Московский Авиационный Институт
(Национальный Исследовательский Университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Лабораторная работа №4 по курсу
«Операционные системы»

Тема работы
“Динамические библиотеки”

Студент: Тутаев Владимир Владимирович
Группа: М8О-201Б-23
Вариант: 12

Преподаватель: Миронов Евгений Сергеевич
Оценка: _____
Дата: _____
Подпись: _____

Москва, 2024

Репозиторий
https://github.com/Volan4ik/MAI_OS.git

Постановка задачи

Задача: 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.

Общие сведения о программе

Программа состоит из нескольких частей, которые работают вместе. В ней используются две разные реализации функций для вычисления наибольшего общего делителя (GCF) и площади фигуры (Square). Одна реализация использует алгоритм Евклида для GCF и обычное произведение для площади, а другая - наивный алгоритм для GCF и половину произведения для площади. Эти функции скомпилированы в две отдельные динамические библиотеки (libd1.so и libd2.so). В главной программе launch.cpp, пользователь может выбрать, какую библиотеку использовать для расчетов. Переключение между библиотеками осуществляется во время выполнения программы с помощью функций dlopen, dlsym, dlclose.
compilation.cpp является программой, к которой библиотека подгружается на этапе компиляции
launch.cpp является программой, к которой библиотека подключается непосредственно в самом коде

Исходный код в Приложении 1

Strace в Приложении 2

Выводы

Эта лабораторная работа научила меня пользоваться динамическими библиотеками. Я узнал, как их загружать во время выполнения программы, написал простые и приятные алгоритмы для нужных функций, рассмотрел изменение динамических библиотек в системных вызовах.

Приложение 1

derivative_first.hpp

```
#ifndef DERIVATIVE_FIRST_HPP
#define DERIVATIVE_FIRST_HPP

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

#endif
```

derivative_second.hpp

```
#ifndef DERIVATIVE_SECOND_HPP
#define DERIVATIVE_SECOND_HPP

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

#endif
```

e_formule.hpp

```
#ifndef E_FORMULA_HPP
#define E_FORMULA_HPP

extern "C" {
    float E(int x);
}

#endif
```

e_series.hpp

```
#ifndef E_SERIES_H
#define E_SERIES_H

extern "C" {
    float E(int x);
}

#endif
```

derivative.cpp

```
#include "../../include/Func_1/derivative_first.hpp"
#include <cmath>

float Derivative(float A, float deltaX) {
    return (cos(A + deltaX) - cos(A)) / deltaX;
}
```

derivative_second.cpp

```cpp
#include "../../include/Func_1/derivative_second.hpp"
#include <cmath>

float Derivative(float A, float deltaX) {
    return (cos(A + deltaX) - cos(A - deltaX)) / (2 * deltaX);
}
```

e_formule.cpp

```cpp
#include "../../include/Func_2/e_formule.hpp"
#include <cmath>

float E(int x) {
    return pow(1 + 1.0 / x, x);
}
```

e_series.cpp

```cpp
#include "../../include/Func_2/e_series.hpp"

float factorial(int n) {
    float result = 1;
    for (int i = 2; i <= n; ++i) {
        result *= i;
    }
    return result;
}

float E(int x) {
    float sum = 0;
    for (int n = 0; n <= x; ++n) {
        sum += 1.0 / factorial(n);
    }
    return sum;
}
```

main1.cpp

```cpp
#include <iostream>
#include "../Lib/include/Func_1/derivative_first.hpp"
#include "../Lib/include/Func_2/e_formule.hpp"

int main() {
    int command;
    while (true) {
        std::cout << "Enter command (1 for Derivative, 2 for E, 0 to exit): ";
        std::cin >> command;

        if (command == 0) {
            break;
        } else if (command == 1) {
            float A, deltaX;
            std::cout << "Enter A and deltaX: ";
```

```
            std::cin >> A >> deltaX;
            std::cout << "Derivative at " << A << " with deltaX " << deltaX << " is " << Derivative(A, deltaX) << std::endl;
        } else if (command == 2) {
            int x;
            std::cout << "Enter x: ";
            std::cin >> x;
            std::cout << "E(" << x << ") = " << E(x) << std::endl;
        } else {
            std::cout << "Unknown command" << std::endl;
        }
    }
    return 0;
}
```

main2.cpp

```
#include <iostream>
#include <dlfcn.h>

typedef float (*DerivativeFunc)(float, float);
typedef float (*EFunc)(int);

int main() {
    void* handle_derivative = dlopen("./libderivative_first.so", RTLD_LAZY);
    void* handle_e_formule = dlopen("./libe_formule.so", RTLD_LAZY);

    if (!handle_derivative || !handle_e_formule) {
        std::cerr << "Cannot open libraries: " << dlerror() << std::endl;
        return 1;
    }

    DerivativeFunc Derivative = (DerivativeFunc)dlsym(handle_derivative, "Derivative");
    EFunc E = (EFunc)dlsym(handle_e_formule, "E");

    if (!Derivative || !E) {
        std::cerr << "Cannot load symbols: " << dlerror() << std::endl;
        dlclose(handle_derivative);
        dlclose(handle_e_formule);
        return 1;
    }

    int command;
    while (true) {
        std::cout << "Enter command (1 for Derivative, 2 for E, 0 to switch, -1 to exit): ";
        std::cin >> command;

        if (command == -1) {
            break;
        } else if (command == 0) {
            // Переключение реализаций
            dlclose(handle_derivative);
            dlclose(handle_e_formule);
```

```
            handle_derivative = dlopen("./libderivative_second.so", RTLD_LAZY);
            handle_e_formule = dlopen("./libe_series.so", RTLD_LAZY);

            if (!handle_derivative || !handle_e_formule) {
                std::cerr << "Cannot open libraries: " << dlerror() << std::endl;
                return 1;
            }

            Derivative = (DerivativeFunc)dlsym(handle_derivative, "Derivative");
            E = (EFunc)dlsym(handle_e_formule, "E");

            if (!Derivative || !E) {
                std::cerr << "Cannot load symbols: " << dlerror() << std::endl;
                dlclose(handle_derivative);
                dlclose(handle_e_formule);
                return 1;
            }

            std::cout << "Switched to derivative_second and e_series libraries" << std::endl;
        } else if (command == 1) {
            float A, deltaX;
            std::cout << "Enter A and deltaX: ";
            std::cin >> A >> deltaX;
            std::cout << "Derivative at " << A << " with deltaX " << deltaX << " is " << Derivative(A, deltaX) << std::endl;
        } else if (command == 2) {
            int x;
            std::cout << "Enter x: ";
            std::cin >> x;
            std::cout << "E(" << x << ") = " << E(x) << std::endl;
        } else {
            std::cout << "Unknown command" << std::endl;
        }
    }

    dlclose(handle_derivative);
    dlclose(handle_e_formule);
    return 0;
}
```

Приложение 2

strace_lab_4_compile_time.txt:

execve("./main_compile_time", ["./main_compile_time"], 0x7ffc27bc0fb0 /* 75 vars */) = 0

brk(NULL)   = 0x625d12388000

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x735b069c1000

access("/etc/ld.so.preload", R_OK)   = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/glibc-hwcaps/x86-64-v4/libderivative_first.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/glibc-hwcaps/x86-64-v4/", 0x7fff6429ce00, 0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/glibc-hwcaps/x86-64-v3/libderivative_first.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/glibc-hwcaps/x86-64-v3/", 0x7fff6429ce00, 0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/glibc-hwcaps/x86-64-v2/libderivative_first.so", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

newfstatat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/glibc-hwcaps/x86-64-v2/", 0x7fff6429ce00, 0) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/libderivative_first.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0775, st_size=15472, ...}) = 0

mmap(NULL, 16408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x735b069bc000

mmap(0x735b069bd000, 4096, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x735b069bd000

mmap(0x735b069be000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x735b069be000

mmap(0x735b069bf000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x735b069bf000

close(3)   = 0

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/libe_formule.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0775, st_size=15528, ...}) = 0

mmap(NULL, 16408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x735b069b7000

mmap(0x735b069b8000, 4096, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x735b069b8000

mmap(0x735b069b9000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x735b069b9000

mmap(0x735b069ba000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x735b069ba000

close(3) = 0

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

fstat(3, {st_mode=S_IFREG|0644, st_size=87915, ...}) = 0

mmap(NULL, 87915, PROT_READ, MAP_PRIVATE, 3, 0) = 0x735b069a1000

close(3) = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=2592144, ...}) = 0

mmap(NULL, 2605376, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x735b06600000

mmap(0x735b0669d000, 1310720, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x9d000) = 0x735b0669d000

mmap(0x735b067dd000, 581632, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1dd000) = 0x735b067dd000

mmap(0x735b0686b000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x26b000) = 0x735b0686b000

mmap(0x735b06879000, 12608, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x735b06879000

close(3) = 0

openat(AT_FDCWD, "/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build/libc.so.6", O_RDONLY|O_CLOEXEC) = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\243\2\0\0\0\0\0"..., 832) = 832

pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

fstat(3, {st_mode=S_IFREG|0755, st_size=2125328, ...}) = 0

pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0"..., 784, 64) = 784

mmap(NULL, 2170256, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x735b06200000

mmap(0x735b06228000, 1605632, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x735b06228000

mmap(0x735b063b0000, 323584, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x1b0000) = 0x735b063b0000

mmap(0x735b063ff000, 24576, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1fe000) = 0x735b063ff000

mmap(0x735b06405000, 52624, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x735b06405000

close(3) = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=952616, ...}) = 0

mmap(NULL, 950296, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x735b068b8000

mmap(0x735b068c8000, 520192, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x10000) = 0x735b068c8000

mmap(0x735b06947000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3,
0x8f000) = 0x735b06947000

mmap(0x735b0699f000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe7000) = 0x735b0699f000

close(3) = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=182944, ...}) = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x735b068b6000

mmap(NULL, 181160, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x735b06889000

mmap(0x735b0688d000, 143360, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x4000) = 0x735b0688d000

mmap(0x735b068b0000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x27000) = 0x735b068b0000

mmap(0x735b068b4000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2b000) = 0x735b068b4000

close(3) = 0

mmap(NULL, 12288, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x735b06886000

arch_prctl(ARCH_SET_FS, 0x735b06886740) = 0

set_tid_address(0x735b06886a10) = 921534

set_robust_list(0x735b06886a20, 24) = 0

rseq(0x735b06887060, 0x20, 0, 0x53053053) = 0

mprotect(0x735b063ff000, 16384, PROT_READ) = 0

mprotect(0x735b068b4000, 4096, PROT_READ) = 0

mprotect(0x735b0699f000, 4096, PROT_READ) = 0

mprotect(0x735b0686b000, 45056, PROT_READ) = 0

mprotect(0x735b069ba000, 4096, PROT_READ) = 0

mprotect(0x735b069bf000, 4096, PROT_READ) = 0

mprotect(0x625d10a35000, 4096, PROT_READ) = 0

mprotect(0x735b069f9000, 8192, PROT_READ) = 0

prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

munmap(0x735b069a1000, 87915) = 0

futex(0x735b068797bc, FUTEX_WAKE_PRIVATE, 2147483647) = 0

getrandom("\xe5\x74\x26\xec\x86\xae\xd2\x07", 8, GRND_NONBLOCK) = 8

brk(NULL) = 0x625d12388000

brk(0x625d123a9000) = 0x625d123a9000

fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...}) = 0

write(1, "Enter command (1 for Derivative,"..., 54) = 54

fstat(0, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...}) = 0

read(0, "1\n", 1024) = 2

write(1, "Enter A and deltaX: ", 20) = 20

read(0, "1 3\n", 1024)   = 4

write(1, "Derivative at 1 with deltaX 3 is"..., 43) = 43

write(1, "Enter command (1 for Derivative,"..., 54) = 54

read(0, "2\n", 1024)   = 2

write(1, "Enter x: ", 9)   = 9

read(0, "5\n", 1024)   = 2

write(1, "E(5) = 2.48832\n", 15)   = 15

write(1, "Enter command (1 for Derivative,"..., 54) = 54

read(0, "0\n", 1024)   = 2

lseek(0, -1, SEEK_CUR)   = -1 ESPIPE (Illegal seek)

exit_group(0)   = ?

+++ exited with 0 +++

strace_lab_4_runtime.txt:

execve("./main_runtime", ["./main_runtime"], 0x7fff823de990 /* 75 vars */) = 0

brk(NULL)   = 0x58e9756d3000

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0x70bcbd303000

access("/etc/ld.so.preload", R_OK)   = -1 ENOENT (No such file or directory)

openat(AT_FDCWD, "/etc/ld.so.cache", O_RDONLY|O_CLOEXEC) = 3

fstat(3, {st_mode=S_IFREG|0644, st_size=87915, ...}) = 0

mmap(NULL, 87915, PROT_READ, MAP_PRIVATE, 3, 0) = 0x70bcbd2ed000

close(3)   = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libstdc++.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=2592144, ...}) = 0

mmap(NULL, 2605376, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbd000000

mmap(0x70bcbd09d000, 1310720, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x9d000) = 0x70bcbd09d000

mmap(0x70bcbd1dd000, 581632, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1dd000) = 0x70bcbd1dd000

mmap(0x70bcbd26b000, 57344, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x26b000) = 0x70bcbd26b000

mmap(0x70bcbd279000, 12608, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x70bcbd279000

close(3) = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libc.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\220\243\2\0\0\0\0"..., 832) = 832

pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0"..., 784, 64) = 784

fstat(3, {st_mode=S_IFREG|0755, st_size=2125328, ...}) = 0

pread64(3, "\6\0\0\0\4\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0\0@\0\0\0\0\0\0"..., 784, 64) = 784

mmap(NULL, 2170256, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbcc00000

mmap(0x70bcbcc28000, 1605632, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x28000) = 0x70bcbcc28000

mmap(0x70bcbcdb0000, 323584, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1b0000) = 0x70bcbcdb0000

mmap(0x70bcbcdff000, 24576, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1fe000) = 0x70bcbcdff000

mmap(0x70bcbce05000, 52624, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_ANONYMOUS, -1, 0) = 0x70bcbce05000

close(3) = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libm.so.6", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\3\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=952616, ...}) = 0

mmap(NULL, 950296, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbcf17000

mmap(0x70bcbcf27000, 520192, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x10000) = 0x70bcbcf27000

mmap(0x70bcbcfa6000, 360448, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x8f000) = 0x70bcbcfa6000

mmap(0x70bcbcffe000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0xe7000) = 0x70bcbcffe000

close(3)   = 0

openat(AT_FDCWD, "/lib/x86_64-linux-gnu/libgcc_s.so.1", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0644, st_size=182944, ...}) = 0

mmap(NULL, 181160, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbd2c0000

mmap(0x70bcbd2c4000, 143360, PROT_READ|PROT_EXEC,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x4000) = 0x70bcbd2c4000

mmap(0x70bcbd2e7000, 16384, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x27000)
= 0x70bcbd2e7000

mmap(0x70bcbd2eb000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2b000) = 0x70bcbd2eb000

close(3)   = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x70bcbd2be000

arch_prctl(ARCH_SET_FS, 0x70bcbd2bf500) = 0

set_tid_address(0x70bcbd2bf7d0)   = 921575

set_robust_list(0x70bcbd2bf7e0, 24)   = 0

rseq(0x70bcbd2bfe20, 0x20, 0, 0x53053053) = 0

mprotect(0x70bcbcdff000, 16384, PROT_READ) = 0

mprotect(0x70bcbd2eb000, 4096, PROT_READ) = 0

mprotect(0x70bcbcffe000, 4096, PROT_READ) = 0

mmap(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) =
0x70bcbd2bc000

mprotect(0x70bcbd26b000, 45056, PROT_READ) = 0

mprotect(0x58e973d3f000, 4096, PROT_READ) = 0

mprotect(0x70bcbd33b000, 8192, PROT_READ) = 0

prlimit64(0, RLIMIT_STACK, NULL, {rlim_cur=8192*1024, rlim_max=RLIM64_INFINITY}) = 0

munmap(0x70bcbd2ed000, 87915)   = 0

futex(0x70bcbd2797bc, FUTEX_WAKE_PRIVATE, 2147483647) = 0

getrandom("\x13\x8b\xfd\x53\x48\xb3\x9f\x6c", 8, GRND_NONBLOCK) = 8

brk(NULL) = 0x58e9756d3000

brk(0x58e9756f4000) = 0x58e9756f4000

openat(AT_FDCWD, "./libderivative_first.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0775, st_size=15472, ...}) = 0

getcwd("/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build", 128) = 42

mmap(NULL, 16408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbd2fe000

mmap(0x70bcbd2ff000, 4096, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x70bcbd2ff000

mmap(0x70bcbd300000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd300000

mmap(0x70bcbd301000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd301000

close(3) = 0

mprotect(0x70bcbd301000, 4096, PROT_READ) = 0

openat(AT_FDCWD, "./libe_formule.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0775, st_size=15528, ...}) = 0

getcwd("/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build", 128) = 42

mmap(NULL, 16408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbd2f9000

mmap(0x70bcbd2fa000, 4096, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x70bcbd2fa000

mmap(0x70bcbd2fb000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd2fb000

mmap(0x70bcbd2fc000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd2fc000

close(3) = 0

mprotect(0x70bcbd2fc000, 4096, PROT_READ) = 0

fstat(1, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...}) = 0

write(1, "Enter command (1 for Derivative,"..., 68) = 68

fstat(0, {st_mode=S_IFCHR|0620, st_rdev=makedev(0x88, 0x1), ...}) = 0

read(0, "1\n", 1024)   = 2

write(1, "Enter A and deltaX: ", 20)   = 20

read(0, "1 3\n", 1024)   = 4

write(1, "Derivative at 1 with deltaX 3 is"..., 43) = 43

write(1, "Enter command (1 for Derivative,"..., 68) = 68

read(0, "2\n", 1024)   = 2

write(1, "Enter x: ", 9)   = 9

read(0, "5\n", 1024)   = 2

write(1, "E(5) = 2.48832\n", 15)   = 15

write(1, "Enter command (1 for Derivative,"..., 68) = 68

read(0, "0\n", 1024)   = 2

munmap(0x70bcbd2fe000, 16408)   = 0

munmap(0x70bcbd2f9000, 16408)   = 0

openat(AT_FDCWD, "./libderivative_second.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0775, st_size=15472, ...}) = 0

getcwd("/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build", 128) = 42

mmap(NULL, 16408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbd2fe000

mmap(0x70bcbd2ff000, 4096, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x70bcbd2ff000

mmap(0x70bcbd300000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd300000

mmap(0x70bcbd301000, 8192, PROT_READ|PROT_WRITE,
MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd301000

close(3)   = 0

mprotect(0x70bcbd301000, 4096, PROT_READ) = 0

openat(AT_FDCWD, "./libe_series.so", O_RDONLY|O_CLOEXEC) = 3

read(3, "\177ELF\2\1\1\0\0\0\0\0\0\0\0\0\3\0>\0\1\0\0\0\0\0\0\0\0\0\0"..., 832) = 832

fstat(3, {st_mode=S_IFREG|0775, st_size=15368, ...}) = 0

getcwd("/home/cbf/MAI/TRASH/MAI_OS/lab_4_os/build", 128) = 42

mmap(NULL, 16408, PROT_READ, MAP_PRIVATE|MAP_DENYWRITE, 3, 0) = 0x70bcbd2f9000

mmap(0x70bcbd2fa000, 4096, PROT_READ|PROT_EXEC, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x1000) = 0x70bcbd2fa000

mmap(0x70bcbd2fb000, 4096, PROT_READ, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd2fb000

mmap(0x70bcbd2fc000, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_FIXED|MAP_DENYWRITE, 3, 0x2000) = 0x70bcbd2fc000

close(3) = 0

mprotect(0x70bcbd2fc000, 4096, PROT_READ) = 0

write(1, "Switched to derivative_second an"..., 53) = 53

write(1, "Enter command (1 for Derivative,"..., 68) = 68

read(0, "1\n", 1024) = 2

write(1, "Enter A and deltaX: ", 20) = 20

read(0, "1 3\n", 1024) = 4

write(1, "Derivative at 1 with deltaX 3 is"..., 44) = 44

write(1, "Enter command (1 for Derivative,"..., 68) = 68

read(0, "2\n", 1024) = 2

write(1, "Enter x: ", 9) = 9

read(0, "5\n", 1024) = 2

write(1, "E(5) = 2.71667\n", 15) = 15

write(1, "Enter command (1 for Derivative,"..., 68) = 68

read(0, "-1\n", 1024) = 3

munmap(0x70bcbd2fe000, 16408) = 0

munmap(0x70bcbd2f9000, 16408) = 0

lseek(0, -1, SEEK_CUR) = -1 ESPIPE (Illegal seek)

exit_group(0) = ?

+++ exited with 0 +++