Московский Авиационный Институт

(Национальный Исследовательский Университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа №5 по курсу**

**«Операционные системы»**

Группа: М80-207Б-20

Студент: Михеева К.О.

Преподаватель: Миронов Е.С.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата: 19.12.21

Москва, 2021.

# Содержание

1. Постановка задачи.
2. Общие сведения о программе.
3. Общий метод и алгоритм решения.
4. Код программы.
5. Демонстрация работы программы.
6. Вывод.

## 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.

**Вариант 1.**

1 Рассчет интеграла функции sin(x) на отрезке [A, B] с шагом e

Float SinIntegral(float A, float B, float e)

Подсчет интеграла методом прямоугольников. Подсчет интеграла методом трапеций.

2 Рассчет производной функции cos(x) в точке A с приращением deltaX

Float Derivative(float A, float deltaX)

f'(x) = (f(A + deltaX) – f(A))/deltaX f'(x) = (f(A + deltaX) – f(A-deltaX))/(2\*deltaX)

## Общие сведения о программе

Программа написана на языке С++ и состоит из 4х файлов: lib1.cpp, lib2.cpp, prog1.cpp и prog2.cpp.

В файлах lib1.cpp, lib2.cpp записаны методы решение функций, а в prog1.cpp и prog2.cpp. реализация библиотек

## Общий метод и алгоритм решения

В файлах lib1.cрр и lib2.cрр реализовала функции методами, соответствующие условию задания. В тестовой программе №1, которая использует данные библиотеки во время компиляции, прописала 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 – SinIntegral, 2-Derivative). 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 SinIntegral и Derivative 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 SinIntegral и Derivative уже другой библиотеки. После окончания ввода, закрываю библиотеку.

## Код программы

**lib1.cрр**

#include <cmath>

extern "C" float SinIntegral(float A, float B, float e); //чтобы были видны глобальные функции()

extern "C" float Derivative(float A, float deltaX);

float SinIntegral(float A, float B, float e){

float Integral = 0;

for(float i = A; i<B; i+=e){

Integral += sinf(i) \*e;

}

return Integral;

}

float Derivative(float A, float deltaX){

return (cosf (A + deltaX) - cosf(A))/deltaX;

}

**lib2.cрр**

#include <cmath>

extern "C" float SinIntegral(float A, float B, float e);

extern "C" float Derivative(float A, float deltaX);

float SinIntegral(float A, float B, float e){

float Integral = 0;

for (float i = A + e; i<B; i+=e){

Integral += (sinf(i)+sinf(i-e))/ 2\*e;

}

return Integral;

}

float Derivative(float A, float deltaX) {

return (cosf(A + deltaX) - cosf(A - deltaX)) / (2 \* deltaX);

}

**prog1.cpp**

#include <iostream>

using namespace std;

extern "C" float SinIntegral(float A, float B, float e);

extern "C" float Derivative(float A, float deltaX);

int main()

{

int flag;

cout << "Please enter the flag:\n";

while (scanf("%d",&flag) > 0 ){

if (flag == 1){

cout<<"Please enter your dates:\n";

float A, B, e;

cin>>A>>B>>e;

cout<<"SinIntegral("<< A<<","<<B<<","<< e<<")="<<SinIntegral(A, B, e)<< endl;

}

else if (flag == 2){

cout<<"Please enter your dates:\n";

float A, deltaX;

cin>>A>>deltaX;

cout<< "Derivative("<<A<<","<<deltaX<<")="<<Derivative(A,deltaX)<<endl;

}

}

return 0;

}

**prog2.cpp**

#include <stdio.h>

#include <iostream>

#include <stdlib.h>

#include <dlfcn.h>

using namespace std;

int main()

{

void\* adress = NULL;//адресс для доступа к библиотеке

//указатели на функции 1 и 2

float (\*SinIntegral)(float A, float B, float e); //объявляем указатели на библиотеки

float (\*Derivative)(float A, float deltaX);

const char\* libeary\_mas[]={"libd1.so", "libd2.so"}; //массив из путей к библиотекам

int curlib;

int statl;

cout<<"Войдите в стартовую библиотеку:"<<endl; //вводим стартовую библиотеку

cout << '\t' << "1 для первой библиотеки" <<endl;

cout << '\t' << "2 для второй библиотеки" <<endl;

cin>>statl;

bool flag = 1; // для считывание текущей библиотеки, чтобы нормально сделать

while(flag){

if (statl==1){

curlib = 0; //индекс массива

flag = 0;

}

else if(statl==2){

curlib=1;

flag=0;

}

else{

cout<<"Вы ввели неправильный номер. Войти снова!"<<endl;

cin>>statl;

}

}

adress = dlopen(libeary\_mas[curlib],RTLD\_LAZY);//RTLD\_LAZY выполняется поиск только тех символов, на которые есть ссылки из кода

if (!adress){

cout<<"Ошибка";

exit(EXIT\_FAILURE);

}

//возвращаем адрес функции из памяти библиотеки

SinIntegral = (float(\*)(float, float, float))dlsym(adress,"SinIntegral");//dlsym присваивает указателю на функцию, объявленному в начале, ее адрес в библиотеке

Derivative = (float(\*)(float, float))dlsym(adress, "Derivative");

int command;

cout<<"Пожалуйста, прочтите следующие правила, прежде чем вводить команду";

cout << '\t' << "0 за изменение контракта;" << std:: endl;

cout << '\t' << "1 для расчета the SinIntegral; " << std:: endl;

cout << '\t' << "2 для расчета the Derivative; " << std:: endl;

while(printf("Пожалуйста, введите вашу команду: ") && (scanf("%d", &command)) != EOF){

if (command==0){ //смена библиотек

dlclose(adress);//освобождает указатель на библиотеку и программа перестает ей пользоваться

if (curlib==0){

curlib=1;

adress=dlopen(libeary\_mas[curlib],RTLD\_LAZY);

if(!adress){

cout<<"Ошибка";

exit(EXIT\_FAILURE);

}

//возвращаем адрес функции из памяти библиотеки

SinIntegral = (float(\*)(float, float, float))dlsym(adress,"SinIntegral"); //dlsym присваивает указателю на функцию, объявленному в начале, ее адрес в библиотеке

Derivative = (float(\*)(float, float))dlsym(adress, "Derivative");

}

else if(curlib==1){

curlib=0;

adress = dlopen(libeary\_mas[curlib],RTLD\_LAZY);

if(!adress){

cout<<"Ошибка";

exit(EXIT\_FAILURE);

}

//возвращаем адрес функции из памяти библиотеки

SinIntegral = (float(\*)(float, float, float))dlsym(adress,"SinIntegral"); //dlsym присваивает указателю на функцию, объявленному в начале, ее адрес в библиотеке

Derivative = (float(\*)(float, float))dlsym(adress, "Derivative");

}

cout << "Вы изменили контракты!" << std:: endl;

}

else if (command==1){

float A, B, e;

cin>>A>>B>>e;

float sinintegral =SinIntegral(A,B,e);

if (sinintegral==-1){

cout<<"Пожалуйста, введите еще раз\n";

}

else{

cout<<"SinIntegral("<< A<<","<<B<<","<< e<<")="<<SinIntegral(A, B, e)<< endl;

}

}

else if (command==2){

float A, deltaX;

cin>>A>>deltaX;

float derivative =Derivative(A, deltaX);

if (derivative==-1){

cout<<"Пожалуйста, введите еще раз\n";

}

else{

cout<< "Derivative("<<A<<","<<deltaX<<")="<<Derivative(A,deltaX)<<endl;

}

}

else{

cout<<"Вам нужно было ввести только 0, 1 или 2!" << std:: endl;

}

}

dlclose(adress);

return 0;

}

## Демонстрация работы программы

kristina@kristina-Lenovo-IdeaPad-S145-15API:~/Рабочий стол/OC/lab5$ ./m1

Please enter the flag:

1

Please enter your dates:

10 5 4

SinIntegral(10,5,4)=0

0

2

Please enter your dates:

10 5 4

Derivative(10,5)=0.0158767

kristina@kristina-Lenovo-IdeaPad-S145-15API:~/Рабочий стол/OC/lab5$ ./m2

Enter the start library:

1 for first library

2 for second library

1

Please read the nexst ruls? before you enter the command 0 for changing the contract;

1 for calculating the SinIntegral;

2 for calculating the Derivative;

Please enter your command: 1

51 10 4

SinIntegral(51,10,4)=0

Please enter your command: 2

51 10 4

Derivative(51,10)=-0.100026

Please enter your command: You had to enter only 0, 1 or 2!

Please enter your command: 0

You have changed contracts!

ёёPlease enter your command: 0

## Вывод

В данной лабораторной работы я выполнила задание с динамическими библиотеками - функций подсчета для первой вычисление интеграла методом прямоугольника и трапеции. А также вычисление производной по заданным формулам в условии. Я познакомилась с динамическими и статическими библиотеками, научилась ими пользоваться в своих программах, ведь это ускоряет компиляцию программ, а динамические библиотеки экономят память: не нужно копировать файлы библиотеки, она подсоединяется в моменте работы программы. В процессе изучении и написание когда для меня стало очень неожиданным то, что использование записи **extern "C"** позволяет работать с динамической библиотекой не только с++/с (при обратной совместимости кода библиотеки с тем, что в программе) программам, но и коду на других языках. Это является крайне полезным для меня, ведь я смогу использовать в своих проектах на языке python вставки кода на с++, для ускорения работы своей программы.