

Министерство образования Республика Беларусь
Учреждение образования “Брестский Государственный Технический
Университет
Кафедра ИИТ

Лабораторная работа № 3
По дисциплине «СПО»
«Работа с динамическими библиотеками»

Выполнил:
Студент 3 курса
Гр. АС-50
Булей Е.В.
Проверила:
Давидюк Ю.И.

Брест 2020

Задания – Windows

Разработать динамическую библиотеку DLL, включающую функцию (функции), реализующую следующий функционал для работы со строками (использовать только стандартные средства – библиотека **string.h**). Продемонстрировать ее подключение и использование.

- 1) Для двух строк, переданных в качестве параметров, получить итоговую строку-результат, представляющую собой комбинацию переданных, составленную по следующему правилу: 1-я буква из 1-й строки, 2-я – из 2-й, 3-я – из 1-й, 4 – из 2-й и т.д.
- 2) Выполнить конкатенацию двух строк по правилу: первая строка остается без изменений, а вторая реверсируется.
- 3) Выполнить операцию «перекрещивания» двух строк, переданных в качестве параметров (для заданных позиций в обеих строках, первая часть новой первой строки берется из первой переданной строки до заданной позиции, а вторая часть – из второй переданной после заданной позиции во второй строке, для второй строки – тоже самое, только берутся альтернативные части оригинальных строк), полученные строки конкатенировать. Пример: для строк «abcd» и «hgjfs» и точек скрещивания 2 и 3 (для первой и второй строки соответственно) результат скрещивания будет «abfs» и «hgjcd», после итоговой конкатенации – «abfshgjcd».

```
workWithDll.cpp
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <windows.h>
#include <string.h>
using namespace std;

int main()
{
    int pos_1 = 0, pos_2 = 0;
    char* str1, * str2;
    str1 = new char[30];
    str2 = new char[30] ;

    cout << "Enter first word: ";
    cin >> str1;
    cout << "Enter second word: ";
    cin >> str2;
    cout << "Enter position 1: ";
    cin >> pos_1;
    cout << "Enter position 2: ";
    cin >> pos_2;
    HMODULE hlib = LoadLibrary(TEXT("Dll1.dll")); //загружаем библиотеку
    if (hlib == NULL) cout << "ERROR_1" << endl;

    typedef void (*function)(char *str1, char *str2, int pos_1, int pos_2);
    function func = (function)GetProcAddress(hlib, "func");
    if (func == NULL) cout << "ERROR_2" << endl;
    else func(str1, str2, pos_1, pos_2);

    FreeLibrary(hlib); //выгружаем
    system("pause");
}

Dll_h.h
#pragma once
char* workWithStr(char* str1, char* str2, int position_1, int position_2);
extern "C" __declspec(dllexport) void func(char* str1, char* str2, int position_1, int position_2);

Source.cpp
#include "pch.h"
#include "Dll_h.h"
#include <iostream>
#include <string.h>
using namespace std;
```

```

void func(char* str1, char* str2, int position_1, int position_2) {
    int len1 = strlen(str1);
    int len2 = strlen(str2);
    if (position_1 >= len1 || position_1 >= len2 || position_2 >= len1 || position_2 >= len2) {
        cout << "ERROR, enter other position \n";
        return;
    }
    char* res_1, * res_2;
    res_1 = workWithStr(str1, str2, position_1, position_2);
    res_2 = workWithStr(str2, str1, position_2, position_1);
    int lenR_1, lenR_2;
    lenR_1 = strlen(res_1); lenR_2 = strlen(res_2);
    char* result = new char[lenR_1 + lenR_2 + 1];
    for (int i = 0, j = 0; i < lenR_1 + lenR_2; i++) {
        if (i < lenR_1) result[i] = res_1[i];
        else { result[i] = res_2[j]; j++; }
    }
    result[lenR_1 + lenR_2] = '\0';
    cout << result << endl;
}

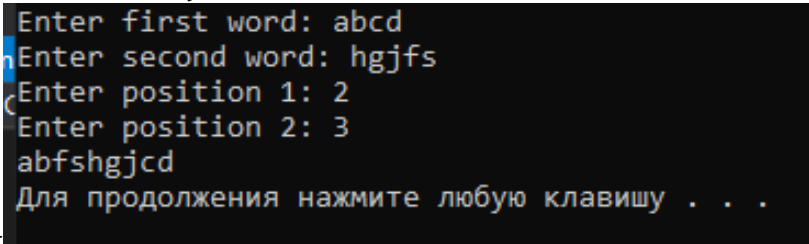
char* workWithStr(char* str1, char* str2, int position_1, int position_2) {
    int len1 = strlen(str1);
    int len2 = strlen(str2);
    char* str = new char[position_1 + len2 - position_2 + 1];

    for (int i = 0; i < position_1; i++) {
        str[i] = str1[i];
    }

    for (int i = position_2, j = position_1; i < len2; i++, j++) {
        str[j] = str2[i];
    }

    str[position_1 + len2 - position_2] = '\0';
    return str;
}

```



```

Enter first word: abcd
Enter second word: hgjfs
Enter position 1: 2
Enter position 2: 3
abfshgjc d
Для продолжения нажмите любую клавишу . . .

```

Задания – Linux

Разработать динамическую библиотеку со, включающую функцию (функции), реализующую следующий функционал. Продемонстрировать ее подключение и использование. Для поиска необходимых функций использовать ресурс <https://www.die.net>.

Все функции работают с домашней директорией пользователя!

- 1) Функция, создающая директорию с заданным именем и директории в ней с числовыми именами от 0 до заданного параметра n
- 2) Функция, дающая файлам в заданной директории новые последовательные числовые имена
- 3) Функция, читающая содержимое указанной директории и выводящая его на экран

func.c

```
#include <dirent.h>
```

```
#include <stdio.h>
```

```
#include <stddef.h>
```

```

#include <stdlib.h>
#include <string.h>
extern void func(const char path[]){
    DIR *dir = opendir(path);
    struct dirent *next_dir;
    int i=0;
    while((next_dir = readdir(dir))!= NULL){
        if( (*next_dir).d_type == DT_REG || (*next_dir).d_type == DT_DIR){
            printf("%s \n",(*next_dir).d_name);
        }
    }
    printf("DONE\n");
    closedir(dir);
}

```

Main.c

```

#include <dirent.h>
#include <stdio.h>
#include <stddef.h>
#include <stdlib.h>
#include <dlfcn.h>

int main(){
    void *handle;
    char *error;
    char path[20];
    printf("Enter path(Example /home/evgeny/): ");
    scanf("%s",path);
    handle = dlopen("/home/evgeny/lab_sem_8/func.so", RTLD_LAZY);
    if (!handle) {
        fprintf(stderr, "%s\n", dlerror());
        return 0;
    }

    dlerror();
    void(*func)(char[]) = dlsym(handle,"func");

    if ((error = dlerror()) != NULL) {
        fprintf(stderr, "%s\n", error);
        return 0;
    }
    func(path);
    dlclose(handle);
    return 0;}

```

```
evgeny@E-VB:~/lab_sem_8$ gcc -fPIC -c func.c
evgeny@E-VB:~/lab_sem_8$ gcc -shared -ofunc.so func.o
evgeny@E-VB:~/lab_sem_8$ gcc main.c -ldl
evgeny@E-VB:~/lab_sem_8$ ./a.out
Enter path(Example /home/evgeny/): /home/evgeny/1/
Untitled Folder
.
1
3
4
2
..
0
Untitled Folder 2
DONE
evgeny@E-VB:~/lab_sem_8$
```

