Московский Авиационный Институт

(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа №5 по курсу**

**«Операционные системы»**

**Тема работы**

**“Динамические библиотеки”**

Студент: Ханнанов Руслан Маратович

Группа: М8О-208Б-20

Вариант: 5

Преподаватель: Миронов Евгений Сергеевич

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_

Москва, 2021

**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/Naksen/OS>

**Постановка задачи**

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:

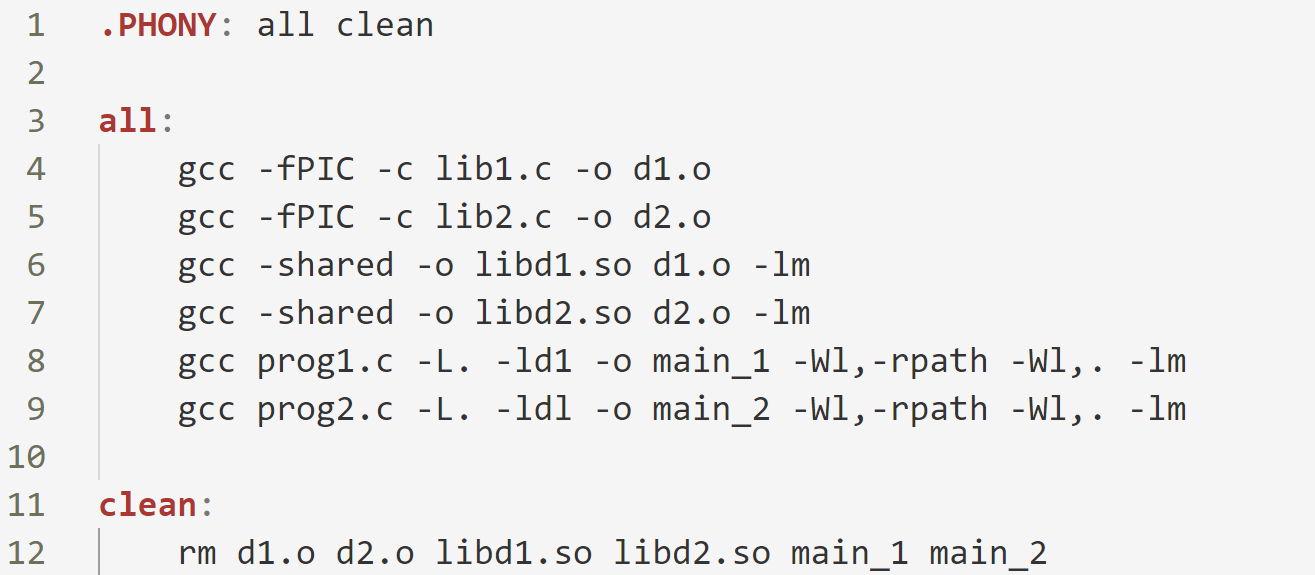
1. Если пользователь вводит команду «0», то программа переключает одну реализацию контрактов на другую (необходимо только для программы №2).

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.

**Общие сведения о программе**

Программа написана на языке Си в UNIX-подобной операционной системе.

Сборка проекта происходит при помощи make-файла:  


Флаг –fPIC означает, что используется относительная переадресация в переходах подпрограмм.

Флаг –shared предписывается, что создается динамическая библиотека.

Флаг –L. - ищет в каталоге файлы библиотеки.

Флаг –ldl используется для работ с функциями dlopen, dlsym, dlclose.

Флаг –lm/-ld1 связывает с файлом библиотеку math/lib1.

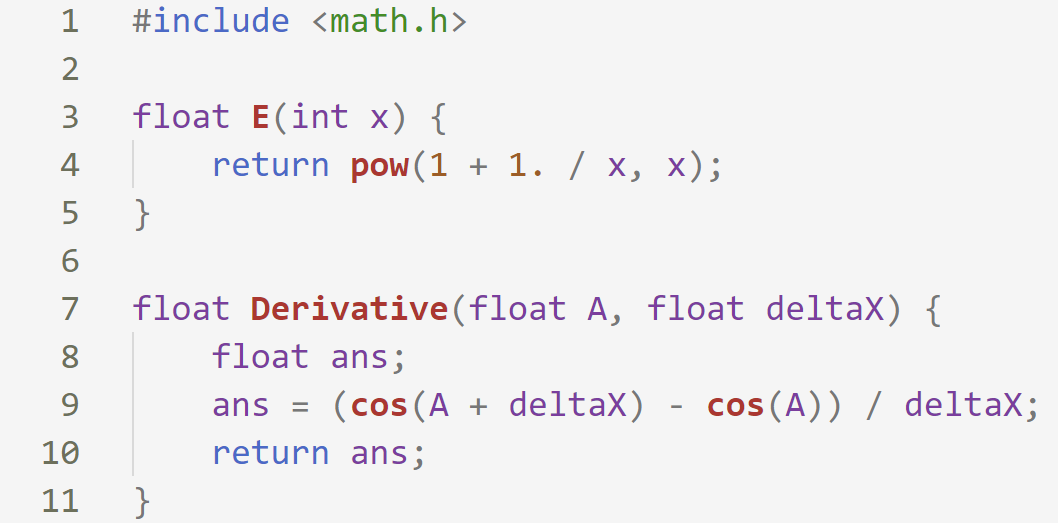
**Общий метод и алгоритм решения**

В файлах lib1.c и lib2.c реализовал функции методами, соответствующие условию задания. В тестовой программе №1, которая использует данные библиотеки во время компиляции, прописал цикл while, в условии которого идет чтение ключа. Пока ключ может считаться, в зависимости от его значения я выбираю функцию, которая будет обрабатываться и выводить результат в консоль (1 –Derivative, 2- E). После выбора функции считываю входные данные для функции, вычисляю её и вывожу результат.

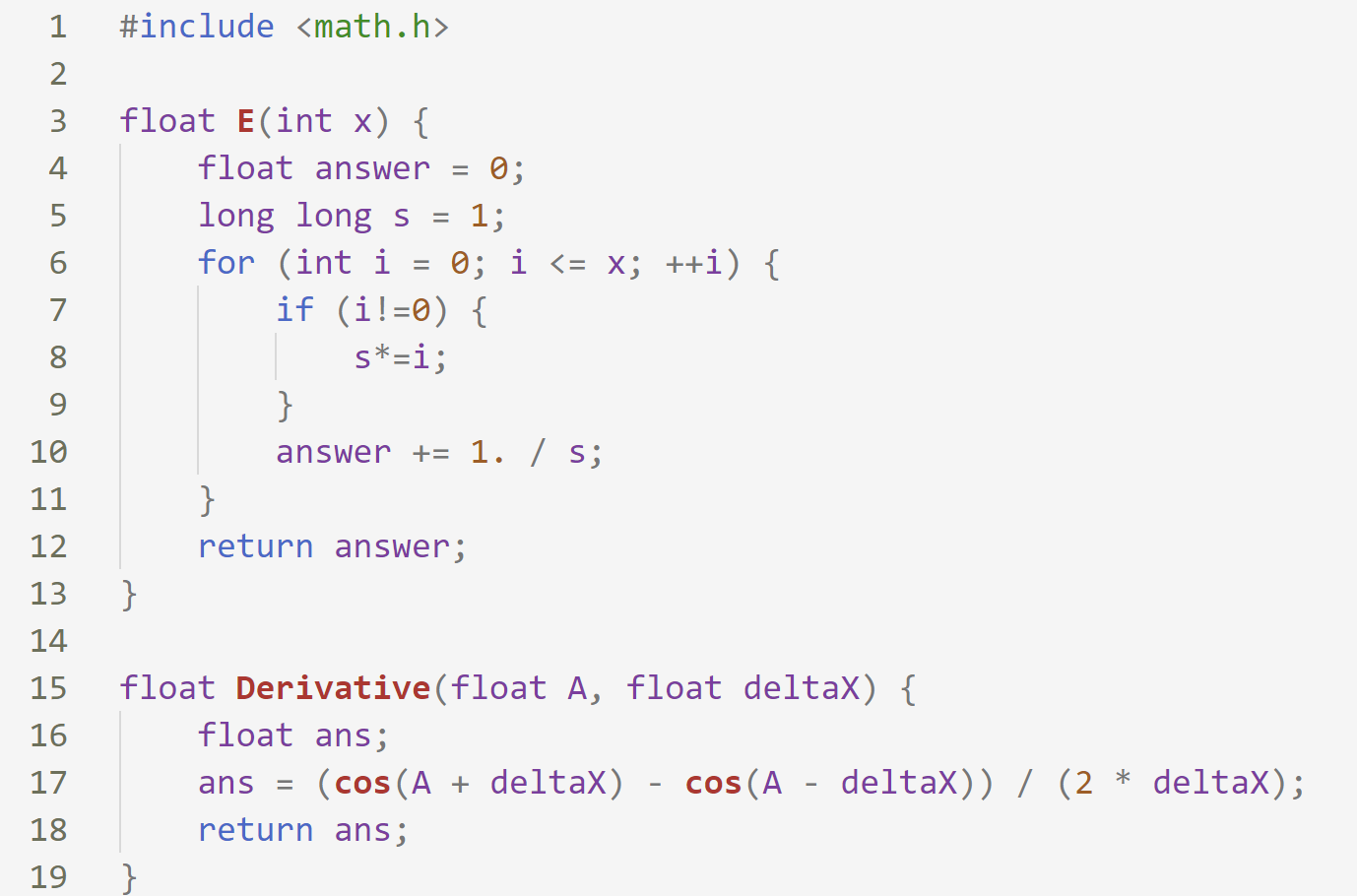
В тестовой программе №2 изначально прошу пользователя выбрать стартовую библиотеку. В зависимости от его выбора, загружаю открываю ту или иную библиотеку с помощью функции dlopen. Нахожу начало адреса функций Derivative и E с помощью функции dlsym. В цикле while считываю ключ и в зависимости от его значения обрабатываю ту или иную функцию и вывожу результат на консоль. Если ключ равен 0, с помощью функции dclose закрываю библиотеку и открываю, и загружаю (dlopen) другую библиотеку. Также с помощью dlsym нахожу начало адреса функций Derivative и E уже другой библиотеки. После окончания ввода, закрываю библиотеку.

**Исходный код**

**lib1.c**



**lib2.c**



**prog1.c**



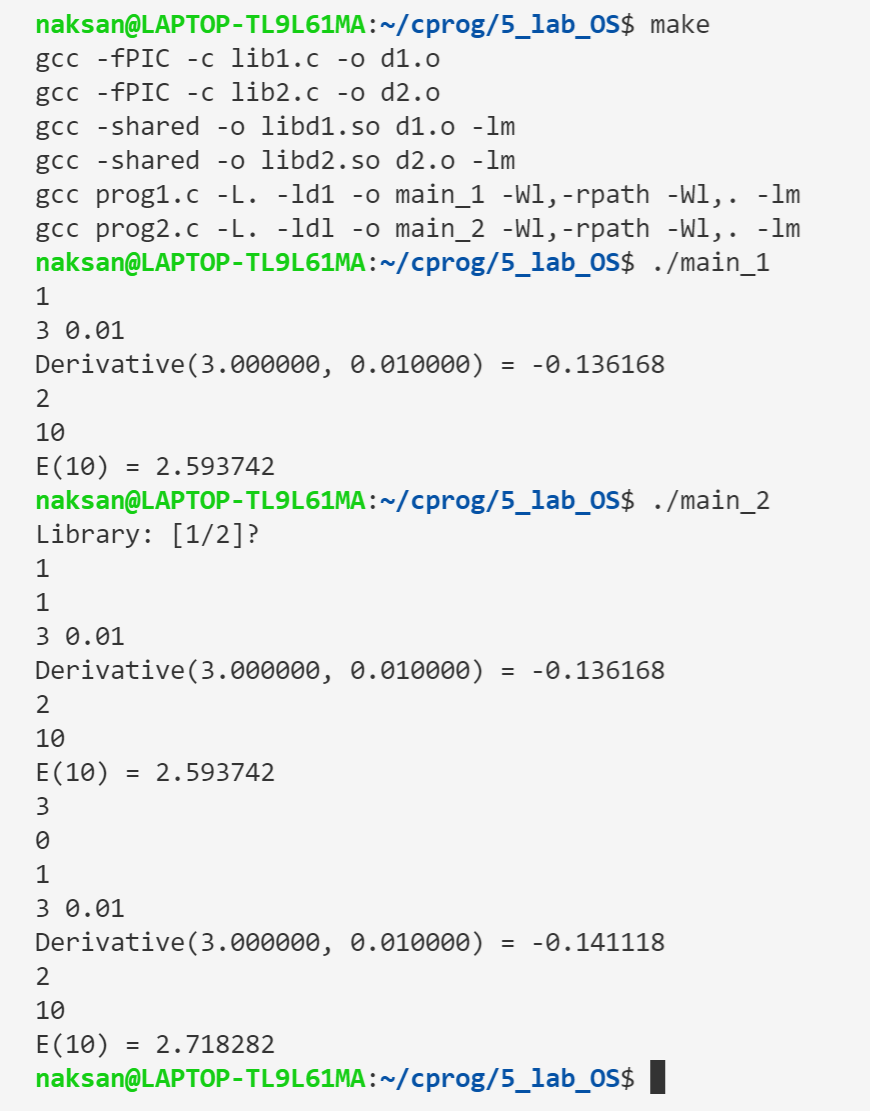
**prog2.c**







**Демонстрация работы программы**



**Выводы**

Эта лабораторная работа ознакомила и научила меня работать с динамическими библиотеками. Я описал динамические библиотеки функций подсчет производной функции cos(x) и вычисления числа e. Также я поработал с теорией, понял основной принцип динамических библиотек, их различие от статических библиотек.