

МИНИСТЕРСТВО ОБРАЗОВАНИЯ ОРЕНБУРГСКОЙ ОБЛАСТИ  
Государственное автономное профессиональное образовательное  
учреждение  
**«ОРЕНБУРГСКИЙ КОЛЛЕДЖ ЭКОНОМИКИ И ИНФОРМАТИКИ»  
(ГАПОУ ОКЭИ)**

**Практическая работа**

*ОКЭИ 09.02.07 7026 07 У*

Тема: *«Работа со динамической библиотекой»*

Выполнил: *Гадияев Ислам Ильгамович*

Оренбург 2026

Содержание

1 Цель работы ..... 3

2 Скриншот структуры готового проекта..... 3

3 Листинг программы ..... 3

4 Результаты работы программы..... 4

					ОКЭИ 09.02.07 7026 07 У								
Изм.	Лист	№ докум.	Подп.	Дата									
Разраб.	Гадияев И. И.				Практическая работа						Лит.	Лист	Листов
											У	2	
											Отделение программирования		

## 1 Цель работы

Задание: создать динамическую библиотеку, реализующую вычисление факториала числа и проект, ее использующий по алгоритму из практической части ниже

## 2 Скриншот структуры готового проекта

На рисунке 1 представлен скриншот структуры готового проекта.

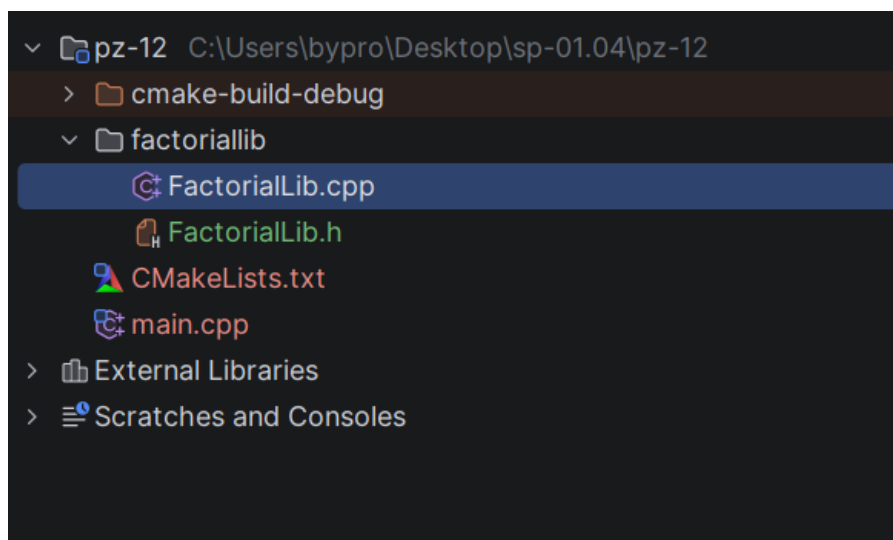


Рисунок 1 – Структура проекта

## 3 Листинг программы

Листинг файла main.cpp представлен ниже.

```
#include <iostream>
#include "factoriallib/FactorialLib.h"

using namespace std;

int main() {
    unsigned int n;
    cout << "Enter n: ";
    cin >> n;

    cout << n << "! = " << factorial(n) << "\n";
    return 0;
}
```

Листинг файла FactorialLib.cpp представлен ниже.

```
#include "FactorialLib.h"

unsigned long long factorial(unsigned int n)
{
    if (n == 0) return 1;
    return n * factorial(n - 1);
}
```

```

    unsigned long long result = 1;

    for (unsigned int i = 2; i <= n; i++)
        result *= i;

    return result;
}

```

Листинг файла FactorialLib.h представлен ниже.

```

#pragma once

#ifdef _WIN32
    #ifdef FACTORIALLIB_EXPORTS
        #define FACTORIALLIB_API __declspec(dllexport)
    #else
        #define FACTORIALLIB_API __declspec(dllimport)
    #endif
#else
    #define FACTORIALLIB_API
#endif

extern "C" FACTORIALLIB_API unsigned long long factorial(unsigned int n);

```

Листинг файла CMakeLists.txt представлен ниже.

```

cmake_minimum_required(VERSION 4.1)
project(pz_12)

set(CMAKE_CXX_STANDARD 20)

add_library(FactorialLib SHARED
    factoriallib/FactorialLib.cpp
)

target_include_directories(FactorialLib PUBLIC
    ${CMAKE_CURRENT_SOURCE_DIR}/factoriallib
)

target_compile_definitions(FactorialLib PRIVATE FACTORIALLIB_EXPORTS)

add_executable(pz_12 main.cpp)

target_link_libraries(pz_12 PRIVATE FactorialLib)

add_custom_command(TARGET pz_12 POST_BUILD
    COMMAND ${CMAKE_COMMAND} -E copy_if_different
    ${<TARGET_FILE:FactorialLib>}
    ${<TARGET_FILE_DIR:pz_12>}
)

```

## 4 Результаты работы программы

Результат выполнения программы показан на рисунке 2.

```
C:\Users\bypro\Desktop\sp-01.04\pz-12\cmake-build-debug\pz_12.exe
Enter n:3

3! = 6

Process finished with exit code 0
```

Рисунок 2 – Результат работы программы