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: Царев Михаил Вадимович

Группа: М8О-208Б-21

Вариант: 19

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/de1phin/OS/lab5

**Постановка задачи**

**Цель работы**

Целью является приобретение практических навыков в:

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

**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 19:

функции:

3) Подсчет количества простых чисел на отрезке [A; B]

7) Подсчет площади плоской геометрической фигуры по двум сторонам.

**Общие сведения о программе**

Программа состоит из 6 файлов: area.c и area.h, primes.c и primes.h, 1.c и 2.c. Модули primes и area определяют необходимые функции, а 1.c и 2.c разные варианты взаимодействия с ними

**Общий метод и алгоритм решения**

1.c динамически компилируется вместе с необходимыми библиотеками:

gcc -c ./primes/primes.c

gcc -c ./area/area.c

gcc -c ./1.c

gcc -g -o ./1 ./1.o ./primes.o ./area.o

2.c подгружает библиотеки во время выполнения, с помощью системных вызовов dlopen, dlsym:

gcc -shared -fPIC -o ./primes.so ./primes/primes.c

gcc -shared -fPIC -o ./area.so ./area/area.c

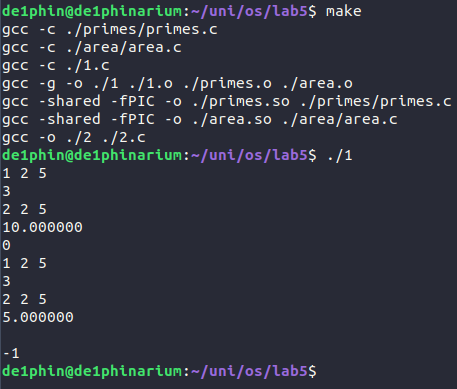
gcc -o ./2 ./2.c

**Исходный код**

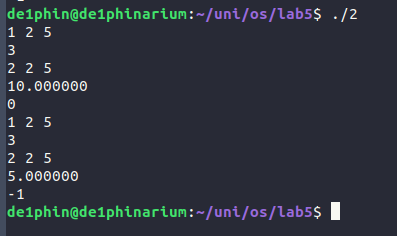
В репозитории.

**Демонстрация работы программы**

1:

****

2:



**Выводы**

Реализовал программу, динамически подгружающую библиотеку, и меняющую выполняемый код во время работы программы.