Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Тихоокеанский государственный университет»

Высшая школа кибернетики и цифровых технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

ПО КУРСУ «ПРОГРАММИРОВАНИЕ»

«РАЗРАБОТКА С ОДНОМЕРНЫМИ И ДВУМЕРНЫМИ МАССИВАМИ»

Выполнил: студент 2 курса группы ПО(б)-31

Корейбо Дмитрий Алексеевич

Проверил: ассистент ВШ КЦТ

Крылов Владимир Андреевич

Хабаровск 2024 г.

Цель работы: изучить теоретический материал по лабораторной работе и реализовать на языке программирования C++ программы для решения заданий.

Задание: необходимо изучить теоретический материал разработке с одномерными и двумерными массивами и реализовать несколько программ на языке программирования C++, чтобы получить практический опыт работы с этими концепциями.

Ход работы:

1. работа с программой “Visual Studio” для созданий программ одномерных и двумерных массивов.

using namespace std – импорт всего пространства имён std.

Include vector в C++ — это подключение заголовочного файла <vector>.

Вектор в языке C++ — это абстрактная модель с рядом определённых функций, позволяющих совершать различные действия, в том числе имитировать работу динамического массива.

Swap в C++ — это функция, которая обменивает значения двух переменных.

Vector в C++ — это стандартный шаблон обобщённого программирования, реализующий динамический массив.

1. составление с помощью данной программы для первого задания (листинг 1).

Листинг 1 — код файла Visual Studio

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "RUS");

const int n = 5;

int a[n][n];

int b = 1;

cout << "Введите элементы матрицы 5x5:" << endl;

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

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

cin >> a[i][j];

}

}

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

b \*= a[i][n - 1 - i];

}

cout << "Произведение элементов побочной диагонали: " << b << endl;

return 0;

}

Результат работы программы первого задания (рисунок 1).

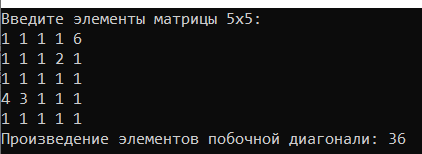


Рисунок 1 – результат работы программы первого задания

1. составление с помощью данной программы для второго задания (листинг 2).

Листинг 2 — код файла Visual Studio

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "RUS");

const int n = 6;

int a[n][n];

cout << "Введите элементы матрицы 6x6:" << endl;

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

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

cin >> a[i][j];

}

}

for (int i = 0; i < n / 2; i++) {

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

swap(a[i][j], a[n - 1 - i][j]);

}

}

cout << "Матрица после замены строк:" << endl;

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

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

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

}

cout << endl;

}

return 0;

}

Результат работы программы второго задания (рисунок 2).

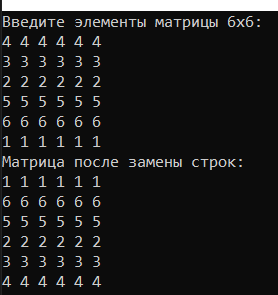


Рисунок 2 – результат работы программы второго задания

1. Составление с помощью данной программы для третьего задания (листинг 3).

Листинг 3 - код файла Visual Studio

#include <iostream>

#include <vector>

using namespace std;

int main() {

setlocale(LC\_ALL, "RUS");

int m, n;

cout << "Введите количество строк (m): ";

cin >> m;

cout << "Введите количество столбцов (n): ";

cin >> n;

vector<vector<int>> a(m, vector<int>(n));

cout << "Введите элементы матрицы:" << endl;

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

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

cin >> a[i][j];

}

}

vector<double> b(m);

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

double sum = 0;

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

sum += a[i][j];

}

b[i] = sum / n;

}

cout << "Массив средних арифметических элементов строк:" << endl;

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

cout << "Среднее для строки " << i + 1 << ": " << b[i] << endl;

}

return 0;

}

Результат работы программы третьего задания (рисунок 3).

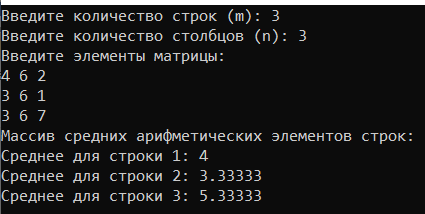


Рисунок 3 – результат работы программы третьего задания

Вывод: в ходе выполнения лабораторной работы мы изучили разработку одномерных и двумерных массивов на языке программирования C++. C++ остается актуальным языком в программировании благодаря своей мощности, гибкости и обширным возможностям. Несмотря на существование более высокоуровневых языков, C++ продолжает использоваться для создания производительных, ресурс затратных приложений, и его изучение может быть очень полезным для понимания основ компьютерной науки и программирования в целом.