

Министерство науки и высшего образования

Московский Авиационный Институт
(национальный исследовательский университет)

ЛАБОРАТОРНАЯ РАБОТА №5
по курсе операционные системы I семестр 2022/2023

Студент: Катин Иван Вячеславович

Группа: М8О-210Б-21

Вариант: 31

Преподаватель : Миронов Евгений Сергеевич

Оценка:

Дата:

Подпись:

Москва, 2022

Содержание

Постановка задачи	2
Цель работы	2
Алгоритм решения	2
Код программы	3
Тест кейсы	6
Вывод	7

Постановка задачи

Вариант 31.

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

В конечном итоге, в лабораторной работе необходимо получить следующие части:

- Динамические библиотеки, реализующие контракты, которые заданы вариантом;
- Тестовая программа (программа №1), которая использует одну из библиотек, используя знания полученные на этапе компиляции;
- Тестовая программа (программа №2), которая загружает библиотеки, используя только их местоположение и контракты.

Цель работы

Приобретение практических навыков в:

- Создание динамических библиотек
- Создание программ, которые используют функции динамических библиотек

Алгоритм решения

Имеем два исполняемых файла. В первом случае библиотека подключается к программе на этапе компиляции. Тут мы работаем с ней практически как с любой другой библиотекой. Во втором случае нам необходимо пользоваться средствами языка для открытия динамических библиотек и получения функций из них. В каждой библиотеке содержатся определения двух функций, кроме этого есть общий заголовочный файл. Сами функции - простые и сводятся к арифметике. В программе с динамическим использованием библиотеки нужно открывать файл динамической библиотеки, получать из него функции и переключаться между разными библиотеками с разными реализациями по команде пользователя. Используем переменную, которая позволяет понять, с какой библиотекой мы сейчас работаем. Перед закрытием программы нужно закрыть библиотеку.

Код программы

Заголовчный файл lib.h

```
1 //
2 // Created by machine on 01.12.2022.
3 //
4
5 #ifndef LAB5_LIB_H
6 #define LAB5_LIB_H
7
8 float e(int x);
9 float square(float a, float b);
10
11 #endif //LAB5_LIB_H
```

Файл с первой реализацией функций lib1

```
1 #include <math.h>
2 #include "lib.h"
3
4 float e(int x) {
5     if (x == 0) {
6         return 1;
7     }
8     return powf(1 + 1 / (float) x, x);
9 }
10
11 float square(float a, float b) {
12     return a * b;
13 }
```

Файл со второй реализацией функций lib2

```
1 #include "lib.h"
2
3 float factorial(int n) {
4     return (n < 2) ? 1 : n * factorial(n - 1);
5 }
6
7 float e(int x) {
8     float divisor = 1;
9     float answer = 0;
10    for (int step = 0; step < x + 1; ++step) {
11        answer += 1 / divisor;
12        divisor *= step + 1;
13    }
14    return answer;
15 }
16
17 float square(float a, float b){
18     return (0.5 * a * b);
19 }
```

Файл с динамической библиотекой dynamic

```
1 #include <stdio.h>
2 #include <stdbool.h>
3 #include <stdlib.h>
4 #include <dlfcn.h>
5
6 float (*e_function)(int x) = NULL;
7 float (*square_function)(float a, float b) = NULL;
8
9 const char* path_library1 = "libFirst.so";
10 const char* path_library2 = "libSecond.so";
11 void *library = NULL;
12 bool is_first_library = true;
13
14 void handle_error(bool expr, char* msg) {
15     if (expr) {
16         perror(msg);
17         exit(-1);
18     }
19 }
20
21 void open_library(const char *path_library) {
22     if (library != NULL) {
23         dlclose(library);
24     }
25
26     library = dlopen(path_library, RTLD_LAZY);
27     handle_error(library == NULL, "can't open library");
28
29     e_function = dlsym(library, "e");
30     square_function = dlsym(library, "square");
31     handle_error(e_function == NULL || square_function == NULL, "Can't find method");
32 }
33
34 void reloadLib() {
35     is_first_library ? open_library(path_library2) : open_library(path_library1);
36     is_first_library = !is_first_library;
37     printf("library changed\n");
38 }
39
40
41 int main() {
42     open_library(path_library1);
43     int query;
44     int arg_function1;
45     float arg1_function2;
46     float arg2_function2;
47     while (scanf("%d", &query) != EOF) {
48         if (query == 0){
49             reloadLib();
50         } else if (query == 1){
51             scanf("%d", &arg_function1);
52             printf("Result first function: %f\n", (*e_function)(arg_function1));
53         } else if (query == 2){
54             scanf("%f %f", &arg1_function2, &arg2_function2);
55             printf("Result second function: %f\n", (*square_function)(arg1_function2, arg2_function2));
56         } else {
57             printf("invalid query");
58         }
59     }
```

```

59 |     }
60 |     dlclose(library);
61 | }

```

Файл со статической библиотекой static

```

1 | #include <stdio.h>
2 | #include "lib.h"
3 |
4 | int main() {
5 |     int query;
6 |     int arg_function1;
7 |     float arg1_function2;
8 |     float arg2_function2;
9 |     while (scanf("%d", &query) != EOF) {
10 |         if (query == 1) {
11 |             scanf("%d", &arg_function1);
12 |             printf("Result first function: %f\n", e(arg_function1));
13 |         } else if (query == 2) {
14 |             scanf("%f %f", &arg1_function2, &arg2_function2);
15 |             printf("Result second function: %f\n", square(arg1_function2, arg2_function2));
16 |         } else {
17 |             printf("invalid query\n");
18 |         }
19 |     }
20 | }

```

Тест кейсы

```
machine@Turing:~/Desktop/MAI/os/lab5$ ./static.o
```

```
1
```

```
5
```

```
Result first function: 2.488321
```

```
2
```

```
12 3
```

```
Result second function: 36.000000
```

```
machine@Turing:~/Desktop/MAI/os/lab5$ ./dynamic.o
```

```
1
```

```
5
```

```
Result first function: 2.488321
```

```
2
```

```
12 3
```

```
Result second function: 36.000000
```

```
0
```

```
library changed
```

```
1
```

```
5
```

```
Result first function: 2.716667
```

```
2
```

```
12 3
```

```
Result second function: 18.000000
```

Вывод

В ходе лабораторной работы я познакомился с динамическими библиотеками. Основное различие динамических библиотек от статических в том, что код второй при компиляции полностью входит в состав исполняемого файла, что делает программу легко переносимой. Код динамических библиотек не входит в исполняемый файл, последний содержит лишь ссылку на библиотеку, однако файл занимает меньше места. Можно заметить, что сам код программ несложный и даже сказать, что лабораторная заточена на написание `makefile` и использование пары функций для открытия библиотеки.