Міністерство освіти і науки України

Західноукраїнський національний університет

Факультет комп’ютерних інформаційних технологій

#### Кафедра інформаційно-

#### обчислювальних систем

#### і управління

група КНШІ-11

**Лабораторна робота № 3**

З дисципліни “Основи програмування”

На тему:

“Одновимірні та багатовимірні масиви, робота з текстовими рядками”

Виконала

Чапкайло Андріяна

Перевірив

Цапів А. Я.

м. Тернопіль

2024 рік

**Мета роботи:** навчитися складати програми обробки лінійних та двовимірних масивів, реалізовувати найпростіші операції з текстовими рядками.

**Завдання:** оформити кожен пункт завдання вибраного варіанту у вигляді функції. Всі необхідні дані для функцій передаються їм в якості параметрів. Використання глобальних змінних у функціях не допускається.

: **Варіант 9**

В одновимірному масиві, що складається з № дійсних елементів, обчислити:

* максимальний за модулем елемент масиву,
* суму елементів масиву, що розташовані між першим і другим додатними елементами.

Перетворити масив таким чином, щоб всі елементи, рівні нулю та одиниці, розташовувались після всіх інших

**Лістинг коду:**

#include <iostream>

using namespace std;

int main()

{

    const int length = 6;

    int array[length];

    int firstIndex = -1;

    int lastIndex = -1;

    for (int i = 0; i < length; i++)

    {

        int n;

        cout << "Enter " << i << "th element: ";

        cin >> n;

        array[i] = n;

        if (n > 0 && firstIndex == -1) firstIndex = i;

        if (n > 0 && lastIndex == -1) lastIndex = i;

    }

    int maxMod = 0;

    for (int i = 0; i < length; i++)

    {

        if (abs(array[i]) > maxMod) maxMod = abs(array[i]);

    }

    int sum = 0;

    for (int i = firstIndex + 1; i < lastIndex; i++)

    {

        sum += array[i];

    }

    int sortedArray[length];

    int fromStart = 0;

    int fromEnd = length - 1;

    for (int i = 0; i < length; i++)

    {

        if (array[i] == 0 || array[i] == 1)

        {

            sortedArray[fromEnd] = array[i];

            fromEnd--;

        }

        else

        {

            sortedArray[fromStart] = array[i];

            fromStart++;

        }

    }

    cout << "Max: " << maxMod << endl;

    cout << "Sum of elements between first positive and second positive elements: " << sum << endl;

    cout << "Sorted array: [ ";

    for (int i = 0; i < length; i++)

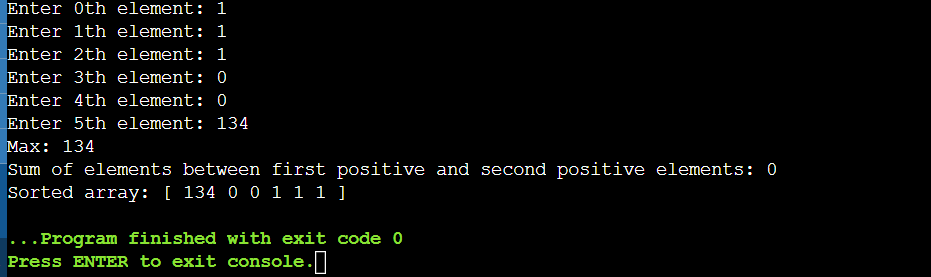
        cout << sortedArray[i] << " ";

    cout << "]";

    return 0;

}

**Скріншот виконання програми:**



Коефіцієнти системи лінійних рівнянь задані у вигляді прямокутної матриці. За допомогою допустимих перетворень звести матрицю до трикутного вигляду. Знайти кількість рядків, середнє арифметичне елементів яких менше заданої величини

**Лістинг код**

#include <iostream>

using namespace std;

int main()

{

    const int rows = 4;

    const int cols = 3;

    int matrix[rows][cols];

    for (int i = 0; i < rows; i++)

    {

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

        {

            cout << "Enter A" << i + 1 << j + 1 << ": ";

            cin >> matrix[i][j];

        }

    }

    int value;

    int rowCount = 0;

    cout << endl;

    cout << "Enter value: ";

    cin >> value;

    for (int i = 0; i < rows; i++)

    {

        int avarage = 0;

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

        {

            avarage += matrix[i][j];

        }

        avarage /= cols;

        if (avarage < value) rowCount++;

    }

    for (int i = 0; i < rows; i++)

    {

        if (matrix[i][i] == 0)

        {

            for (int k = i + 1; k < rows; k++)

            {

                if (matrix[k][i] != 0)

                {

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

                    {

                        int buffer = matrix[j][i];

                        matrix[j][i] = matrix[j][k];

                        matrix[j][k] = buffer;

                    }

                    break;

                }

            }

        }

        int pivot = matrix[i][i];

        if (pivot != 0)

        {

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

            {

                matrix[i][j] /= pivot;

            }

        }

        for (int k = i + 1; k < rows; k++)

        {

            int factor = matrix[k][i];

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

            {

                matrix[k][j] -= factor \* matrix[i][j];

            }

        }

    }

    cout << "Number of rows satysfying the condition: " << rowCount << endl;

    cout << "The matrix is reduced to a triangle form: " << endl;

    for (int i = 0; i < rows; i++)

    {

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

            cout << matrix[i][j] << " ";

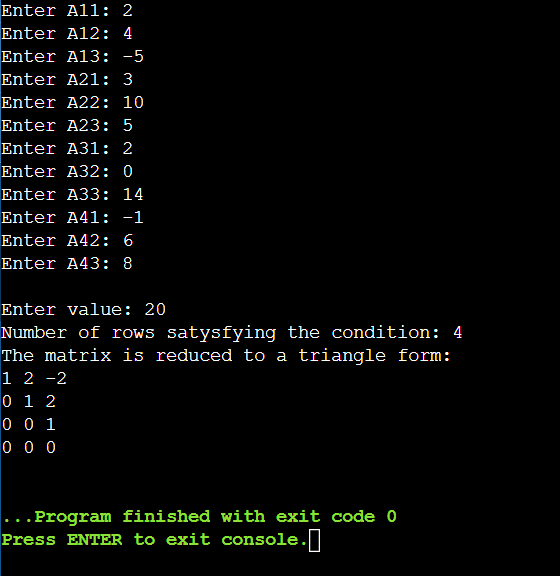
        cout << endl;

    }

    return 0;

}

**Скріншот виконання програми:**

****3 клавіатури вводиться текстовий рядок. Скласти програму, яка підраховує кількість слів у тексті, які закінчуються на голосну літеру, виводить на екран всі слова, довжина яких менша п'яти символів; видаляє всі слова, які містять хоча б одну латинську літеру.

**Лістинг код**

#include <iostream>

#include <string>

using namespace std;

int main()

{

    string text;

    string wordArr[100];

    int wordIndex = 0;

    cout << "Enter text: ";

    getline(cin, text);

    string word = "";

    for (char letter : text)

    {

        if (letter == ' ')

        {

            wordArr[wordIndex] = word;

            wordIndex++;

            word = "";

        }

        else

        {

            word += letter;

        }

    }

    wordArr[wordIndex] = word;

    wordIndex++;

    char vowels[5] = { 'a', 'e', 'i', 'o', 'u' };

    int wordCount = 0;

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

    {

        string word = wordArr[i];

        bool vowel = false;

        for (int i = 0; i < 5; i++)

        {

            if (word[word.length() - 1] == vowels[i])

            {

                vowel = true;

                break;

            }

        }

        if (vowel == true) wordCount++;

    }

    cout << "Number of words which ends at vowel letter: " << wordCount << endl;

    cout << "Words less than 5 characters long: ";

    for (string word : wordArr)

    {

        if (word.length() < 5) cout << word << " ";

    }

    cout << "Words that do not contain at least one latin letter: ";

    for (string word : wordArr)

    {

        for (char letter : word)

        {

            if (isalpha(letter)) break;

            else cout << word << " ";

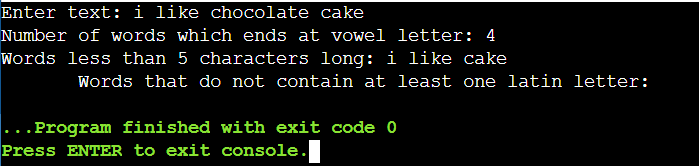
        }

    }

    return 0;

}

**Скріншот виконаного завдання**

****

**Висновок**: У ході виконання роботи я навчилася розробляти програми для обробки лінійних і двовимірних масивів, а також реалізовувати базові операції з текстовими рядками. Я розглянула різні алгоритми обробки масивів, включаючи пошук максимального за модулем елемента, обчислення суми елементів за певних умов та перетворення масиву за специфічними правилами. Це дозволило глибше зрозуміти структури даних, такі як масиви, і навчитися ефективно працювати з ними в рамках програмування на мові C++. Також я отримала базові навички роботи з текстовими рядками, що є необхідною частиною обробки текстових даних у програмуванні.