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: Смирнов А.В.

Группа: М8О-207Б-21

Вариант: 23

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/Liguha/OS>

**Постановка задачи**

**Цель работы**

Целью является приобретение практических навыков в:

* Создание динамических библиотек
* Создание программ, которые используют функции динамических библиотек
* Работа со сборочной системой

**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.

**Общие сведения о программе**

Динамические библиотеки с функциями компилируются из файлов functions1.c и functions2.c соответственно. Основные программы компилируются из файлов program1.c и program2.c с использование заголовочного файла functions.h (описание функций из динамических библиотек).

Во второй программе используются библиотечные вызовы dlopen, dlsym, dlclose – открытие динамической библиотеки, получение функции и закрытие соответственно.

**Общий метод и алгоритм решения**

Во второй программе при получении 0 на вход происходит следующее: закрывается текущая динамическая библиотека, открывается новая (имена библиотек берутся из массива размера N по формуле (n + 1) % N, где n – текущая библиотека) и из неё загружаются функции вместо функций старой.

**Исходный код**

|  |
| --- |
| **functions1.c** |
| int GCD(int a, int b)  {  while (a != 0)  {  b = b % a;  int tmp = a;  a = b;  b = tmp;  }  return b;  }  float E(int x)  {  float a = 1 + 1.0 / x;  float res = 1.0;  while (x != 0)  {  if (x % 2 == 1)  res \*= a;  a = a \* a;  x = x / 2;  }  return res;  } |

|  |
| --- |
| **functions2.c** |
| int GCD(int a, int b)  {  int res = 1;  for (int i = 2; i <= a && i <= b; i++)  {  if (a % i == 0 && b % i == 0)  res = i;  }  return res;  }  float E(int x)  {  float res = 1.0;  long long n = 1;  for (int i = 1; i <= x; i++)  {  n = n \* i;  res += 1.0 / n;  }  return res;  } |

|  |
| --- |
| **program1.c** |
| #include "stdio.h"  #include "functions.h"  #ifdef SYSTEM  #define PRINT\_OS printf("Operation system: %s\n", SYSTEM)  #else  #define PRINT\_OS  #endif  int main()  {  while (1)  {  int t;  scanf("%d", &t);  if (t == 1)  {  int a, b;  scanf("%d %d", &a, &b);  PRINT\_OS;  printf("GCD: %d\n", GCD(a, b));  }  if (t == 2)  {  int x;  scanf("%d", &x);  PRINT\_OS;  printf("E: %.10f\n", E(x));  }  if (t == -1)  break;  }  } |

|  |
| --- |
| **program2.c** |
| #include <stdio.h>  #include <dlfcn.h>  #ifdef SYSTEM  #define PRINT\_OS printf("Operation system: %s\n", SYSTEM)  #else  #define PRINT\_OS  #endif  #define CHECK\_ERROR(expr, message) \  do \  { \  void\* res = (expr); \  if (res == NULL) \  { \  perror(message); \  return -1; \  } \  } while (0)  const int N = 2;  const char\* names[] = {"./libdynamic1.so", "./libdynamic2.so"};  int main()  {  int n = 0;  void\* handle;  float(\*E)(int); int(\*GCD)(int, int);  CHECK\_ERROR(handle = dlopen(names[n], RTLD\_LAZY), "dlopen error");  CHECK\_ERROR(E = dlsym(handle, "E"), "dlsym error (E)");  CHECK\_ERROR(GCD = dlsym(handle, "GCD"), "dlsym error (GCD)");  while(1)  {  int t;  scanf("%d", &t);  if (t == 0)  {  n = (n + 1) % N;  if (dlclose(handle) != 0)  {  perror("dlclose error");  return -1;  };  CHECK\_ERROR(handle = dlopen(names[n], RTLD\_LAZY), "dlopen error");  CHECK\_ERROR(E = dlsym(handle, "E"), "dlsym error (E)");  CHECK\_ERROR(GCD = dlsym(handle, "GCD"), "dlsym error (GCD)");  }  if (t == 1)  {  int a, b;  scanf("%d %d", &a, &b);  PRINT\_OS;  printf("GCD: %d\n", (\*GCD)(a, b));  }  if (t == 2)  {  int x;  scanf("%d", &x);  PRINT\_OS;  printf("E: %.10f\n", (\*E)(x));  }  if (t == -1)  break;  }  } |

**Демонстрация работы программы**

liguha@Laptop:~/OS/LR5/build$ cmake -D LIB\_PRINT\_TYPE=ABCD ..

CMake Error at CMakeLists.txt:13 (message):

Unexpected LIB\_PRINT\_TYPE: ABCD

-- Configuring incomplete, errors occurred!

See also "/home/liguha/OS/LR5/build/CMakeFiles/CMakeOutput.log".

liguha@Laptop:~/OS/LR5/build$ cmake -U LIB\_PRINT\_TYPE ..

-- Configuring done

-- Generating done

-- Build files have been written to: /home/liguha/OS/LR5/build

liguha@Laptop:~/OS/LR5/build$ cmake --build .

[ 20%] Built target dynamic1

[ 40%] Built target dynamic2

[ 60%] Built target program11.out

[ 80%] Built target program12.out

[100%] Built target program2.out

liguha@Laptop:~/OS/LR5/build$ ./program11.out

1 198 26

GCD: 2

2 15

E: 2.6328814030

-1

liguha@Laptop:~/OS/LR5/build$ ./program12.out

1 198 26

GCD: 2

2 15

E: 2.7182819843

-1

liguha@Laptop:~/OS/LR5/build$ ./program2.out

1 198 26

GCD: 2

2 15

E: 2.6328814030

0

1 198 26

GCD: 2

2 15

E: 2.7182819843

-1

liguha@Laptop:~/OS/LR5/build$ cmake -D LIB\_PRINT\_TYPE=EXTENDED ..

-- Configuring done

-- Generating done

-- Build files have been written to: /home/liguha/OS/LR5/build

liguha@Laptop:~/OS/LR5/build$ cmake --build .

[ 10%] Building C object CMakeFiles/dynamic1.dir/src/functions1.c.o

[ 20%] Linking C shared library libdynamic1.so

[ 20%] Built target dynamic1

[ 30%] Building C object CMakeFiles/dynamic2.dir/src/functions2.c.o

[ 40%] Linking C shared library libdynamic2.so

[ 40%] Built target dynamic2

[ 50%] Building C object CMakeFiles/program11.out.dir/src/program1.c.o

[ 60%] Linking C executable program11.out

[ 60%] Built target program11.out

[ 70%] Building C object CMakeFiles/program12.out.dir/src/program1.c.o

[ 80%] Linking C executable program12.out

[ 80%] Built target program12.out

[ 90%] Building C object CMakeFiles/program2.out.dir/src/program2.c.o

[100%] Linking C executable program2.out

[100%] Built target program2.out

liguha@Laptop:~/OS/LR5/build$ ./program11.out

2 15

Operation system: Linux-5.15.0-56-generic

E: 2.6328814030

-1

**Выводы**

Составлены и отлажены программы на языке Си, осуществляющие работу с динамическими библиотеками. Одна подключает библиотеки на этапе линковки, другая во время работы по средствам ОС.