

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: Знай Артемий Олегович
Группа: М8О-201Б-21
Вариант: 33
Преподаватель: Миронов Евгений Сергеевич
Оценка: _____
Дата: _____
Подпись: _____

Москва, 2022

Содержание

1. Репозиторий
2. Постановка задачи
3. Общие сведения о программе
4. Общий метод и алгоритм решения
5. Исходный код
6. Демонстрация работы программы
7. Выводы

Репозиторий

https://github.com/znako/OS_LABS/

Постановка задачи

Цель работы

Целью является приобретение практических навыков в:

- Создание динамических библиотек
- Создание программ, которые используют функции динамических библиотек
- Работа со сборочной системой

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

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.

Провести анализ двух типов использования библиотек.

Вариант:

6	Расчет значения числа e (основание натурального логарифма)	Float E(int x)	$(1 + 1/x)^x$	Сумма ряда по n от 0 до x , где элементы ряда равны: $(1/(n!))$
9	Отсортировать целочисленный массив	Int * Sort(int * array)	Пузырьковая сортировка	Сортировка Хоара

Общие сведения о программе

Динамические библиотеки с функциями компилируются из файлов lib1.c и lib2.c соответственно. Основные программы компилируются из файлов prog1.c и prog2.c с использованием заголовочного файла lab5.h (описание функций из динамических библиотек).

Во второй программе используются библиотечные вызовы

dlopen - открытие динамической библиотеки,

dlsym - получение функции

dlclose – закрытие.

Общий метод и алгоритм решения

Во второй программе при получении 0 на вход происходит следующее: закрывается текущая динамическая библиотека, открывается новая (имена библиотек берутся из массива размера N по формуле $(n + 1) \% N$, где n – текущая библиотека) и из неё загружаются функции вместо функций старой.

Исходный код

func1.c
<pre>#include "lab5.h" float power(int x) { float result = 1; for (int i = 1; i <= x; i++){ result *= (1+1.0/x); } return result; } float E(int x) { return power(x); } void Sort(int* array, unsigned long size) { for (unsigned long i = 0; i < size; ++i) { for (unsigned long j = 0; j < size - 1; ++j) { if (array[j] > array[j + 1]) { int temp = array[j]; array[j] = array[j + 1]; array[j + 1] = temp; } } } }</pre>

func2.c
<pre>#include "lab5.h" #include <stdlib.h> int Compare(const void* a, const void* b) { return (*(int*)a - *(int*)b); } float E(int x) { float e = 1.0; int factorial = 1; for (int i = 1; i <= x; i++) { factorial *= i; e += 1.0/factorial; } }</pre>

```

    }
    return e;
}

void Sort(int* array, unsigned long size) {
    qsort(array, size, sizeof(int), Compare);
}

```

prog1.c

```

#include<stdio.h>
#include"lab5.h"
#include<stdlib.h>

int main(){

    int choice;
    printf("1: calculate e;\n");
    printf("2: Bubble sort of array.\n");

    printf("Make a choice: ");

    while(scanf("%d", &choice) != EOF){
        switch (choice) {
            case 1:{

                printf("Enter x to calculate e = ");

                int x;
                scanf("%d", &x);

                float e = E(x);

                printf("e = %f\n", e);
                break;

            }
            case 2:{

                printf("Size of array = ");

                unsigned long size;
                scanf("%lu", &size);

                int* arr = malloc(size * sizeof(int));

                printf("Filling the array:\n");

                for (unsigned long i = 0; i < size; ++i){
                    scanf("%d", arr + i);
                }

                Sort(arr, size);

                printf("Sorted array: \n");
                for (unsigned long i = 0; i < size; ++i){
                    printf("%d ", arr[i]);
                }

                printf("\n");
                free(arr);
                break;

            }
            default:{

```

```

        printf("1: calculate e;\n");
        printf("2: Bubble sort of array.\n");
        break;

    }

}

}
return 0;
}

```

prog2.c

```

#include <stdio.h>
#include <dlfcn.h>
#include <stdlib.h>
#include <lab5.h>

int main(){

    int countLib = 2;
    char* namesLib[] = {"./libFirst.so", "./libSecond.so"};

    int numberOfLyb = 0;
    void* hdl;
    hdl = dlopen(namesLib[numberOfLyb], RTLD_LAZY);

    if (hdl == NULL){
        printf("dlopen error\n");
        return -1;
    }

    TypeOfE eType;
    TypeOfStort sortType;

    #pragma GCC diagnostic push
    #pragma GCC diagnostic ignored "-Wpedantic"
    eType = (TypeOfE)dlsym(hdl, "E");
    sortType = (TypeOfStort)dlsym(hdl, "Sort");
    #pragma GCC diagnostic pop

    int choice;
    printf("0: Switch lib;\n");
    printf("1: calculate e;\n");
    printf("2: Bubble sort of array.\n");

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

        switch (choice) {
            case 0:{

                numberOfLyb = (numberOfLyb + 1) % countLib;

                if (dlclose(hdl) != 0){
                    perror("dlclose error");
                    return -1;
                }

                if ((hdl = dlopen(namesLib[numberOfLyb], RTLD_LAZY)) == NULL){
                    printf("dlopen error\n");
                    return -1;
                }

                #pragma GCC diagnostic push
                #pragma GCC diagnostic ignored "-Wpedantic"

```

```

        eType = (TypeOfE)dlsym(hdl, "E");
        sortType = (TypeOfSort)dlsym(hdl, "Sort");
#pragma GCC diagnostic pop

        printf("Lib swithced. Used lib %d.\n", numberOfLib+1);
        break;
    }
    case 1:{

        printf("Enter x to calculate e = ");

        int x;
        scanf("%d", &x);

        float e = (*eType)(x);
        printf("e = %f\n", e);

        break;
    }
    case 2:{

        printf("Size of array = ");

        unsigned long size;
        scanf("%lu", &size);

        int* arr = malloc(size * sizeof(int));

        printf("Filling the array:\n");

        for (unsigned long i = 0; i < size; ++i){
            scanf("%d", &arr[i]);
        }

        (*sortType)(arr, size);

        printf("Sorted array: \n");
        for (unsigned long i = 0; i < size; ++i){
            printf("%d ", arr[i]);
        }

        printf("\n");
        free(arr);
        break;
    }
    default:{

        printf("0: Switch lib;\n");
        printf("1: calculate e;\n");
        printf("2: Bubble sort of array.\n");
        break;
    }
}

return 0;
}

```

Демонстрация работы программы

```
znako@znako-VirtualBox:~/UtoW/OS_LABS/build/lab5$ ./prog1L1
```

```
1: calculate e;
```

```
2: Bubble sort of array.
```

```
Make a choice: 1
```

```
Enter x to calculate e = 5
```

```
e = 2.488320
```

```
2
```

```
Size of array = 4
```

```
Filling the array:
```

```
2 4 5 6
```

```
Sorted array:
```

```
2 4 5 6
```

```
^C
```

```
znako@znako-VirtualBox:~/UtoW/OS_LABS/build/lab5$ ./prog1L1
```

```
1: calculate e;
```

```
2: Bubble sort of array.
```

```
Make a choice: 1
```

```
Enter x to calculate e = 5
```

```
e = 2.488320
```

```
2
```

```
Size of array = 4
```

```
Filling the array:
```

```
1
```

```
2
```

```
3
```

```
1
```

```
Sorted array:
```

```
1 1 2 3
```

```
znako@znako-VirtualBox:~/UtoW/OS_LABS/build/lab5$ ./prog1L2
```

```
1: calculate e;
```

```
2: Bubble sort of array.
```

```
Make a choice: 1
```

```
8
```


Enter x to calculate e = 5

e = 2.716667

2

Size of array = 4

Filling the array:

1

2

3

1

Sorted array:

1 1 2 3

znako@znako-VirtualBox:~/UtoW/OS_LABS/build/lab5\$./prog2

0: Switch lib;

1: calculate e;

2: Bubble sort of array.

1

Enter x to calculate e = 5

e = 2.488320

2

Size of array = 4

Filling the array:

1

2

3

1

Sorted array:

1 1 2 3

0

Lib swithced. Used lib 2.

1

Enter x to calculate e = 5

e = 2.716667

2

9

```
Size of array = 4
```

```
Filling the array:
```

```
1
```

```
2
```

```
3
```

```
1
```

```
Sorted array:
```

```
1 1 2 3
```

```
znako@znako-VirtualBox:~/UtoW/OS_LABS/build/lab5$
```

Выводы

Составлены и отлажены программы на языке Си, осуществляющие работу с динамическими библиотеками. Одна подключает библиотеки на этапе линковки, другая во время работы по средствам ОС.