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: М8О-209БВ-24

Студент: Касаева Я.М.

Преподаватель: Миронов Е.С.

Оценка: _____

Дата: 15.12.25

Москва, 2025

Постановка задачи

Вариант 28.

Рассчет значения числа Пи при заданной длине ряда (K), подсчет площади плоской геометрической фигуры по двум сторонам

Общий метод и алгоритм решения

Использованные системные вызовы:

- dlopen() - загрузка динамической библиотеки в память

- dlsym() - получение указателя на функцию из библиотеки

- dlclose() - выгрузка библиотеки из памяти

- dlerror() - получение информации об ошибках работы с библиотеками

- gcc с флагами -fPIC -shared - создание динамических библиотек

- gcc с флагом -ldl - линковка с библиотекой динамической загрузки

- gcc с флагом -l<library> - статическая линковка с библиотекой

Алгоритм работы программы:

Создание библиотек: компиляция исходников в libmath_impl1.so и libmath_impl2.so

Программа 1 (статическая линковка):

- Компиляция с привязкой к libmath_impl1.so

- Чтение команд: "1 K" для Pi, "2 A B" для Square

- Вызов функций через статические ссылки

Программа 2 (динамическая загрузка):

- Загрузка библиотеки через dlopen()

- Получение указателей на функции через dlsym()

- Обработка команд: "1 K", "2 A B", "0" для переключения библиотек

- Переключение реализаций через dlclose() и dlopen()

- Освобождение ресурсов через dlclose()

Код программы

math_functions.h

```
#ifndef MATH_FUNCTIONS_H
#define MATH_FUNCTIONS_H
```

```
float Pi(int K);
float Square(float A, float B);

#endif
```

math_impl1.c

```
#include "math_functions.h"

float Pi(int K) {
    float s = 0;
    for (int i = 0; i < K; i++) {
        float t = 1.0 / (2*i + 1);
        if (i % 2) t = -t;
        s += t;
    }
    return 4 * s;
}

float Square(float A, float B) {
    return A * B;
}
```

math_impl2.c

```
#include "math_functions.h"

float Pi(int K) {
    float p = 1.0;
    for (int i = 1; i <= K; i++) {
        p *= (4.0*i*i) / (4.0*i*i - 1);
    }
    return 2 * p;
}

float Square(float A, float B) {
    return 0.5 * A * B;
}
```

program1.c

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "math_functions.h"

int main() {
    char cmd[100];

    printf("1 K - Pi(K)\n2 A B - Square(A,B)\n0 - exit\n");

    while (1) {
        printf("> ");
        if (!fgets(cmd, 100, stdin)) break;

        if (cmd[0] == '0') break;

        if (cmd[0] == '1') {
            int K = atoi(cmd + 2);
            printf("Pi: %f\n", Pi(K));
        }
        else if (cmd[0] == '2') {
```

```
            float A, B;
            sscanf(cmd + 2, "%f %f", &A, &B);
            printf("Square: %f\n", Square(A, B));
        }
    }

    return 0;
}
```

program2.c

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

typedef float (*FuncPi)(int);
typedef float (*FuncSquare)(float, float);

int main() {
    void *lib = dlopen("./libmath_impl1.so", RTLD_LAZY);
    FuncPi Pi = (FuncPi)dlsym(lib, "Pi");
    FuncSquare Square = (FuncSquare)dlsym(lib, "Square");
    int lib_num = 1;

    char cmd[100];
    printf("1 K - Pi\n2 A B - Square\n0 - switch lib\n3 - exit\n> ");

    while (fgets(cmd, 100, stdin)) {

        if (cmd[0] == '3') break;

        if (cmd[0] == '0') {
            dlclose(lib);
            lib_num = (lib_num == 1) ? 2 : 1;
            char lib_name[30];
            sprintf(lib_name, "./libmath_impl%d.so", lib_num);
            lib = dlopen(lib_name, RTLD_LAZY);
            Pi = (FuncPi)dlsym(lib, "Pi");
            Square = (FuncSquare)dlsym(lib, "Square");
            printf("Switched to lib %d\n> ", lib_num);
            continue;
        }

        if (cmd[0] == '1') {
            int K = atoi(cmd + 2);
            printf("Pi: = %f (lib %d)\n", Pi(K), lib_num);
        }
        else if (cmd[0] == '2') {
            float A, B;
            sscanf(cmd + 2, "%f %f", &A, &B);
            printf("Square: %f (lib %d)\n", Square(A, B), lib_num);
        }
        printf("> ");
    }

    dlclose(lib);
    return 0;
}
```

Makefile

```
CC = gcc
CFLAGS = -fPIC
```

```
all: libmath_impl1.so libmath_impl2.so program1 program2

libmath_impl1.so: math_impl1.c math_functions.h
	$(CC) $(CFLAGS) -shared -o $@ $<

libmath_impl2.so: math_impl2.c math_functions.h
	$(CC) $(CFLAGS) -shared -o $@ $<

program1: program1.c
	$(CC) -o $@ $< -L. -lmath_impl1 -Wl,-rpath,.

program2: program2.c
	$(CC) -o $@ $< -ldl

clean:
	rm -f *.so program1 program2
```

Протокол работы программы

Тестирование 1:

(base) yanakasaeva@MacBook-Air--YanaK src % ./program1

1 K - Pi(K)

2 A B - Square(A,B)

0 - exit

> 1 100

Pi: 3.131593

> 2 3 4

Square: 12.000000

> 0

Тестирование 2:

(base) yanakasaeva@MacBook-Air--YanaK src % ./program2

1 K - Pi

2 A B - Square

0 - switch lib

3 - exit

> 1 10

Pi: = 3.041840 (lib 1)

> 0

Switched to lib 2

> 1 100

Pi: = 3.133787 (lib 2)

> 2 3 4

Square: 6.000000 (lib 2)

> 0

Switched to lib 1

> 3

Strace:

Program1

1113 execve("./program1", ["./program1"], 0xffffd67732f8 /* 8 vars */) = 0

1113 brk(NULL) = 0xaaaabd14d000

1113 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff9a4c8000

1113 faccessat(AT_FDCWD, "/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)

1113 openat(AT_FDCWD, "./libmath_impl1.so", O_RDONLY|O_CLOEXEC) = 3

1113 read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

1113 fstat(3, {st_mode=S_IFREG|0755, st_size=69272, ...}) = 0

1113 mmap(NULL, 196640, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_DENYWRITE, -1, 0) = 0xffff9a45e000

1113 mmap(0xffff9a460000, 131104, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff9a460000

1113 munmap(0xffff9a45e000, 8192) = 0

1113 mprotect(0xffff9a461000, 122880, PROT_NONE) = 0

1113 mmap(0xffff9a47f000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xf000) = 0xffff9a47f000

1113 close(3) = 0

1113 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

1113 mmap(NULL, 8467, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffff9a4c5000

1113 close(3) = 0

1113 openat(AT_FDCWD, "/lib/aarch64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

1113 mmap(NULL, 1892240, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_DENYWRITE, -1, 0) = 0xffff9a292000

1113 mmap(0xffff9a2a0000, 1826704, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffff9a2a0000

1113 mprotect(0xffff9a439000, 81920, PROT_NONE) = 0

1113 mmap(0xffff9a44d000, 20480, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19d000) = 0xffff9a44d000

1113 mmap(0xffff9a452000, 49040, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffff9a452000

1113 close(3) = 0

1113 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffff9a4c3000

1113 write(1, "1 K - Pi(K)\n", 12) = 12

1113 write(1, "2 A B - Square(A,B)\n", 20) = 20

1113 write(1, "0 - exit\n", 9) = 9

1113 read(0, "1 100\n2 3 4\n0\n", 4096) = 14

1113 write(1, "> Pi: 3.131593\n", 15) = 15

1113 write(1, "> Square: 12.000000\n", 20) = 20

1113 exit_group(0) = ?

1113 +++ exited with 0 +++

Program2

1117 execve("./program2", ["./program2"], 0xffffe83a48e8 /* 8 vars */) = 0

1117 brk(NULL) = 0xaaaae896b000

1117 mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xffffa06dc000

1117 faccessat(AT_FDCWD, "/etc/ld.so.preload", R_OK) = -1 ENOENT (No such file or directory)

1117 openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

1117 fstat(3, {st_mode=S_IFREG|0644, st_size=8467, ...}) = 0

1117 mmap(NULL, 8467, PROT_READ, MAP_PRIVATE, 3, 0) = 0xffffa06d9000

1117 close(3) = 0

1117 openat(AT_FDCWD, "/lib/aarch64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

1117 read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\360\206\2\0\0\0\0"..., 832) = 832

1117 fstat(3, {st_mode=S_IFREG|0755, st_size=1722920, ...}) = 0

1117 mmap(NULL, 1892240, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_DENYWRITE, -1, 0) = 0xffffa04d5000

1117 mmap(0xffffa04e0000, 1826704, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa04e0000

1117 munmap(0xffffa04d5000, 45056) = 0

1117 mprotect(0xffffa0679000, 81920, PROT_NONE) = 0

1117 mmap(0xffffa068d000, 20480, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x19d000) = 0xffffa068d000

1117 mmap(0xffffa0692000, 49040, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0xffffa0692000

1117 close(3) = 0

1117 munmap(0xffffa06d9000, 8467) = 0

1117 openat(AT_FDCWD, "./libmath_impl1.so", O_RDONLY|O_CLOEXEC) = 3

1117 read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

1117 mmap(NULL, 196640, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_DENYWRITE, -1, 0) = 0xffffa04af000

1117 mmap(0xffffa04b0000, 131104, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa04b0000

1117 munmap(0xffffa04af000, 4096) = 0

1117 mprotect(0xffffa04b1000, 122880, PROT_NONE) = 0

1117 mmap(0xffffa04cf000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xf000) = 0xffffa04cf000

1117 close(3) = 0

1117 mprotect(0xffffa04cf000, 4096, PROT_READ) = 0

1117 write(1, "1 K - Pi\n", 9) = 9

1117 write(1, "2 A B - Square\n", 15) = 15

1117 write(1, "0 - switch lib\n", 15) = 15

1117 write(1, "3 - exit\n", 9) = 9

1117 read(0, "1 10\n0\n1 100\n2 3 4\n0\n3\n", 4096) = 23

1117 write(1, "> Pi: = 3.041840 (lib 1)\n", 25) = 25

1117 munmap(0xffffa04b0000, 131104) = 0

1117 openat(AT_FDCWD, "./libmath_impl2.so", O_RDONLY|O_CLOEXEC) = 3

1117 read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

1117 mmap(NULL, 196640, PROT_NONE, MAP_PRIVATE|MAP_ANONYMOUS|MAP_DENYWRITE, -1, 0) = 0xffffa04af000

```
**1117  mmap**(0xffffa04b0000, 131104, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa04b0000

**1117  munmap**(0xffffa04af000, 4096)      = 0

**1117  mprotect**(0xffffa04b1000, 122880, PROT_NONE) = 0

**1117  mmap**(0xffffa04cf000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xf000) = 0xffffa04cf000

**1117  close**(3)                      = 0

**1117  mprotect**(0xffffa04cf000, 4096, PROT_READ) = 0

**1117  write**(1, "> Switched to lib 2\n", 20) = 20

**1117  write**(1, "> Pi: = 3.133787 (lib 2)\n", 25) = 25

**1117  write**(1, "> Square: 6.000000 (lib 2)\n", 27) = 27

**1117  munmap**(0xffffa04b0000, 131104)    = 0

**1117  openat**(AT_FDCWD, "./libmath_impl1.so", O_RDONLY|O_CLOEXEC) = 3

**1117  read**(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0\267\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) =
832

**1117  mmap**(NULL, 196640, PROT_NONE,
MAP_PRIVATE|MAP_ANONYMOUS|MAP_DENYWRITE, -1, 0) = 0xffffa04af000

**1117  mmap**(0xffffa04b0000, 131104, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0) = 0xffffa04b0000

**1117  munmap**(0xffffa04af000, 4096)      = 0

**1117  mprotect**(0xffffa04b1000, 122880, PROT_NONE) = 0

**1117  mmap**(0xffffa04cf000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xf000) = 0xffffa04cf000

**1117  close**(3)                      = 0

**1117  mprotect**(0xffffa04cf000, 4096, PROT_READ) = 0

**1117  write**(1, "> Switched to lib 1\n", 20) = 20

**1117  munmap**(0xffffa04b0000, 131104)    = 0

**1117  write**(1, "> ", 2)                = 2

**1117  exit_group**(0)                     = ?

1117  +++ exited with 0 +++
```

Вывод

В ходе лабораторной работы были изучены два способа использования динамических библиотек в C. Первый способ - статическая линковка (program1) - подключает библиотеку на этапе компиляции. Второй способ - динамическая загрузка (program2) - загружает библиотеку во время

выполнения. Динамическая загрузка позволяет переключать реализации функций "на лету" командой "0", что обеспечивает гибкость. Однако это требует дополнительных системных вызовов openat, mmap, munmap при каждом переключении. Статическая линковка проще и быстрее, но не позволяет менять реализацию без перекомпиляции. Оба метода корректно работают с математическими функциями.