Кристиян Златинов Александров група:20а факултетен №361222038

Ръководител: доц. д-р инж. Даниела Минковска

Курсов Проект

Специалност : ИСИИ Дисциплина : ПИ

Картина, която съдържа лого

Описанието е генерирано автоматично

**Условие на задачата :**

Да се състави програма за обработка на масива A [N,N], където данните са реални числа в интервала [-1000;1000]. Програмата да извърши следните действия : отпечатване условието на задачата ; отпечатване на имената на автора на програмата ; въвеждане на входните данни ; отпечатване на входните данни ; a) да се образува едномерен масив C, елементите на който са елементите от масива A намиращи се под главния диагонал ; б) получения масив да се сортира по големина; отпечатване на получените резултати след обработка a) и след обработка б) .

Описание на използваните функции в кода :

Създадена от мен програма, която обработва 2D масив A и създава 1D масив C с елементи от A, разположени под главния диагонал. След това сортира елементите на C във възходящ ред и отпечатва сортирания масив.

Ето обясненията стъпка по стъпка как работи този код:

Кодът започва с дефиниране на функция print\_task\_condition(), която просто отпечатва условието на задачата на конзолата. Тази функция се извиква от основната функция за отпечатване на описанието на задачата.

В основната функция програмата иска от потребителя да въведе размера на 2D масива A (N) с помощта на израза std::cin.

След това се създава 2D вектор A с размер NxN с помощта на контейнера std::vector. След това програмата иска от потребителя да въведе елементите на A ред по ред, използвайки вложени цикли for и израза std::cin.

След това програмата отпечатва входния масив A, използвайки вложени цикли for и израза std::cout.

След това програмата създава 1D вектор C за съхраняване на елементите на A, разположени под главния диагонал. Това се прави с помощта на два вложени цикъла for, които обикалят редовете и колоните на A и проверяват дали текущият елемент е под главния диагонал. Ако е така, елементът се добавя към вектора C с помощта на метода push\_back().

След това програмата отпечатва формирания масив C с помощта на for цикъл и командата std::cout.

След това програмата сортира масива C във възходящ ред, използвайки алгоритъма std::sort() от библиотеката <algorithm>.

Накрая програмата отпечатва сортирания масив C с помощта на for цикъл и оператора std::cout.

Програмата завършва като връща 0, което показва успешно изпълнение.

Като цяло тази програма е проста реализация на задача, която включва въвеждане на 2D масив, обработката му за получаване на 1D масив и след това сортиране на 1D масива.

**Програмен Код:**

#include <iostream>

#include <vector>

#include <algorithm>

// Function to print the task condition

void print\_task\_condition() {

std::cout << "Task: Process the 2D array A[N,N] with real numbers in the range [-1000;1000]." << std::endl;

std::cout << "Step a) Form a 1D array C with elements of A located below the main diagonal." << std::endl;

std::cout << "Step b) Sort the resulting array C by size." << std::endl;

}

int main() {

// Print the task condition and the author's name

print\_task\_condition();

// Declare the size of the array (N) and input it

int N;

std::cout << "Enter the size of the array (N): ";

std::cin >> N;

// Create a 2D array A of size NxN and input its elements

std::vector<std::vector<double>> A(N, std::vector<double>(N));

std::cout << "Enter the elements of array A (one row at a time): " << std::endl;

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

std::cin >> A[i][j];

}

}

// Print the input array A

std::cout << "Input array A: " << std::endl;

for (int i = 0; i < N; i++) {

for (int j = 0; j < N; j++) {

std::cout << A[i][j] << " ";

}

std::cout << std::endl;

}

// Create a 1D array C to store elements of A located below the main diagonal

std::vector<double> C;

for (int i = 1; i < N; i++) {

for (int j = 0; j < i; j++) {

C.push\_back(A[i][j]);

}

}

// Print the formed array C

std::cout << "Formed array C: ";

for (double c : C) {

std::cout << c << " ";

}

std::cout << std::endl;

// Sort the array C

std::sort(C.begin(), C.end());

// Print the sorted array C

std::cout << "Sorted array C: ";

for (double c : C) {

std::cout << c << " ";

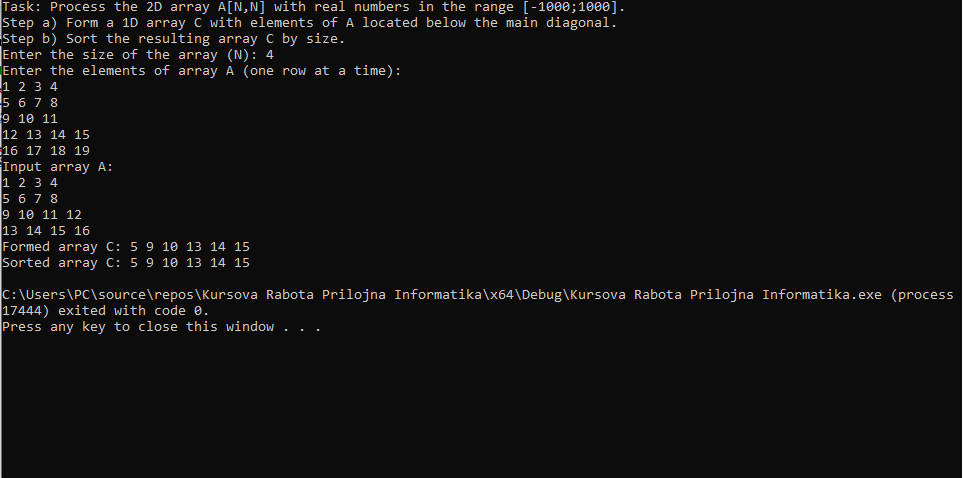
}

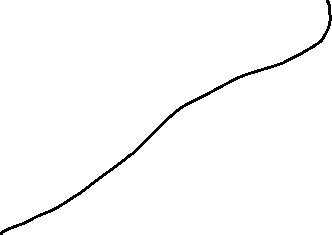
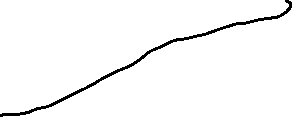
std::cout << std::endl;

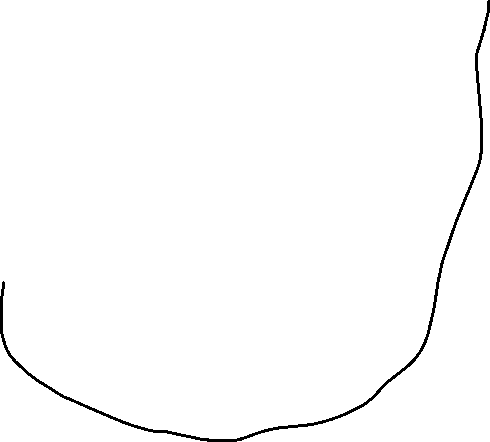
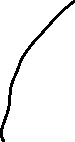
return 0;

}

**Примерно Тестване:**

****

****

****

**Блок Схема:**

****