* B;

}

int\* Sort(int\* array, int size)

{

int tmp;

bool noSwap;

for (int i = size - 1; i >= 0; i--)

{

noSwap = 1;

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

{

if (array[j] > array[j + 1])

{

tmp = array[j]; array[j] = array[j + 1]; array[j + 1] = tmp; noSwap = 0;

}

}

if (noSwap == 1) break;

}

}

### lib2.c

#include "lib.h"

float Square(float A, float B)

{

return (A \* B) / 2;

}

void swap(int\* a, int\* b)

{

int tmp; tmp = \*b;

\*b = \*a;

\*a = tmp;

}

void quick\_sort(int\* a, int l, int r)

{

if (r - l + 1 >= 2) { int i = l;

int j = r;

int x = a[(i + j) / 2]; while (i < j) {

while (a[i] < x) i++; while (a[j] > x) j--; if (i < j) {

swap(&a[i], &a[j]);

} i++; j--;

}

quick\_sort(a, i, r); quick\_sort(a, l, j);

}

}

int\* Sort(int\* array, int size)

{

quick\_sort(array, 0, size - 1);

}

### main.c

#include <stdio.h> #include <stdbool.h> #include "lib.h"

int main()

{

int cmd;

while (scanf("%d", &cmd) != EOF) { if (cmd == 1) {

int n; scanf("%d", &n); int a[n];

for (int i = 0; i < n; i++) { scanf("%d", &a[i]);

}

Sort(a, n);

for (int i = 0; i < n; i++) { printf("%d ", a[i]);

} printf("\n");

} else if (cmd == 2) { float a, b;

scanf("%f %f", &a, &b);

printf("%f\n", Square(a, b));

} else {

printf("Unknow command\n");

}

}

return 0;

}

1. **din\_main.c** #include <stdio.h> #include <stdbool.h> #include <dlfcn.h> #include <stdlib.h>

int\* (\*Sort)(int\*, int) = NULL;

float (\*Square)(float, float) = NULL;

char\* Libname1 = "../Library/lib1.so"; char\* Libname2 = "../Library/lib2.so";

char\* cur\_name; void\* cur\_handle;

void load\_functions(void\* handle)

{

Sort = dlsym(handle, "Sort"); Square = dlsym(handle, "Square");

}

void\* load()

{

void\* handle = dlopen(Libname1, RTLD\_LAZY); if (handle == NULL) {

printf("Open dl error\n"); exit(1);

}

load\_functions(handle); return handle;

}

void unload(void\* handle)

{

int check = dlclose(handle); if (check != 0) {

printf("Error closing dl\n"); exit(2);

}

}

void reload(void\* handle)

{

if (cur\_name == Libname1) { unload(handle); cur\_name = Libname2;

} else {

unload(handle); cur\_name = Libname1;

}

void\* new\_handle = dlopen(cur\_name, RTLD\_LAZY); if (new\_handle == NULL) {

printf("Open dl error\n"); exit(1);

}

load\_functions(new\_handle);

}

int main()

{

cur\_handle = load(); cur\_name = Libname1; int cmd;

while (scanf("%d", &cmd) != EOF) { if (cmd == 0) {

reload(cur\_handle); printf("Contract changed\n");

} else if (cmd == 1) { int n;

scanf("%d", &n); int a[n];

for (int i = 0; i < n; i++) { scanf("%d", &a[i]);

}

Sort(a, n);

for (int i = 0; i < n; i++) { printf("%d ", a[i]);

} printf("\n");

} else if (cmd == 2) { float a, b;

scanf("%f %f", &a, &b);

printf("%f\n", Square(a, b));

} else {

printf("Unknown command\n");

}

}

dlclose(cur\_handle); return 0;

}

## Тестирование программы и вывод команды strace

danila@danscomp:~/Документы/Institute/2 курс/ОС/lab5/Library$ ./Prog2 1

5

5 4 3 2 1

1 2 3 4 5

2 3 5

15.000000

4

Unknown command 0

Contract changed 2 3 5

7.500000

2 10 10

50.000000

1

10

29 89 -5 6 3 56 34 -113 2 76

-113 -5 2 3 6 29 34 56 76 89

0

Contract changed 2 4 8

32.000000

danila@danscomp:~/Документы/Institute/2 курс/ОС/lab5/Library$ strace ./Prog2 execve("./Prog2", ["./Prog2"], 0x7ffc07f58d60 /\* 54 vars \*/) = 0

brk(NULL) = 0x5653be201000

arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7ffcd72ce040) = -1 EINVAL (Недопустимый аргумент)

access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (Нет такого файла или каталога) openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3

fstat(3, {st\_mode=S\_IFREG|0644, st\_size=110089, ...}) = 0

mmap(NULL, 110089, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7fbaa3682000

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\0\3\0>\0\1\0\0\0 \22\0\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) = 0x7fbaa3680000

mmap(NULL, 20752, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fbaa367a000

mmap(0x7fbaa367b000, 8192, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x1000) = 0x7fbaa367b000

mmap(0x7fbaa367d000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x3000) = 0x7fbaa367d000

mmap(0x7fbaa367e000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x3000) = 0x7fbaa367e000

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\0\360q\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\20\0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0", 32, 848) = 32 pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\363\377?\332\200\270\27\304d\245n\355Y\377\ t\334"..., 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\0", 32, 848) = 32 pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0\363\377?\332\200\270\27\304d\245n\355Y\377\ t\334"..., 68, 880) = 68

mmap(NULL, 2036952, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fbaa3488000

mprotect(0x7fbaa34ad000, 1847296, PROT\_NONE) = 0 mmap(0x7fbaa34ad000, 1540096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE| MAP\_FIXED|MAP\_DENYWRITE, 3, 0x25000) = 0x7fbaa34ad000 mmap(0x7fbaa3625000, 303104, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x19d000) = 0x7fbaa3625000

mmap(0x7fbaa3670000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE| MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1e7000) = 0x7fbaa3670000 mmap(0x7fbaa3676000, 13528, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE| MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7fbaa3676000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, - 1, 0) = 0x7fbaa3485000

arch\_prctl(ARCH\_SET\_FS, 0x7fbaa3485740) = 0 mprotect(0x7fbaa3670000, 12288, PROT\_READ) = 0

mprotect(0x7fbaa367e000, 4096, PROT\_READ) = 0

mprotect(0x5653bde46000, 4096, PROT\_READ) = 0

mprotect(0x7fbaa36ca000, 4096, PROT\_READ) = 0

munmap(0x7fbaa3682000, 110089) = 0 brk(NULL) = 0x5653be201000 brk(0x5653be222000) = 0x5653be222000

openat(AT\_FDCWD, "../Library/lib1.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@\20\0\0\0\0\0\0"..., 832) = 832 fstat(3, {st\_mode=S\_IFREG|0775, st\_size=15672, ...}) = 0

getcwd("/home/danila/\ 320\224\320\276\320\272\321\203\320\274\320\265\320\275\321\202\321\213/Institute/2 \

320\272\321\203\321\200\321\201/\320\236\320\241/lab5/Library", 128) = 71

mmap(NULL, 16424, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fbaa3698000

mmap(0x7fbaa3699000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x1000) = 0x7fbaa3699000

mmap(0x7fbaa369a000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x2000) = 0x7fbaa369a000

mmap(0x7fbaa369b000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x2000) = 0x7fbaa369b000

close(3) = 0

mprotect(0x7fbaa369b000, 4096, PROT\_READ) = 0

fstat(0, {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0x1), ...}) = 0 read(0, 1

"1\n", 1024) = 2

read(0, 5 5 4 3 2 1

"5 5 4 3 2 1\n", 1024) = 12

fstat(1, {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0x1), ...}) = 0 write(1, "1 2 3 4 5 \n", 111 2 3 4 5

) = 11

read(0, 0

"0\n", 1024) = 2

munmap(0x7fbaa3698000, 16424) = 0

openat(AT\_FDCWD, "../Library/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\200\20\0\0\0\0\0\0"..., 832) = 832 fstat(3, {st\_mode=S\_IFREG|0775, st\_size=16088, ...}) = 0

getcwd("/home/danila/\ 320\224\320\276\320\272\321\203\320\274\320\265\320\275\321\202\321\213/Institute/2 \

320\272\321\203\321\200\321\201/\320\236\320\241/lab5/Library", 128) = 71

mmap(NULL, 16440, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fbaa3698000

mmap(0x7fbaa3699000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x1000) = 0x7fbaa3699000

mmap(0x7fbaa369a000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x2000) = 0x7fbaa369a000

mmap(0x7fbaa369b000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x2000) = 0x7fbaa369b000

close(3) = 0

mprotect(0x7fbaa369b000, 4096, PROT\_READ) = 0

write(1, "Contract changed\n", 17Contract changed

) = 17

read(0, 0

"0\n", 1024) = 2

munmap(0x7fbaa3698000, 16440) = 0

openat(AT\_FDCWD, "../Library/lib1.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@\20\0\0\0\0\0\0"..., 832) = 832 fstat(3, {st\_mode=S\_IFREG|0775, st\_size=15672, ...}) = 0

getcwd("/home/danila/\ 320\224\320\276\320\272\321\203\320\274\320\265\320\275\321\202\321\213/Institute/2 \

320\272\321\203\321\200\321\201/\320\236\320\241/lab5/Library", 128) = 71

mmap(NULL, 16424, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) =

0x7fbaa3698000

mmap(0x7fbaa3699000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x1000) = 0x7fbaa3699000

mmap(0x7fbaa369a000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x2000) = 0x7fbaa369a000

mmap(0x7fbaa369b000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED| MAP\_DENYWRITE, 3, 0x2000) = 0x7fbaa369b000

close(3) = 0

mprotect(0x7fbaa369b000, 4096, PROT\_READ) = 0 write(1, "Contract changed\n", 17Contract changed

) = 17

read(0, 2 3 5

"2 3 5\n", 1024) = 6

write(1, "15.000000\n", 1015.000000

) = 10

read(0, ^Cstrace: Process 45181 detached

<detached ...>

## Анализ использования библиотек двумя способами

Так как при первом способе библиотека встраивается в программу, размер файла увеличивается, а изменение частей библиотеки приводит к необходимости компилировать все файлы заново. Второй способ не имеет подобных недостатков и имеет ряд преимуществ: множество программ могут использовать один экземпляр библиотеки, загруженный в память.

## Вывод

В лабораторной работе были определены основные отличия статических и динамических библиотек и преимущества и недостатки использования динамических двумя способами: статической линковкой и во время исполнения программы.

Учитывая явные преимущества второго способа, становится очевидным его использование в крупных проектах.