Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт №8 “Компьютерные науки и прикладная математика”

Кафедра №806 “Вычислительная математика и программирование”

**Лабораторная работа №4 по курсу**

**«Операционные системы»**

Группа: М80-206Б-20

Студент: Филатов А.К.

Преподаватель: Миронов Е.С.

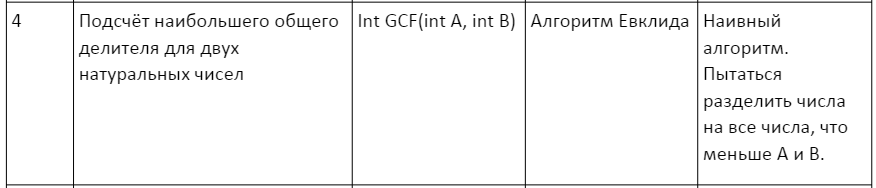
Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

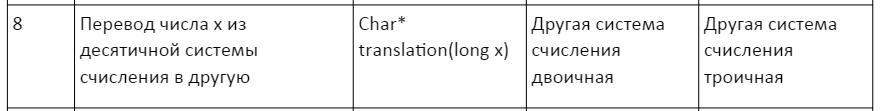
Дата: 16.12.2023

Москва, 2023

**Постановка задачи**

Необходимо реализовать две динамические библиотеки:





**Общий метод и алгоритм решения**

Использованные системные вызовы:

* void \*dlopen(const char \*filename, int flag) -загружает динамическую библиотеку, имя которой указано в строке filename, и возвращает прямой указатель на начало динамической библиотеки.
* void \*dlsym(void \*handle, char \*symbol) - использует указатель на динамическую библиотеку, возвращаемую dlopen, и оканчивающееся нулем символьное имя, а затем возвращает адрес, указывающий, откуда загружается этот символ.
* int dlclose(void \*handle) - уменьшает на единицу счетчик ссылок на указатель динамической библиотеки handle. Если нет других загруженных библиотек, использующих ее символы и если счетчик ссылок принимает нулевое значение, то динамическая библиотека выгружается.

В начале создадим две библиотеки, каждая из которых содержит две функции: нахождение НОД двух чисел и перевод десятичного числа в двоичную/троичную систему счисления. После этого создадим две программы. В первой будем использовать библиотеку, которую получим на этапе компиляции. Во второй программе с помощью системных вызовов откроем библиотеку и вызовем функции поочередно. Также сделаем возможность замены библиотеки с одной на другую.

**Код программы**

**prog1.c**

#include <stdio.h>

#include "lib1/lib1.h"

int main(){

    int check;

    while(scanf("%d", &check) != EOF){

        if (check != 0 && check != 1 && check != 2){

            printf("Wrong argument\n");

            return 1;

        }

        else if(check == 1){

            int arg1, arg2;

            scanf("%d %d", &arg1, &arg2);

            int result1 = GCF(arg1, arg2);

            printf("NOD is: %d\n", result1);

        }

        else if(check == 2){

            long arg3;

            scanf("%ld", &arg3);

            char\* result2 = translation(arg3);

            printf("From 10 to 2 system: %s\n", result2);

        }

        else{

            printf("See you again!\n");

            return 0;

        }

    }

    return 0;

}

**program2.c**

#include <stdio.h>

#include <dlfcn.h>

int main(){

    int check;

    int key = 2;

    int result1;

    char\* result2;

    int arg1, arg2;

    long arg3;

    void \*current\_lib;

    current\_lib = dlopen("./lib2/lib2.so", RTLD\_LAZY); // Загрузка библиотеки

    if (!current\_lib) {

        printf("Library loading error\n");

        return 1;

    }

    int (\*GCF)(int, int) = dlsym(current\_lib, "GCF"); // Получение указателя на функцию

    char\* (\*translation)(long) = dlsym(current\_lib, "translation");

    if (!GCF) {

        printf("Could not find function\n");

        return 1;

    }

    while(scanf("%d", &check) != EOF){

        if (check != 0 && check != 1 && check != 2){

            printf("Wrong argument\n");

            dlclose(current\_lib); // Закрытие библиотеки

            return 1;

        }

        else if(check == 1){

            scanf("%d %d", &arg1, &arg2);

            result1 = GCF(arg1, arg2);

            printf("NOD is: %d\n", result1);

        }

        else if(check == 2){

            scanf("%ld", &arg3);

            result2 = translation(arg3);

            printf("From 10 to 3 system: %s\n", result2);

        }

        else if(check == 0){

            if (key == 2){

                current\_lib = dlopen("./lib1/lib1.so", RTLD\_LAZY);

                key = 1;

            }

            else{

                current\_lib = dlopen("./lib2/lib2.so", RTLD\_LAZY);

                key = 2;

            }

            if (!current\_lib) {

                printf("Library loading error\n");

                return 1;

            }

            GCF = dlsym(current\_lib, "GCF");

            translation = dlsym(current\_lib, "translaton");

        }

    }

    dlclose(current\_lib);

    return 0;

}

**lib2.c**

#include "lib2.h"

int GCF(int a, int b){

    int min;

    int result = 1;

    if (a < b) min = a;

    else if (a > b) min = b;

    else return a;

    for (int i = 2; i <= min; i++){

        if ((a % i == 0) & (b % i == 0)){

            result = i;

        }

    }

    return result;

}

char\* translation(long x){

    char\* result = (char\*)malloc(sizeof(char));

    int index = 0;

    while(x > 0) {

        result[index] = x % 3 + '0';

        index += 1;

        x /= 3;

        result = realloc(result, (index + 1) \* sizeof(char));

    }

    result[index] = '\0';

    char symbol;

    for(int permutation\_index = 0; permutation\_index < index / 3; ++permutation\_index) {

        symbol = result[permutation\_index];

        result[permutation\_index] = result[index - permutation\_index - 1];

        result[index - permutation\_index - 1] = symbol;

    }

    return result;

    free(result);

}

**lib2.h**

#ifndef MYLIBRARY2

#define MYLIBRARY2

#include <stdio.h>

#include <math.h>

#include <stdlib.h>

int GCF(int a, int b);

char\* translation(long x);

#endif

**lib1.c**

#include "lib1.h"

int GCF(int a, int b){

    while (a != b){

        if (a > b){

            a -= b;

        }

        else{

            b -= a;

        }

    }

    return a;

}

char\* translation(long x){

    char\* result = (char\*)malloc(sizeof(char));

    int index = 0;

    while(x > 0) {

        result[index] = x % 2 + '0';

        index += 1;

        x /= 2;

        result = realloc(result, (index + 1) \* sizeof(char));

    }

    result[index] = '\0';

    char symbol;

    for(int permutation\_index = 0; permutation\_index < index / 2; ++permutation\_index) {

        symbol = result[permutation\_index];

        result[permutation\_index] = result[index - permutation\_index - 1];

        result[index - permutation\_index - 1] = symbol;

    }

    return result;

    free(result);

}

//gcc -c -fPIC lib1.c

//gcc -shared lib1.o -o lib1.so -lm

**lib1.h**

#ifndef MYLIBRARY1

#define MYLIBRARY1

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int GCF(int a, int b);

char\* translation(long x);

#endif

**Протокол работы программы**

**Тестирование:**

cblphblu@DESKTOP-3PDEL6G:/mnt/c/Users/user/Desktop/МАИ/2 курс/ОСИ/LR4FILATOV$ ./prog1

1 12 16

NOD is: 4

2 31

From 10 to 2 system: 11111

0

See you again!

cblphblu@DESKTOP-3PDEL6G:/mnt/c/Users/user/Desktop/МАИ/2 курс/ОСИ/LR4FILATOV$ ./prog2

2 123

From 10 to 3 system: 12110

1 23 23

NOD is: 23

0

1 14 18

NOD is: 2

**Strace:**

cblphblu@DESKTOP-3PDEL6G:/mnt/c/Users/user/Desktop/МАИ/2 курс/ОСИ/LR4FILATOV$ strace -f ./prog1

execve("./prog1", ["./prog1"], 0x7ffce3ff6bb8 /\* 19 vars \*/) = 0

brk(NULL) = 0x55bc5e9fd000

arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7ffe255971b0) = -1 EINVAL (Недопустимый аргумент)

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f7f2f16f000

access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (Нет такого файла или каталога)

openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=35003, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 35003, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f7f2f166000

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\0P\237\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

pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48

pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0i8\235HZ\227\223\333\350s\360\352,\223\340."..., 68, 896) = 68

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=2216304, ...}, AT\_EMPTY\_PATH) = 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, 2260560, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f7f2ef3e000

mmap(0x7f7f2ef66000, 1658880, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7f7f2ef66000

mmap(0x7f7f2f0fb000, 360448, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1bd000) = 0x7f7f2f0fb000

mmap(0x7f7f2f153000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x214000) = 0x7f7f2f153000

mmap(0x7f7f2f159000, 52816, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f7f2f159000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f7f2ef3b000

arch\_prctl(ARCH\_SET\_FS, 0x7f7f2ef3b740) = 0

set\_tid\_address(0x7f7f2ef3ba10) = 105

set\_robust\_list(0x7f7f2ef3ba20, 24) = 0

rseq(0x7f7f2ef3c0e0, 0x20, 0, 0x53053053) = 0

mprotect(0x7f7f2f153000, 16384, PROT\_READ) = 0

mprotect(0x55bc5d2f2000, 4096, PROT\_READ) = 0

mprotect(0x7f7f2f1a9000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7f7f2f166000, 35003) = 0

newfstatat(0, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0

getrandom("\xaa\x8f\xca\x66\x41\xfd\xc1\xc3", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x55bc5e9fd000

brk(0x55bc5ea1e000) = 0x55bc5ea1e000

read(0, 1 5 10

"1 5 10\n", 1024) = 7

newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0

write(1, "NOD is: 5\n", 10NOD is: 5

) = 10

read(0, 2 123

"2 123\n", 1024) = 6

write(1, "From 10 to 2 system: 1111011\n", 29From 10 to 2 system: 1111011

) = 29

read(0, 3

"3\n", 1024) = 2

write(1, "Wrong argument\n", 15Wrong argument

) = 15

lseek(0, -1, SEEK\_CUR) = -1 ESPIPE (Недопустимая операция смещения)

exit\_group(1) = ?

+++ exited with 1 +++

cblphblu@DESKTOP-3PDEL6G:/mnt/c/Users/user/Desktop/МАИ/2 курс/ОСИ/LR4FILATOV$ strace -f ./prog2

execve("./prog2", ["./prog2"], 0x7fff643eaf88 /\* 19 vars \*/) = 0

brk(NULL) = 0x55b69be1c000

arch\_prctl(0x3001 /\* ARCH\_??? \*/, 0x7ffc82ece3a0) = -1 EINVAL (Недопустимый аргумент)

mmap(NULL, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f71a1163000

access("/etc/ld.so.preload", R\_OK) = -1 ENOENT (Нет такого файла или каталога)

openat(AT\_FDCWD, "/etc/ld.so.cache", O\_RDONLY|O\_CLOEXEC) = 3

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=35003, ...}, AT\_EMPTY\_PATH) = 0

mmap(NULL, 35003, PROT\_READ, MAP\_PRIVATE, 3, 0) = 0x7f71a115a000

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\0P\237\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

pread64(3, "\4\0\0\0 \0\0\0\5\0\0\0GNU\0\2\0\0\300\4\0\0\0\3\0\0\0\0\0\0\0"..., 48, 848) = 48

pread64(3, "\4\0\0\0\24\0\0\0\3\0\0\0GNU\0i8\235HZ\227\223\333\350s\360\352,\223\340."..., 68, 896) = 68

newfstatat(3, "", {st\_mode=S\_IFREG|0644, st\_size=2216304, ...}, AT\_EMPTY\_PATH) = 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, 2260560, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f71a0f32000

mmap(0x7f71a0f5a000, 1658880, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x28000) = 0x7f71a0f5a000

mmap(0x7f71a10ef000, 360448, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1bd000) = 0x7f71a10ef000

mmap(0x7f71a1147000, 24576, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x214000) = 0x7f71a1147000

mmap(0x7f71a114d000, 52816, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_ANONYMOUS, -1, 0) = 0x7f71a114d000

close(3) = 0

mmap(NULL, 12288, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_ANONYMOUS, -1, 0) = 0x7f71a0f2f000

arch\_prctl(ARCH\_SET\_FS, 0x7f71a0f2f740) = 0

set\_tid\_address(0x7f71a0f2fa10) = 112

set\_robust\_list(0x7f71a0f2fa20, 24) = 0

rseq(0x7f71a0f300e0, 0x20, 0, 0x53053053) = 0

mprotect(0x7f71a1147000, 16384, PROT\_READ) = 0

mprotect(0x55b69a8ad000, 4096, PROT\_READ) = 0

mprotect(0x7f71a119d000, 8192, PROT\_READ) = 0

prlimit64(0, RLIMIT\_STACK, NULL, {rlim\_cur=8192\*1024, rlim\_max=RLIM64\_INFINITY}) = 0

munmap(0x7f71a115a000, 35003) = 0

getrandom("\xac\xa1\x44\x7f\x4e\x9e\x53\x7a", 8, GRND\_NONBLOCK) = 8

brk(NULL) = 0x55b69be1c000

brk(0x55b69be3d000) = 0x55b69be3d000

**openat(AT\_FDCWD, "./lib2/lib2.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

newfstatat(3, "", {st\_mode=S\_IFREG|0777, st\_size=15576, ...}, AT\_EMPTY\_PATH) = 0

getcwd("/mnt/c/Users/user/Desktop/\320\234\320\220\320\230/2 \320\272\321\203\321\200\321\201/\320\236\320\241\320\230/LR4FILATOV", 128) = 62

mmap(NULL, 16440, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f71a115e000

mmap(0x7f71a115f000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f71a115f000

mmap(0x7f71a1160000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f71a1160000

mmap(0x7f71a1161000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f71a1161000

close(3) = 0

mprotect(0x7f71a1161000, 4096, PROT\_READ) = 0

newfstatat(0, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0

read(0, 1 12

"1 12\n", 1024) = 5

read(0, 2

"2\n", 1024) = 2

newfstatat(1, "", {st\_mode=S\_IFCHR|0620, st\_rdev=makedev(0x88, 0), ...}, AT\_EMPTY\_PATH) = 0

write(1, "NOD is: 2\n", 10NOD is: 2

) = 10

read(0, 0

"0\n", 1024) = 2

**openat(AT\_FDCWD, "./lib1/lib1.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

newfstatat(3, "", {st\_mode=S\_IFREG|0777, st\_size=15576, ...}, AT\_EMPTY\_PATH) = 0

getcwd("/mnt/c/Users/user/Desktop/\320\234\320\220\320\230/2 \320\272\321\203\321\200\321\201/\320\236\320\241\320\230/LR4FILATOV", 128) = 62

mmap(NULL, 16440, PROT\_READ, MAP\_PRIVATE|MAP\_DENYWRITE, 3, 0) = 0x7f71a0f2a000

mmap(0x7f71a0f2b000, 4096, PROT\_READ|PROT\_EXEC, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x1000) = 0x7f71a0f2b000

mmap(0x7f71a0f2c000, 4096, PROT\_READ, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f71a0f2c000

mmap(0x7f71a0f2d000, 8192, PROT\_READ|PROT\_WRITE, MAP\_PRIVATE|MAP\_FIXED|MAP\_DENYWRITE, 3, 0x2000) = 0x7f71a0f2d000

close(3) = 0

mprotect(0x7f71a0f2d000, 4096, PROT\_READ) = 0

read(0, 1 13 26

"1 13 26\n", 1024) = 8

write(1, "NOD is: 13\n", 11NOD is: 13

) = 11

read(0, 3

"3\n", 1024) = 2

write(1, "Wrong argument\n", 15Wrong argument

) = 15

munmap(0x7f71a0f2a000, 16440) = 0

lseek(0, -1, SEEK\_CUR) = -1 ESPIPE (Недопустимая операция смещения)

exit\_group(1) = ?

+++ exited with 1 +++

**Вывод**

В ходе выполнения данной лабораторной работы я научился использовать библиотеки двумя способами. Было интересно узнать некоторые особенности второго способа. К тому же, при выполнении работы можно было легко отслеживать прогресс, ибо реализация самих функций НОД и перевода в другую систему счисления далась мне легко.