Міністерство освіти і науки України

Національний технічний університет України

“Київський політехнічний інститут”

Кафедра АСОІУ

**ЗВІТ**

про виконання комп’ютерного практикуму № 9

з дисципліни

“ООП”

Тема: «DLL С++»

|  |  |  |
| --- | --- | --- |
| Прийняв: |  | Виконав: |
| Головченко  Максим  Миколайович |  | студент 2-го курсу  гр. ІП-51 ФІОТ  Субцельний Олександр Володимирович |

Київ – 2016

**ЗМІСТ:**

[1 Мета роботи 3](#_Toc463896091)

[2 Постановка задачі (варіант 20, рівень В) 4](#_Toc463896092)

[3 Діаграма класів 5](#_Toc463896093)

[4 Код програми 6](#_Toc463896094)

[5 Приклади виконання програми 10](#_Toc463896095)

[6 Висновок 11](#_Toc463896096)

# Мета роботи

Цель работы – изучить особенности использования DLL, рассмотреть принципы статического и динамического подключения.

# Постановка задачі (варіант 20, рівень В)

Преобразовать матрицу так, чтобы все элементы выше главной диагонали были нулевыми. Вывести исходную и полученную матрицы.

Использовать динамическое подключение библиотеки

# Код програми

“main.cpp”

#include <iostream>

#include <windows.h>

using namespace std;

typedef void (\*FunctionFunc)();

typedef void (\*generateMatrixFunc)(int, int\*\*);

typedef void (\*ShowFunc)(int, int\*\*);

typedef void (\*changeMatrixFunc)(int, int\*\*);

int main()

{

int x;

cout<<"VVEDITE width of MATRIX"<<endl;

cin>>x;

int \*\*arr = new int\* [x];

for (int coun = 0; coun < x; coun++)

arr[coun] = new int [x];

FunctionFunc \_FunctionFunc;

generateMatrixFunc \_generateMatrixFunc;

changeMatrixFunc \_changeMatrixFunc;

ShowFunc \_ShowFunc;

HINSTANCE hInstLibrary = LoadLibrary("DLL\_MATRIX.dll");

if (hInstLibrary)

{

\_FunctionFunc = (FunctionFunc)GetProcAddress(hInstLibrary,"Function");

\_generateMatrixFunc = (generateMatrixFunc)GetProcAddress(hInstLibrary,"generateMatrix");

\_changeMatrixFunc = (changeMatrixFunc)GetProcAddress(hInstLibrary,"changeMatrix");

\_ShowFunc = (ShowFunc)GetProcAddress(hInstLibrary,"Show");

if (\_FunctionFunc)

{

\_FunctionFunc();

}

if (\_generateMatrixFunc)

{

\_generateMatrixFunc(x,arr);

}

if (\_ShowFunc)

{

\_ShowFunc(x,arr);

}

if (\_changeMatrixFunc)

{

\_changeMatrixFunc(x,arr);

}

cout<<endl;

if (\_ShowFunc)

{

\_ShowFunc(x,arr);

}

FreeLibrary(hInstLibrary);

}

else

{

std::cout << "DLL Failed To Load!" << std::endl;

}

std::cin.get();

for (int coun = 0; coun < x; coun++)

delete [] arr[coun];

return 0;

}

“DLL\_MATRIX.h”

#ifndef \_DLL\_MATRIX\_H\_

#define \_DLL\_MATRIX\_H\_

#include <iostream>

#include <stdlib.h>

#include <time.h>

#if defined DLL\_EXPORT

#define DECLDIR \_\_declspec(dllexport)

#else

#define DECLDIR \_\_declspec(dllimport)

#endif

extern "C"

{

DECLDIR void Function( void );

DECLDIR void generateMatrix(int x,int \*\*matr);

DECLDIR void changeMatrix(int x,int \*\*matr);

DECLDIR void Show (int x,int \*\*matr);

}

#endif

“DLL\_MATRIX.cpp”

#define DLL\_EXPORT

#include "DLL\_MATRIX.h"

using namespace std;

extern "C"

{

DECLDIR void Function(void)

{

std::cout << "DLL Called!" << std::endl;

}

DECLDIR void generateMatrix(int x,int \*\*matr)

{

for (int count\_row = 0; count\_row < x; count\_row++)

for (int count\_column = 0; count\_column < x; count\_column++)

matr[count\_row][count\_column] = (rand() % 10 + 1);

}

DECLDIR void Show (int x,int \*\*matr)

{

for (int count\_row = 0; count\_row < x; count\_row++){

for (int count\_column = 0; count\_column < x; count\_column++)

cout<<matr[count\_row][count\_column]<<" ";

cout<<endl;

}

}

DECLDIR void changeMatrix(int x,int \*\*matr)

{

for (int count\_row = 0; count\_row < x; count\_row++)

for (int count\_column = 0; count\_column < x; count\_column++)

if (count\_row>=count\_column) matr[count\_row][count\_column] = 0;

}

}

“libDLL\_Matrix.def”

EXPORTS

Function @1

Show @2

changeMatrix @3

generateMatrix @4

# Приклади виконання програми

Приклад виконання програми наведений на рисунку 5.1, рисунку 5.2:

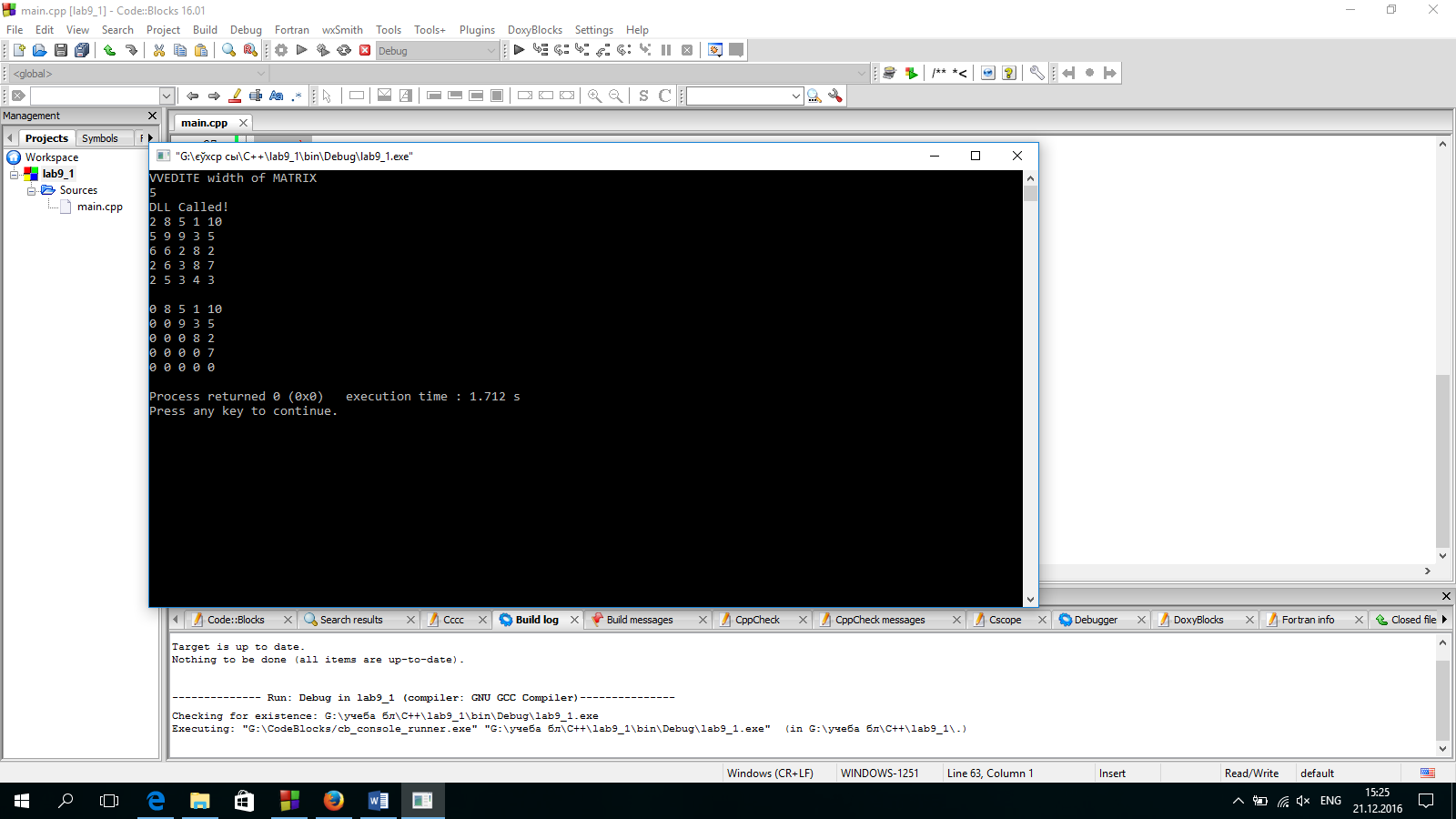


Рисунок 5.1 – Приклад виконання програми

# Висновок

Отже, дана програма динамічно підгружає бібліотеку DLL\_MATRIX, яка була створена попередньо. Програма рандомом генерує матрицю, виводить її, потім змінює усі елементи вище головної діагоналі на 0 і вивлодить її знову. В бібліотеці DLL\_MATRIX реалізовано 3 функції generateMatrix, Show, changeMatrix.