

Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

ОТЧЕТ

ЛАБОРАТОРНАЯ РАБОТА №3

«Динамические библиотеки в ОС Windows и Linux»

СПО

Выполнил студент
группы АС-50:
Протасевич А.В.
Проверил:
Давидюк Ю.И.

Брест 2020

Вариант 18

Цель работы:

закрепить навыки решения задач на строки и манипуляции с файловыми системами, научиться проектировать и разрабатывать динамические библиотеки в ОС Windows и Linux

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.

8) Для строки-параметра получить строку-результат конкатенацией n копий исходной строки

DLL библиотека:

Файл заголовка:

```
#pragma once
#ifdef KONKATENACIA_EXPORTS
#define KONKATENACIA_API __declspec(dllexport)
#else
#define KONKATENACIA_API __declspec(dllimport)
#endif

extern "C" KONKATENACIA_API void fnkonkatenacia(char str[], int n);
```

Исходный файл:

```
#include "pch.h"
#include "framework.h"
#include "konkatenacia.h"
#include "cstring"
#include "iostream"
#pragma warning(disable : 4996).

void fnkonkatenacia(char str[], int n)
{
    char s[1000];
    strcpy(s, str);
    for (int i = 0; i < n; i++) {
        strcat(str, s);
    }
    std::cout << "Result: " << std::endl;
    std::cout << str;
}
```

Приложение использующее библиотеку:

Исходный файл:

```
#include "pch.h"
#include <iostream>
#include "konkatenacia.h"

int main()
{
    char a[1000] = "Hello world!!";
    std::cout << "Original string:" << std::endl << a<<std::endl;
```

```

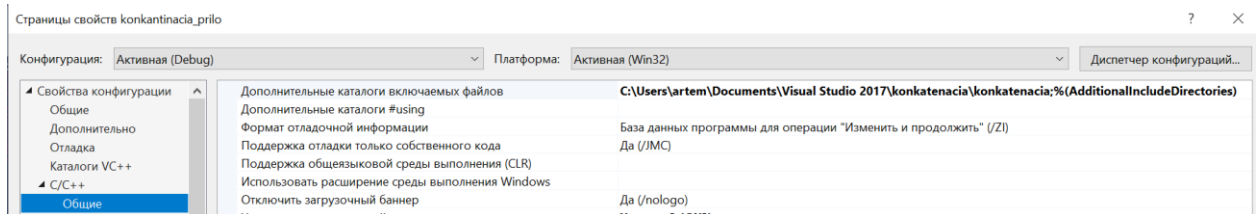
std::cout << "n= " << 3 << std::endl;
fnkonkatenacia(a,3);
return 0;
}

```

Подключение библиотеки:

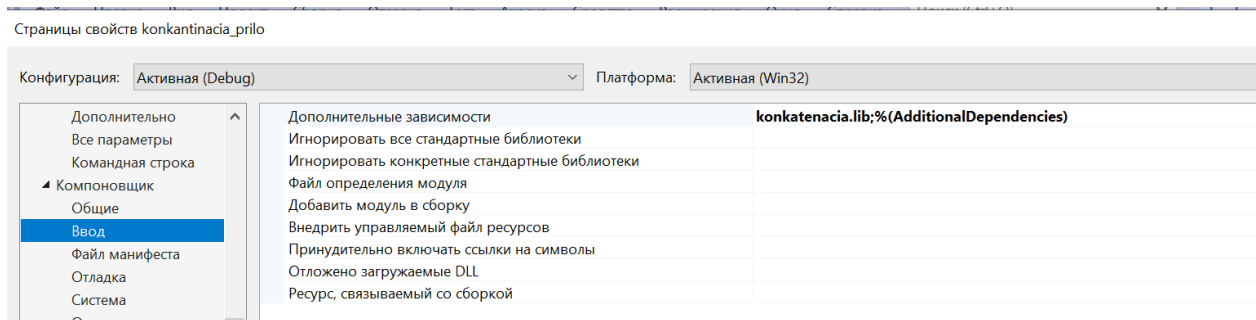
Добавление заголовка библиотеки DLL в путь включени

Изменяем поле Дополнительные каталоги включаемых файлов

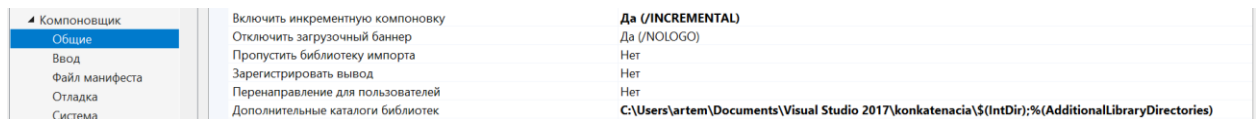


Добавление библиотеки импорта DLL в проект

Изменяем поле Дополнительные зависимости

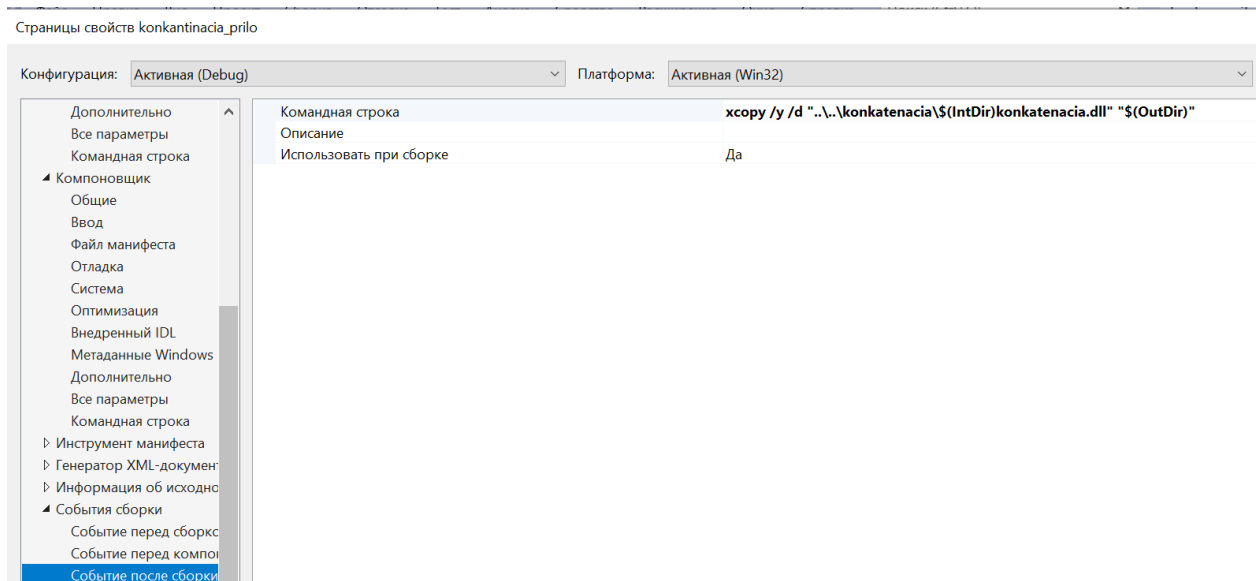


Изменяем поле Дополнительные каталоги библиотек



Копирование библиотеки DLL в событие после сборки

Изменяем поле Командная строка



Запускаем приложение с подключенной библиотекой



```
Консоль отладки Microsoft Visual Studio
Original string:
Hello world!!
n= 3
Result:
Hello world!!Hello world!!Hello world!!Hello world!!
```

Задания – Linux

Разработать динамическую библиотеку `so`, включающую функцию (функции), реализующую следующий функционал. Продемонстрировать ее подключение и использование.

8) Функция, проверяющая существование указанного файла

Файл `lib8.c`:

```
#include <stdio.h>

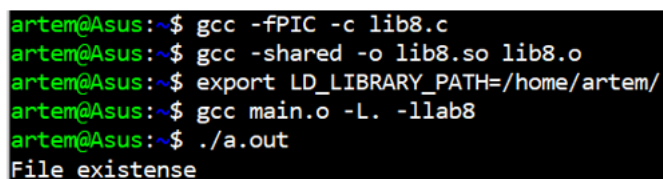
#include <unistd.h>

int file_is_ok(const char *name)
{
    return access(name, F_OK);
}
```

Файл `main.c`:

```
#include <stdio.h>

int main()
{
    int status = file_is_ok("str");
    if (status == 0)
        puts("File existence");
    else
        puts("Error");
    return 0;
}
```



```
artem@Asus:~$ gcc -fPIC -c lib8.c
artem@Asus:~$ gcc -shared -o lib8.so lib8.o
artem@Asus:~$ export LD_LIBRARY_PATH=/home/artem/
artem@Asus:~$ gcc main.o -L. -llab8
artem@Asus:~$ ./a.out
File existence
```

Вывод: закрепил навыки решения задач на строки и манипуляции с файловыми системами.