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: Шапошник Никита Сергеевич

Группа: М8О-208Б-21

Вариант: 6

Преподаватель: Миронов Евгений Сергеевич

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Дата: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_

Москва, 2023

**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/Nikitashap/OS/tree/main/lab5**](https://github.com/Nikitashap/OS/tree/main/lab5)

**Постановка задачи**

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 7:** Рассчет интеграла функции sin(x) на отрезке [A, B] с шагом e.

Подсчет площади плоской геометрической фигуры по двум сторонам.

**Общие сведения о программе:** Программа состоит из четырёх файлов: r1.cpp и r2.cpp — содержат в себе реализацию функций двумя различными способами, p1.cpp — использует библиотеки на этапе компиляции, p2.cpp — загружает библиотеки, используя их местоположение и контракты.

**Общий метод и алгоритм решения**

В библиотеке r1.cpp реализованы рассчет интеграла функции sin(x) одним способом и подсчет площади прямоугольника по двум сторонам, в библиотеке r2.cpp реализованы рассчет интеграла функции sin(x) другим способом и подсчет площади треугольника по двум сторонам. Создание динамических библиотек осуществляется с помощью команды g++ -fPIC -c r1.cpp -o d1.o && g++ -shared d1.o -o libd1.so для r1.cpp и g++ -fPIC -c r2.cpp -o d2.o && g++ -shared d2.o -o libd2.so для r2.cpp. В p1.cpp выбор подключаемой библиотеки осуществляется на этапе компиляции программы, g++ p1.cpp -L. -ld1 -o main1.out -Wl,-rpath -Wl,. подключает r1.cpp, g++ p1.cpp -L. -ld2 -o main2.out -Wl,-rpath -Wl,. подключает r2.cpp. В p2.cpp возможен выбор необходимой библиотеки непосредственно при выполнении программы, скомпилированной при помощи g++ p2.cpp -ldl -o main.out .

**Исходный код**

r1.cpp

#include <cmath>

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

extern "C" float Square(float A, float B);

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

float Integral = 0;

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

Integral+=sin(x-e/2)\*e;

}

return Integral;

}

float Square(float A, float B){

return A \* B;

}

r2.cpp

#include <cmath>

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

extern "C" float Square(float A, float B);

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

float step\_number = (B-A)/e;

float Integral = 0;

for (float i=0; i<step\_number; i++){

Integral+=sin(A+i\*e);

}

Integral+=(sin(A)+sin(B))/2;

Integral\*=e;

return Integral;

}

float Square(float A, float B){

return A\*B/2;

}

p1.cpp

#include <iostream>

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

extern "C" float Square(float A, float B);

using namespace std;

int main(){

int command;

cout << "Enter 1 for Integral calculation, 2 for square calculation, 3 for exit" << '\n';

cin >> command;

while (1){

if (command==1){

float A,B,e;

cout << "Input A,B and step e" << '\n';

cin >> A >> B >> e;

cout << "Integral sin from " << A <<" to " << B << " is " << SinIntegral(A,B,e) << '\n';

cout << "Enter 1 for Integral calculation, 2 for square calculation, 3 for exit" << '\n';

}

if (command==2){

float A,B;

cout << "Input A,B" << '\n';

cin >> A >> B;

cout << "Square of rectangle with sides " << A << " and " << B << " is " << Square(A,B) << '\n';

cout << "Enter 1 for Integral calculation, 2 for square calculation, 3 for exit" << '\n';

}

if (command == 3)

{

break;

}

cin >> command;

}

return 0;

}

**p2.cpp**

#include <iostream>

#include <dlfcn.h>

#include <vector>

using namespace std;

int main()

{

void\* r = NULL;

float (\*SinIntegral)(float A, float B, float e);

float (\*Square)(float A, float B);

int lib;

cout << "0 for change libs, 1 for r1.cpp, 2 for r2.cpp, 3 for exit:\n";

cin >> lib;

while ((lib != 1)&&(lib != 2))

{

cout << "Input error, try again:\n";

cin >> lib;

}

if (lib == 1)

{

r = dlopen("./libd1.so", RTLD\_LAZY);

}

if (lib == 2)

{

r = dlopen("./libd2.so", RTLD\_LAZY);

}

SinIntegral = (float(\*)(float, float, float))dlsym(r, "SinIntegral");

Square = (float(\*)(float, float))dlsym(r, "Square");

unsigned command;

cout << "0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:\n";

cin >> command;

while (1)

{

if (command == 0)

{

if (lib == 1)

{

dlclose(r);

r = dlopen("./libd2.so", RTLD\_LAZY);

SinIntegral = (float(\*)(float, float, float))dlsym(r, "SinIntegral");

Square = (float(\*)(float, float))dlsym(r, "Square");

lib = 2;

cout << "r1 changed on r2\n";

cout << "0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:\n";

cin >> command;

continue;

}

else

{

dlclose(r);

r = dlopen("./libd1.so", RTLD\_LAZY);

SinIntegral = (float(\*)(float, float, float))dlsym(r, "SinIntegral");

Square = (float(\*)(float, float))dlsym(r, "Square");

lib = 1;

cout << "r2 changed on r1\n";

cout << "0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:\n";

cin >> command;

continue;

}

}

if (command == 1)

{

float a, b, e;

cout << "Input A,B and step e" << '\n';

cin >> a >> b >> e;

cout << "Integral sin from " << a <<" to " << b << " is " << SinIntegral(a,b,e) << '\n';

cout << "0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:\n";

}

if (command == 2)

{

float a,b;

cout << "Input A,B" << '\n';

cin >> a >> b;

if (lib==1){

cout << "Square of rectangle with sides " << a << " and " << b << " is " << Square(a,b) << '\n';

}

else{

cout << "Square of triangle with sides " << a << " and " << b << " is " << Square(a,b) << '\n';

}

cout << "0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:\n";

}

if (command == 3)

{

break;

}

cin >> command;

}

dlclose(r);

return 0;

}

**Демонстрация работы программы**

iMac-Apple:lab5 appleimac$ ./main1.out

Enter 1 for Integral calculation, 2 for square calculation, 3 for exit

1

Input A,B and step e

1

2

0.005

Integral sin from 1 to 2 is 0.960651

Enter 1 for Integral calculation, 2 for square calculation, 3 for exit

3

iMac-Apple:lab5 appleimac$ ./main.out

0 for change libs, 1 for r1.cpp, 2 for r2.cpp, 3 for exit:

1

0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:

1

Input A,B and step e

1

2

0.005

Integral sin from 1 to 2 is 0.960651

0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:

0

r1 changed on r2

0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:

1

Input A,B and step e

1

2

0.005

Integral sin from 1 to 2 is 0.960655

0 for change libs, 1 for SinIntegral, 2 for Square, 3 for exit:

3

**Выводы**

Проделав лабораторную работу, я приобрёл практические навыки, необходимые для работы с динамическими библиотеками.