**Отчёт по лабораторной работе №5**

**по дисциплине “Операционные системы”**

Работу выполнил(а) студент(ка) группы М8О-206Б-20

Фурлетова Дарья Александровна, № по списку: 24

Контакты: e-mail: dafurletova@mail.ru

Работа сдана: 11 декабря 2021 г.

Преподаватель: Соколов А.А.

Итоговая оценка: \_\_\_\_\_.

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_.

1. **Тема.**

Динамические библиотеки.

1. **Цель работы.**

Целью является приобретение практических навыков в:

* создании динамических библиотек;
* создании программ, которые используют функции динамических библиотек.

1. **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 используют 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.

Пользовательский ввод для обоих программ должен быть организован следующим образом:

1. Если пользователь вводит команду «0», то программа переключает одну реализацию контрактов на другую (необходимо только для программы №2). Можно реализовать лабораторную работу без данной функции, но максимальная оценка в этом случае будет «хорошо»;
2. «1 arg1 arg2 … argN», где после «1» идут аргументы для первой функции, предусмотренной контрактами. После ввода команды происходит вызов первой функции, и на экране появляется результат её выполнения;
3. «2 arg1 arg2 … argM», где после «2» идут аргументы для второй функции, предусмотренной контрактами. После ввода команды происходит вызов второй функции, и на экране появляется результат её выполнения.

**Вариант № 21**

Варианты:

| Номер | Функция 1 | Функция 2 |
| --- | --- | --- |
| 21 | 3 | 9 |

Контракты и реализации функций:

| № | Описание | Сигнатура | Реализация 1 | Реализация 2 |
| --- | --- | --- | --- | --- |
| 3 | Подсчёт количества простых чисел на отрезке [A, B] (A, B - натуральные) | Int PrimeCount (int A, int B) | Наивный алгоритм. Проверить делимость текущего числа на все предыдущие числа. | Решето Эратосфена. |
| 9 | Отсортировать целочисленный массив | Int \* Sort(int \* array) | Пузырьковая сортировка | Сортировка Хоара |

1. **Описание программы.**

В библиотеке lib1.c написаны первые реализации функций: подсчет количества простых чисел при помощи наивного алгоритма (prime\_count) и пузырьковая сортировка целочисленного массива (sort).

Во второй библиотеке (lib2.c) созданы вторые реализации функций: подсчет количества простых чисел при помощи решета Эратосфена (prime\_count) и сортировка целочисленного массива сортировкой Хоара (sort).

В первой программе я выбрала первые реализации обеих функций из задания и подключила библиотеку, в которой они находятся на этапе линковки, т.е. подключив её строкой: #include “my\_lib.h”.

Во второй программе я подключала функции из динамических библиотек. Эти библиотеки в процессе линковки не добавляются в общий код выполнимого файла, в нём указывается только путь, по которому надо найти нужную функцию в динамической библиотеке. Кроме этого, в этой программе возможно переключение между разными реализациями одной функции. Для этого при помощи dlopen сначала загружаются динамические библиотеки и получаются указатели на них. Затем при помощи функции dlsym в библиотеке ищется заданная функция и получается указатель на неё (который и определяет, какая функция будет выполняться). При смене реализации (т.е. переключении библиотеки) dlsym ищет ту же самую функцию в другой библиотеке и выполняет уже вторую реализацию. В конце при помощи dlclose убираются указатели на библиотеки.

1. **Тестирование.**

**Данные, введенные пользователем**

Первый способ (main\_1.c)

prime\_count - 1 (int)arg1 (int)arg2

sort - 2 (int)arg1 (int)arg2 ...

Enter number the command and args: **1**

**2**

**10**

Counting prime numbers using the first implementation.

The count of prime numbers is 4

Enter number the command and args: **2**

**4**

**3 -2 0 1**

Sorting integers using the first implementation.

Sorted sequence: -2 0 1 3

Enter number the command and args:

Второй способ (main\_2.c)

Change lib - 0

prime\_count - 1 (int)arg1 (int)arg2

sort - 2 (int)arg1 (int)arg2 ...

Enter number the command and args: **1**

**23**

**713**

Counting prime numbers using the first implementation.

The count of prime numbers is 119

Enter number the command and args: **2**

**5**

**10 4 -3 -6 0**

Sorting integers using the first implementation.

Sorted sequence: -6 -3 0 4 10

Enter number the command and args: **0**

Enter number the command and args: **1**

**157**

**203**

Counting prime numbers using the second implementation.

The count of prime numbers is 10

Enter number the command and args: **2**

**3**

**13 9 13**

Sorting integers using the second implementation.

Sorted sequence: 9 13 13

Enter number the command and args:

1. **Листинг программы.**

Продемонстрирован во время сдачи.

1. **Вывод.**

В ходе выполнения данной лабораторной работы я узнала различные способы подключения библиотек в создаваемые программы. Интересным оказалось подключение динамических библиотек, написанных самостоятельно: то, что эти библиотеки не добавляются в исполняемый файл меня приятно удивило. Кроме того, я узнала, какие ключи компиляции необходимы при таком подключении библиотек.