МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Дніпровський національний університет  
залізничного транспорту імені академіка В. Лазаряна**

Кафедра «Комп’ютерні інформаційні технології»

**Лабораторна робота №4**

**з дисципліни «Основи програмування»**

**на тему: *«*Динамичні масиви в мові С++.  Обробка матриць.»**

Виконав: студент гр. ПЗ2011

Кулик Сергій Вадимович

Прийняла: доц. Нежуміра О.І.

Дніпро, 2020

**Лабораторна робота №4**

**Тема.** Динамичні масиви в мові С++.  Обробка матриць.

**Мета.** Набути практичних навичок розробки алгоритмів, написання і налагодження програм для створення і обробки різного типу структур динамічних матриць.

Варіант 5

1. Потсановка задачі

В м. Дніпро протягом вересня кожен час доби заміряли швидкість повітря. Результати заносили в таблицю, рядки якої відповідають дням місяця.  
1)    визначити день і час, коли в перший раз була зафіксована найбільша швидкість повітря;  
2)    відсортувати рядки таблиці за полуденною швидкістю повітря по зростанню.

2. Вимоги до програми:

1)    обробка некоректного введення користувачем розмірності матриці;

2)    виділення і звільнення пам’яті для матриць виконується операторами *new* і *delete*;

3)    заповнення обох матриц паралельно випадковими значеннями відповідно до умов завдання (в результаті отримуються матриці, елементи яких однакові);

4)    використання для кожної частини завдання динамічної матриці такої структури, яка є більш ефективною для її виконання (економія часу, пам’яті, короткий код);

5)   виведення результатів роботи програми на екран;

6)    реалізація частин завдання у вигляді функцій.

3. Зовнішні специфікації

Вхідні дані:

1) Дні місяця. 2) Години доби.

Вимоги до формату вхідних даних наведені у табл.1

**Формат вхідних даних**

Таблиця 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Найменування даних | Умовне позначення | Вимоги до даних | Приклад |
| 1 | Кількість днів місяця | day | Натуральне число | 30 |
| 2 | Години добини | hours | Натуральне число | 24 |

Вихідні дані:

1) Найбільша швидкість вітру.  
2) Відсортований масив по полуденній швидкості вітру.  
3) День і час коли була зафіксована найбільша швидкість вітру

Вимоги до вихідних даних наведені в таблиці 2

**Формат вихідних даних**

Таблиця 2

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Найменування даних | Умовне позначення | Вимоги до даних | Приклад |
| 1 | Найбільша швидкість вітру | max | Дійсне додатнє число | 24.5 |
| 2 | Відсортований массив по 12 стовпчику | arr | Дійсні додатні числа | 2.5; 3; 14;2 |
| 3 | День і час з найбільшою швидкістю вітру | max\_i  max\_j | Натуральні числа | 5, 3, 6 |

4. Функціональні вимоги до программи

1) Створення двомірного динамічного массиву для запису швидкості вітру та заповнення його випадковим числами.

2) Знаходження в масиві найбільшої швидкості вітру та виводу дня та часу на екран.

3) Сортування массиву по полуденній швидкості вітру( 12 стовпчику).

4) Представлення одномірного массиву у вигляді матриці.

5. Вибір методу рішення

1) Завести змінну для записумаксимального значення, за допомогою двох циклів порівнювати наступне значення масиву з записаним максимальним. Якщо наступне значення більше ніж записане, замінити його.

2) Для сортування массиву використовував метод сортування бульбашкою по 12 стовпчику.

6. Тести

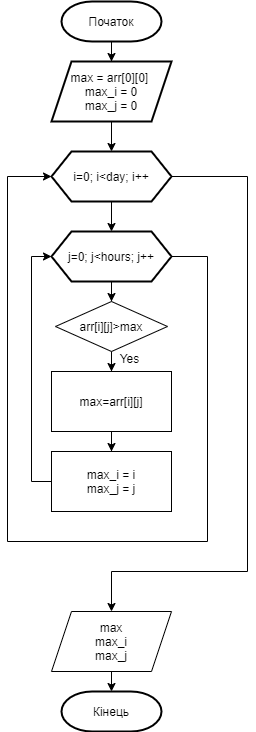
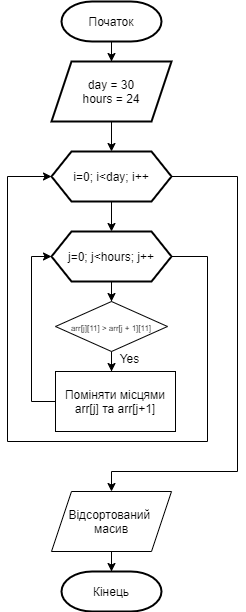
Таблиця 3

|  |  |  |  |
| --- | --- | --- | --- |
| № | Назва | Вхідні дані | Очікувані результати |
| 1. | Початковий масив | day = 30  hours = 24 | Масив заповнено  випадковими дійсними числами >0 |
| 2. | Максимальна швидкість  вітру | day = 30  hours = 24 | Алгоритм знайде в масиві найбільше значення, та виведе рядок та стовпчик |
| 3 | Відсортовани й массив | day = 30  hours = 24 | В 12-у стовпчику елементи будуть розміщуватись по-зростанню. Рядки також будуть мінятись місцями. |

7. Алгоритм



Алгоритм пошуку максимального Алгоритм сортування   
 елемента

8. Текст программи

//В м.Дніпро протягом вересня кожен час доби заміряли швидкість повітря.Результати заносили в таблицю, рядки якої відповідають дням місяця.

//1) визначити день і час, коли в перший раз була зафіксована найбільша швидкість повітря;

//2) відсортувати рядки таблиці за полуденною швидкістю повітря по зростанню.

#include<iostream>

#include<iomanip>

using namespace std;

//Заполнение массива

void fill(float\*\* arr, int day, int hours)

{

for (int i = 0; i < day; i++)

{

for (int j = 0; j < hours; j++)

{

arr[i][j] = static\_cast<float>(rand() % 255) / 10;

}

}

}

//Вывод массива

void print(float\*\* arr, int day, int hours)

{

for (int i = 0; i < day; i++)

{

cout << i + 1 << " Сентября ";

for (int j = 0; j < hours; j++)

{

cout << left << setw(6) << arr[i][j];

}

cout << endl << endl;

}

}

//Поиск максимального значения

void maximum(float\*\* arr, int day, int hours)

{

int max\_i = 0;

int max\_j = 0;

float max = arr[0][0];

for (int i = 0; i < day; i++)

{

for (int j = 0; j < hours; j++)

{

if (arr[i][j] > max)

{

max = arr[i][j];

max\_i = i;

max\_j = j;

}

}

}

cout << "======================================================================" << endl << endl;

cout << "Максимальная скорость ветра: " << max << "м/с была " << max\_i + 1 << " сентября," << " в " << max\_j << ":00" << endl << endl;

cout << "======================================================================" << endl;

}

//Сортировка

void sort(float\*\* arr, int day, int hours)

{

cout << "======================================================================" << endl << endl;

cout << "Сортировка таблици за полуденною скоростью ветра" << endl << endl;

cout << "======================================================================" << endl;

for (int i = 0; i < day - 1; i++)

{

for (int j = 0; j < day - i - 1; j++)

{

if (arr[j][12] > arr[j + 1][12])

{

swap(arr[j], arr[j + 1]);

}

}

}

}

//Запослнение одномерного массива

void fillarr2(float\* array, int n, int m)

{

for (int i = 0; i < n \* m; i++)

{

array[i] = static\_cast<float>(rand() % 255) / 10;

}

}

//Вывод одномерного массива в виде матрици

void printarr2(float\* array, int day, int hours)

{

int k = 0;

for (int i = 0; i < day; i++)

{

cout << i + 1 << " Сентября ";

for (int j = 0; j < hours; j++, k++)

{

cout << setw(6) << array[k];

}

cout << endl << endl;

}

}

//Поиск максимального значения

void maximum2(float\* array, int day, int hours)

{

float max2 = array[0];

int max2\_i = 0;

int max2\_j = 0;

for (int i = 0; i < day\*hours; i++)

{

for (int j = 0; j < hours; j++)

{

if (array[i] > max2)

{

max2 = array[i];

max2\_i = i / hours + j;

max2\_j = i % hours + j;

}

}

}

cout << "======================================================================" << endl << endl;

cout << "Максимальная скорость ветра: " << max2 << "м/с была " << max2\_i + 1 << " сентября," << " в " << max2\_j << ":00" << endl << endl;

cout << "======================================================================" << endl;

}

int main()

{

setlocale(LC\_ALL, "ru");

srand(0);

int day = 30;

int hours = 24;

//===========================| Инициализация массива |=========================

float\*\* arr = new float\* [day];

for (int i = 0; i < day; i++)

{

arr[i] = new float[hours];

}

//=============================| Обработка массива |==========================

fill(arr, day, hours);

print(arr, day, hours);

maximum(arr, day, hours);

system("pause");

sort(arr, day, hours);

print(arr, day, hours);

system("pause");

//===========================| Удаление из памяти |=========================

for (int i = 0; i < hours; i++)

{

delete[] arr[i];

}

delete[] arr;

//============================| 2-й способ представления массива |===========================

srand(0);

float\* array = new float[day \* hours];

cout << "=================================" << endl << endl;

cout << "2-й способ представления массива" << endl << endl;

cout << "=================================" << endl;

fillarr2(array, day, hours);

printarr2(array, day, hours);

maximum2(array, day, hours);

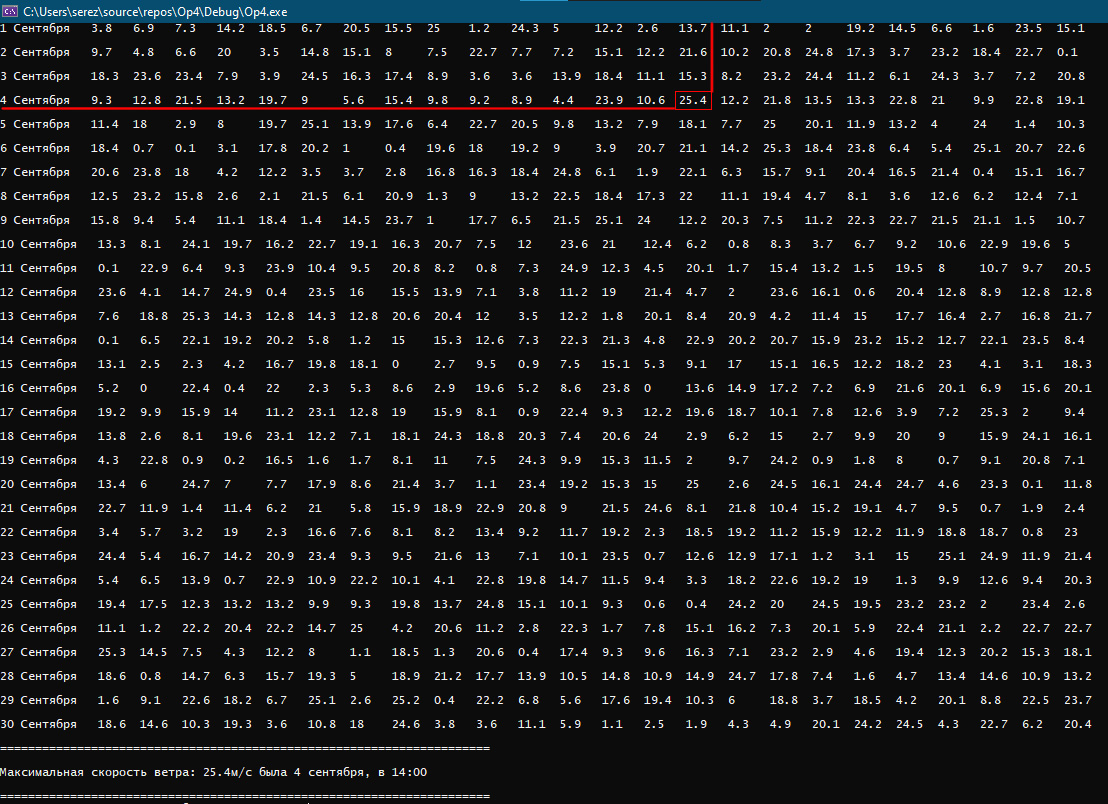
delete [] array;

return 0;

}

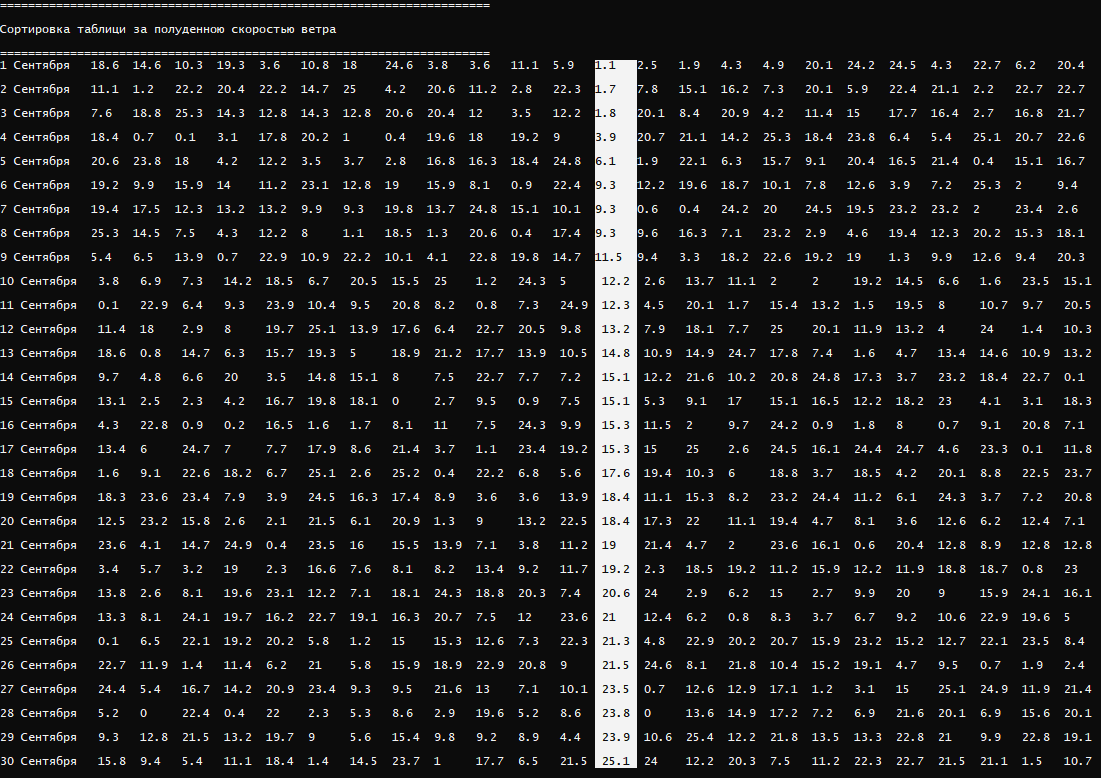
9. Результати тестувань

1) Результат пошуку найбільшої швидкості вітру



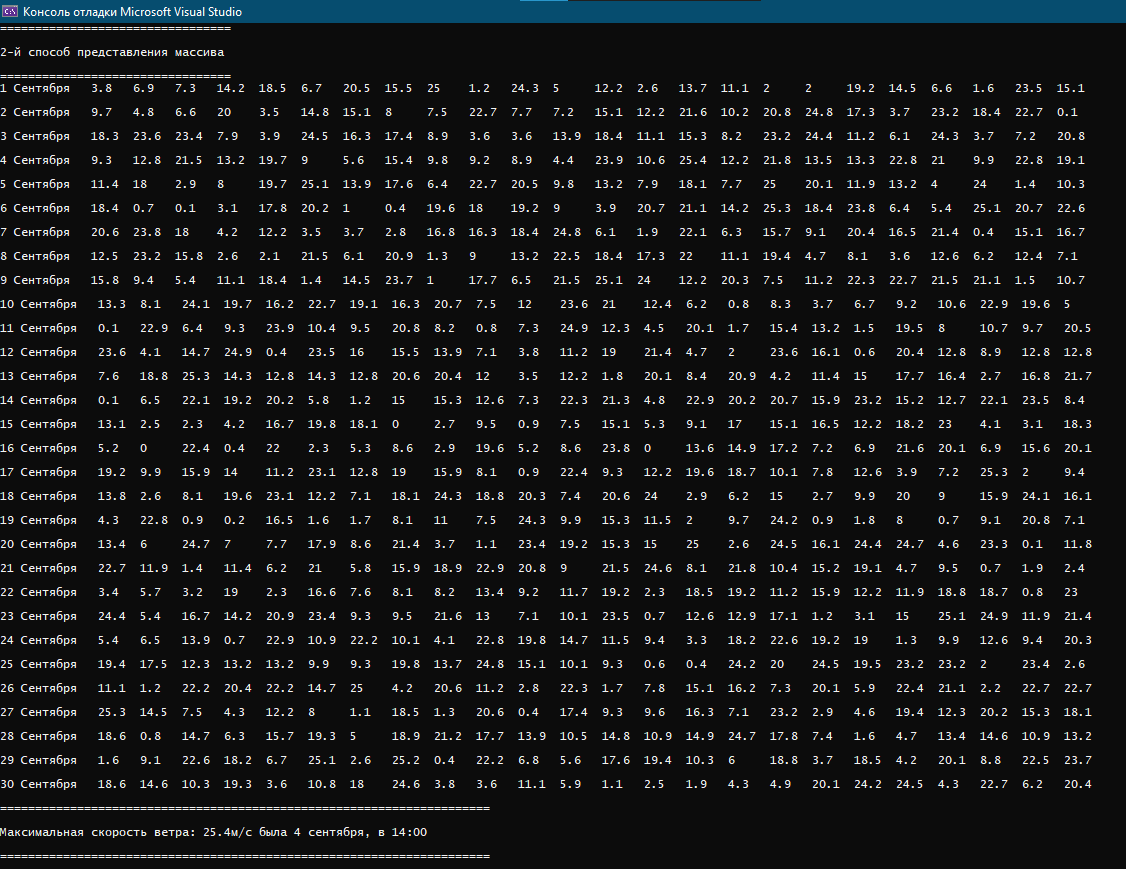
Як видно на скріні, найбільша швидкість, день та час були знайдені вірно.

2) Результат сортування массиву



Як бачимо, числа у виділеному стовпчику розміщені по зростанню, отже масив сортується правильно.

3) Резульат виводу звичайного одновимірного массиву у вигляді матриці



10. Висновок

Під час виконнаня лабораторної роботи працював з динамічною пам'яттю(купою) та динамічними масивами. Для ініціалізації двовимірного масиву спочатку створив вказівник на масив с вказівниками на рядки масиву. Потім доступ до елементів масиву виконувався як і з статичними масивами. На мою думку, динамічний масив більш зручний у використанні ніж статичний, тому що його розмір можна задати вже під час виконання програми, проте слід не забувати про видалення масиву після закінчення роботи алгоритму оператором delete, інакше втрата пам’яті. Для другого методу вирішення задачі використав динамічний одновимірний масив. Та вивів його у вигляді матриці. Також знайшов найбільшу швидкість вітру, сортування масиву не вуконував, адже у двовимірному масиві рядки міняються місцями, а тут є лише один великий рядок.