

Міністерство освіти і науки України
Львівський національний університет імені Івана Франка

Факультет електроніки та комп'ютерних технологій
Кафедра радіоелектронних і комп'ютерних систем

Звіт

про виконання лабораторної роботи №9
“Застосування динамічних бібліотек в ОС Windows і Linux ”

Виконав
студент групи Фел-23
Морозенко Богдан Андрійович
Перевірив
ас. Сінькевич О. О.

Львів – 2019

Хід роботи

Реалізувати застосування для Linux і Windows, що може бути розширене під час виконання. Інтерфейс модуля розширення функцій типу void без параметрів. Після запуску застосування видає на екран підказку й очікує введення команди з клавіатури. Можливі такі команди: load ім'я_модуля (завантаження модуля в пам'ять), unload ім'я_модуля (вилучення модуля з пам'яті) call ім'я_функції (виклик функції з модуля). Кожен модуль розширення повинен містити код, який виконується під час його завантаження в пам'ять і вилучення з пам'яті.

Код програми

lib1.c

```
#include <stdio.h>

void print1(){

    printf("Hello, my friend");

}
```

lib2.c

```
#include <stdio.h>

void print2(){

    printf("Have a nice day!");

}
```

main.c

```
#include <dlfcn.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
double (*fn)();
void* lib = NULL;
void load(char* libName)
{
```

```

if (lib != NULL)
{
printf("Some library already loaded!\n");
return;
}
lib = dlopen(libName, RTLD_NOW);
if (!lib)
{
fprintf(stderr, "%s\n", dlerror());
}
else
{
printf("Library %s loaded.\n", libName);
}
}

void call(char* funcName)
{
if (lib == NULL)
{
printf("Any library doesn't currently loaded!\n");
return;
}
fn = dlsym(lib, funcName);
char* error;
if ((error = dlerror()) != NULL)
{
fprintf(stderr, "%s\n", error);
return;
}
(*fn)();
printf("\n");
}

void unload(char* libName) {
if (lib == NULL)
{
printf("Any library doesn't currently loaded!\n");
return;
}
dlclose(lib);

```

```

printf("Unload %s library.\n", libName);
}
int main()
{
char command[50];
printf("Library loader.\n");
while(1) {
printf("\n>");
scanf("%s", command);
if (strcmp(command, "load") == 0)
{
printf("\nEnter path : ");
scanf("%s", command);
load(command);
}
else if (strcmp(command, "unload") == 0)
{
printf("\nEnter path : ");
scanf("%s", command);
unload(command);
}
else if (strcmp(command, "call") == 0)
{
printf("\nInput function name : ");
scanf("%s", command);
call(command);
}
else if (strcmp(command, "exit") == 0)
{
return 0;
}
else
{
printf("\nIncorrect input!\n");
}
}
return 0;
}

```

Результати роботи

```
bohdan@bohdan-VirtualBox:~/lab9$ gcc -c lib1.c
bohdan@bohdan-VirtualBox:~/lab9$ gcc -shared -o lib1.so lib1.o
bohdan@bohdan-VirtualBox:~/lab9$ gcc -c lib2.c
bohdan@bohdan-VirtualBox:~/lab9$ gcc -shared -o lib2.so lib2.o
bohdan@bohdan-VirtualBox:~/lab9$ gcc main.c -ldl
```

```
bohdan@bohdan-VirtualBox:~/lab9$ ./a.out
Library loader.

>load

Enter path : /home/bohdan/lab9/lib2.so
Library /home/bohdan/lab9/lib2.so loaded.

>call

Input function name : print2
Have a nice day!

>unload

Enter path : /home/bohdan/lab9/lib2.so
Unload /home/bohdan/lab9/lib2.so library.

>
```

```
>load

Enter path : /home/bohdan/lab9/lib1.so
Library /home/bohdan/lab9/lib1.so loaded.

>call

Input function name : print1
Hello, my friend

>unload

Enter path : /home/bohdan/lab9/lib1.so
Unload /home/bohdan/lab9/lib1.so library.

>
```

Висновок

На цій практичній роботі я спробував застосовувати динамічні бібліотеки у ОС Linux та Windows, навчився компілювати та динамічно викликати dll-бібліотеки з-під програми.