

Министерство цифрового развития, связи
и массовых коммуникаций Российской Федерации

Сибирский государственный университет
телекоммуникаций и информатики

Кафедра прикладной математики и кибернетики

ЛАБОРАТОРНАЯ РАБОТА №7

По дисциплине: «Операционные системы»

Выполнили:

Студенты 3 курса группы ИП-111
Корнилов А.А.,
Попов М.И.,
Толкач А.А.

Проверил:

Профессор кафедры ПМиК
Малков Е.А.

Новосибирск, 2023

Задание: разработайте программу с динамической загрузкой библиотеки совместного доступа, созданную в лабораторной 5. Проконтролируйте загрузку и выгрузку библиотеки, используя файл /proc/<идентификатор процесса>/maps

Цель: знакомство со структурой адресного пространства процесса

Выполнение работы:

Для выполнения лабораторной работы была взята программа №5 лабораторной работы (листинг 1).

```
#include <stdio.h>
#include "lab05.h"

int main() {
    system("clear");
    int choice;
    while (1) {
        printf("\nМеню:\n");
        printf("1. Добавить запись\n");
        printf("2. Редактировать запись\n");
        printf("3. Удалить запись\n");
        printf("4. Поиск записи\n");
        printf("5. Просмотр всех записей\n");
        printf("6. Экспорт базы данных в файл\n");
        printf("7. Импорт базы данных из файла\n");
        printf("0. Выйти\n");
        printf("Выберите действие (1-0): ");
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                addRecord();
                break;
            case 2:
                editRecord();
                break;
            case 3:
                deleteRecord();
                break;
            case 4:
                searchRecord();
                break;
            case 5:
                viewAllRecords();
                break;
            case 6:
                exportDatabaseToFile();
                break;
            case 7:
                importDatabaseFromFile();
                break;
            case 0:
                printf("Выход из программы.\n");
                return 0;
            default:
```

```

        printf("Неверный выбор. Попробуйте снова.\n");
    }
}
return 0;
}

```

Листинг 1 – программа lab05.c

Для динамической загрузки библиотек необходимо использовать библиотеку `<dlfcn.h>`, необходимо открыть `.so` файл с флагом «`RTLD_LAZY`» для разрешения неопределенных символов в коде.

```

void* handle = dlopen("./liblab05.so", RTLD_LAZY);
if (!handle) {
    fprintf(stderr, " нету .so файла, проверьте название файла \n", dlerror());
    return 1;
};

```

Далее необходимо вызвать для каждого выбора в меню вызывать необходимую функцию из библиотеки, для этого создаем указатель на не типизированную функцию и в него заносим в помощью «`dlsym`» указывавшем куда мы открыли `.so` файл и какую функцию хотим вызвать. Так делаем для всего меню.

```

void* (*func1)();
/*-----сокращение кода функции-----*/
switch (choice) {
case 1:
    func1 = dlsym(handle, "addRecord");
    (*func1)();
    break;

```

В конце не забываем закрывать библиотеку «`dlclose(handle);`».

При компиляции и компоновке не забываем указать флаг библиотеки `-ldl`.

```

.PHONY: all
all:
    gcc -Wall -ldl lab07s.c -o lab7lazy

```

Проверяем загрузку и выгрузку библиотеки с помощью `map` файла

```

miron@DESKTOP-UMC1Q46:/$ cat ./proc/3421/maps
55c100799000-55c10079a000 r--p 00000000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079a000-55c10079b000 r-xp 00001000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079b000-55c10079c000 r--p 00002000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079c000-55c10079d000 r--p 00002000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079d000-55c10079e000 rw-p 00003000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079e000-55c1007a1000 rw-p 00000000 00:00 0
55c1024c7000-55c1024e8000 rw-p 00000000 00:00 0 [heap]
7fee29095000-7fee29098000 rw-p 00000000 00:00 0
7fee29098000-7fee290c0000 r--p 00000000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee290c0000-7fee29255000 r-xp 00028000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee29255000-7fee292ad000 r--p 001bd000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee292ad000-7fee292b1000 r--p 00214000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee292b1000-7fee292b3000 rw-p 00218000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee292b3000-7fee292c0000 rw-p 00000000 00:00 0
7fee292c0000-7fee292c8000 rw-p 00000000 00:00 0
7fee292c8000-7fee292ca000 r--p 00000000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee292ca000-7fee292f4000 r-xp 00002000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee292f4000-7fee292ff000 r--p 0002c000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee29300000-7fee29302000 r--p 00037000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee29302000-7fee29304000 rw-p 00039000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fff42da0000-7fff42dc1000 rw-p 00000000 00:00 0 [stack]
7fff42dc1000-7fff42dc5000 r--p 00000000 00:00 0 [vvar]
7fff42dc5000-7fff42dc7000 r-xp 00000000 00:00 0 [vdso]

```

До загрузки .so файла

```

miron@DESKTOP-UMC1Q46:/$ cat ./proc/3421/maps
55c100799000-55c10079a000 r--p 00000000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079a000-55c10079b000 r-xp 00001000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079b000-55c10079c000 r--p 00002000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079c000-55c10079d000 r--p 00002000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079d000-55c10079e000 rw-p 00003000 00:53 5910974511035678 /mnt/u/Documents/B BY3/OS/7/lazy/lab7lazy
55c10079e000-55c1007a1000 rw-p 00000000 00:00 0
55c1024c7000-55c1024e8000 rw-p 00000000 00:00 0 [heap]
7fee2908d000-7fee2908e000 r--p 00000000 00:53 6192449487746341 /mnt/u/Documents/B BY3/OS/7/lazy/liblab05.so
7fee2908e000-7fee2908f000 r-xp 00001000 00:53 6192449487746341 /mnt/u/Documents/B BY3/OS/7/lazy/liblab05.so
7fee2908f000-7fee29090000 r--p 00002000 00:53 6192449487746341 /mnt/u/Documents/B BY3/OS/7/lazy/liblab05.so
7fee29090000-7fee29091000 r--p 00002000 00:53 6192449487746341 /mnt/u/Documents/B BY3/OS/7/lazy/liblab05.so
7fee29091000-7fee29092000 rw-p 00003000 00:53 6192449487746341 /mnt/u/Documents/B BY3/OS/7/lazy/liblab05.so
7fee29092000-7fee29098000 rw-p 00000000 00:00 0
7fee29098000-7fee290c0000 r--p 00000000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee290c0000-7fee29255000 r-xp 00028000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee29255000-7fee292ad000 r--p 001bd000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee292ad000-7fee292b1000 r--p 00214000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee292b1000-7fee292b3000 rw-p 00218000 08:20 6521 /usr/lib/x86_64-linux-gnu/libc.so.6
7fee292b3000-7fee292c0000 rw-p 00000000 00:00 0
7fee292c0000-7fee292c8000 rw-p 00000000 00:00 0
7fee292c8000-7fee292ca000 r--p 00000000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee292ca000-7fee292f4000 r-xp 00002000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee292f4000-7fee292ff000 r--p 0002c000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee29300000-7fee29302000 r--p 00037000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fee29302000-7fee29304000 rw-p 00039000 08:20 6325 /usr/lib/x86_64-linux-gnu/ld-linux-x86-64.so.2
7fff42da0000-7fff42dc1000 rw-p 00000000 00:00 0 [stack]
7fff42dc1000-7fff42dc5000 r--p 00000000 00:00 0 [vvar]
7fff42dc5000-7fff42dc7000 r-xp 00000000 00:00 0 [vdso]

```

После загрузки

Итоговый листинг программы:

```

#include <stdio.h>
#include <dlfcn.h>
// #include "lab07.h"

struct Person {
    char name[50];

```

```

    int age;
    char address[100];
};
struct Person database[100];
int databaseSize = 0;
char* menu = "Меню";
char* n1 = "1. Добавить запись";
char* n2 = "2. Редактировать запись";
char* n3 = "3. Удалить запись";
char* n4 = "4. Поиск записи";
char* n5 = "5. Просмотр всех записей";
char* n6 = "6. Экспорт базы данных в файл";
char* n7 = "7. Импорт базы данных из файла";
char* n0 = "0. Выйти";
char* action = "Выберите действие (1-0): ";
char* exit_text = "Выход из программы.";
char* wrong_text = "Неверный выбор. Попробуйте снова.";

int main() {
    system("clear");
    printf("%s\n", "для загрузки .so нажмите Enter");
    getchar();
    void* handle = dlopen("./liblab05.so", RTLD_LAZY);
    if (!handle) {
        fprintf(stderr, " нету .so файла, проверьте название файла \n", dlerror());
        return 1;
    };

    void* (*func1)();

    int choice;
    while (1) {
        printf("\n%s\n", menu);
        printf("%s\n", n1);
        printf("%s\n", n2);
        printf("%s\n", n3);
        printf("%s\n", n4);
        printf("%s\n", n5);
        printf("%s\n", n6);
        printf("%s\n", n7);
        printf("%s\n", n0);
        printf("%s\n", action);
        scanf("%d", &choice);

        switch (choice) {
            case 1:
                func1 = dlsym(handle, "addRecord");
                (*func1)();
                break;
            case 2:
                func1 = dlsym(handle, "editRecord");
                (*func1)();
                break;
            case 3:
                func1 = dlsym(handle, "deleteRecord");
                (*func1)();
                break;
            case 4:
                func1 = dlsym(handle, "searchRecord");
                (*func1)();
                break;
            case 5:
                func1 = dlsym(handle, "viewAllRecords");
                (*func1)();
                break;
        }
    }
}

```

```

    case 6:
        func1 = dlsym(handle, "exportDatabaseToFile");
        (*func1)();
        break;
    case 7:
        func1 = dlsym(handle, "importDatabaseFromFile");
        (*func1)();
        break;
    case 0:
        printf("%s\n", exit_text);
        dlclose(handle);
        getchar();
        return 0;
    default:
        printf("%s\n", wrong_text);
    }
}
return 0;
}

```