Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»



Звіт

З лабораторної роботи № 3, варіант 8

З дисципліни « ПРОГРАМУВАННЯ, ЧАСТИНА 2 (ОБ’ЄКТНООРІЄНТОВАНЕ ПРОГРАМУВАННЯ) »

Виконав: ст. гр. КІ-15

Гербей О.М.

Перевірив: викладач

Козак Н.Б

Львів – 2020

Завдання: Задано квадратну матрицю, всі елементи якої рівні одиниці. Написати функцію void func (int\*\* arr, int n), котра заповняє заштриховану область матриці (згідно варіанту) нулями. Розмір масиву вводиться з клавіатури

#include <iostream>

#include <iomanip>

using namespace std;

// заповнення масиву одиницями

void fill\_array\_one(int\*\*, size\_t);

// заповнення масиву по схемі

void fill\_array\_scheme(int\*\*, size\_t);

// вивід масиву

void print\_array(int\*\*, size\_t);

int main() {

size\_t n;

int\*\* my\_arr;

cout << "Enter size" << endl;

cin >> n;

my\_arr = new int\* [n];

for (size\_t i = 0; i < n; i++)

my\_arr[i] = new int[n];

cout << "Filled with 1\n" << endl;

fill\_array\_one(my\_arr, n);

print\_array(my\_arr, n);

cout << "Filled according to the scheme\n" << endl;

fill\_array\_scheme(my\_arr, n);

print\_array(my\_arr, n);

for (size\_t i = 0; i < n; i++)

delete[] my\_arr[i];

delete[] my\_arr;

return 0;

}

void fill\_array\_one(int\*\* arr, size\_t n) {

for (size\_t i = 0; i < n; ++i)

for (size\_t j = 0; j < n; ++j)

arr[i][j] = 1;

}

void fill\_array\_scheme(int\*\* arr, size\_t n) {

for (size\_t i = 0; i < n / 2; ++i) {

for (size\_t j = n / 2; j < n; ++j) {

if (j + i >= n - 1)

arr[i][j] = 0;

}

}

for (size\_t i = n / 2; i < n; ++i) {

for (size\_t j = n / 2; j < n; j++) {

if (j >= i)

arr[i][j] = 0;

}

}

}

void print\_array(int\*\* array, size\_t n) {

for (size\_t i = 0; i < n; ++i) {

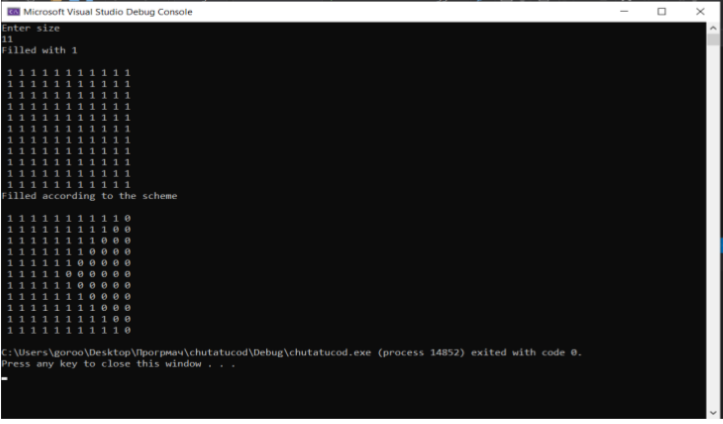
for (size\_t j = 0; j < n; ++j)

cout << setw(2) << array[i][j];

cout << endl;

}

}



Висновок: виконюючи цю лабораторну роботу, я ознайомився із засобами розботи з динамічною пам’яттю та динамічними масивами.