Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт: Институт цифрового развития

Кафедра: Межинститутская базовая кафедра

**Отчёт по Лабораторной работе № 2 по информационным технологиям и программированию**

выполнил:

Студент, 1 курс, группа ПИЖ-б-о-23-2

направление подготовки

09.03.04 Программная инженерия,

направленность «Разработка и сопровождение программного обеспечения »

форма обучения очная

Черников Дмитрий Дмитриевич

Ставрополь, 2024г

**Лабораторная работа № 2. Файловые потоки ввода-вывода. Работа с текстовыми и бинарными файлами**

**Цель работы**: Разработка программ языке С++ с использованием файловых потоков ввода- вывода. Научиться разрабатывать программы языке С++ для обработки текстовых и бинарных файлов.

**Вариант** – 9

GitHub: https://github.com/dimain1/information\_technology\_and\_programming/tree/main/Lab%202

Здание 1

Дан файл вещественных чисел a.txt Найти количество нулевых элементов и сумму отрицательных элементов.

Код программы:

#include<iostream>

#include<string>

#include<fstream>

#include<vector>

using namespace std;

vector<float> split(const string& str, const char& ch);

int main()

{

string line;

int zero = 0;

float sumNegative = 0;

ifstream in("a.txt");

if (in.is\_open())

{

while (getline(in, line))

{

for (auto c : split(line, ' '))

{

if (c == 0)

{

zero++;

}

else if (c < 0) {

sumNegative+= c;

}

else {

continue;

}

}

}

in.close();

}

cout << "Zero numbers = " << zero << endl;

cout << "Sum negative numbers = " << sumNegative << endl;

}

vector<float> split(const string& str, const char& ch)

{

string next;

vector<float> result;

for (string::const\_iterator it = str.begin(); it != str.end(); it++) {

if (\*it == ch) {

if (!next.empty()) {

result.push\_back(stof(next));

next.clear();

}

}

else {

next += \*it;

}

}

if (!next.empty())

result.push\_back(stof(next));

return result;

}

Изображение выглядит как текст, Мультимедийное программное обеспечение, программное обеспечение, снимок экрана

Автоматически созданное описание

Здание 2

Дан файл вещественных чисел a.txt или a.txt. Переписать в файл а2.txt ненулевые элементы файла а.txt разделенные на 5.

Код программы:

#include<iostream>

#include<string>

#include<fstream>

#include<vector>

using namespace std;

vector<float> split(const string& str, const char& ch);

int main()

{

string line;

int zero = 0;

float sumNegative = 0;

ifstream in("a.txt");

ofstream out("a2.txt");

if (in.is\_open())

{

while (getline(in, line))

{

if (out.is\_open())

{

for (auto c : split(line, ' '))

{

if (c != 0) {

out << c / 5;

out << " ";

}

}

out.close();

}

}

in.close();

}

}

vector<float> split(const string& str, const char& ch)

{

string next;

vector<float> result;

for (string::const\_iterator it = str.begin(); it != str.end(); it++) {

if (\*it == ch) {

if (!next.empty()) {

result.push\_back(stof(next));

next.clear();

}

}

else {

next += \*it;

}

}

if (!next.empty())

result.push\_back(stof(next));

return result;

} Изображение выглядит как текст, Шрифт, Мультимедийное программное обеспечение, снимок экрана

Автоматически созданное описание

Задание 3

Из заданного входного файла считать символы и записать в один новый файл только большие латинские буквы, в другой новый файл только малые латинские буквы и посчитать количество цифр

Код программы:

#include<iostream>

#include<fstream>

#include<string>

#include<vector>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

string line;

int count = 0;

ifstream in("T3.txt");

ofstream in1("T3Lower.txt");

ofstream in2("T3Upper.txt");

if (in.is\_open()) {

while (getline(in, line)) {

for (auto c : line) {

int code = static\_cast<int>(c);

if (code >= static\_cast<int>('a') && code <= static\_cast<int>('z')) {

if (in1.is\_open()) {

in1 << c;

}

}

else if (code >= static\_cast<int>('A') && code <= static\_cast<int>('Z')) {

if (in2.is\_open()) {

in2 << c;

}

}

else if (code >= static\_cast<int>('0') && code <= static\_cast<int>('9')) {

count++;

}

else {

continue;

}

}

}

}

cout << "Количество цифр в файле = " << count;

}

Изображение выглядит как снимок экрана, программное обеспечение, Мультимедийное программное обеспечение, текст

Автоматически созданное описание

Здание 4

Организовать файл целых чисел. Вычислить среднее значение среди положительных значений файла расположенных на чётных позициях.

Код программы:

#include <iostream>

#include <fstream>

using namespace std;

int main() {

setlocale(LC\_ALL, "ru");

std::string filePath;

cout << "Введите путь до файла ";

cin >> filePath;

std::ifstream file(filePath);

if (file.is\_open()) {

int value;

int Count = 0;

int sum = 0;

bool even = false;

while (file >> value) {

if (even && value > 0) {

sum += value;

Count++;

}

even = !even;

}

file.close();

double result = 0;

if (Count > 0) {

result = (sum) / Count;

}

std::cout << "Среднее значение положительных чисел на четных позициях: " << result << std::endl;

}

return 0;

}

Изображение выглядит как текст, снимок экрана, Шрифт, Мультимедийное программное обеспечение

Автоматически созданное описание

Задание 5. создать текстовый файл с произвольной информацией. Орга- низовать просмотр содержимого файла. Организовать чтение и обработку данных из файла в соответствии с индивидуальным заданием. Сохранить полученные результаты в новый текстовый файл.

«Владелец телефона»: фамилия; имя; отчество; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); № телефона. Вывести данные про владельцев телефона номер, которого начинается на 720.

Код программы:

#include <iostream>

#include <fstream>

#include <sstream>

#include <vector>

using namespace std;

struct Owner {

std::string lastName;

std::string firstName;

std::string middleName;

std::string address;

std::string phoneNumber;

};

std::vector<Owner> getOwners(const std::string& filePath) {

std::vector<Owner> owners;

std::ifstream file(filePath);

if (!file.is\_open()) {

std::cerr << "Unable to open file." << std::endl;

return owners;

}

std::string line;

while (std::getline(file, line)) {

std::istringstream iss(line);

Owner owner;

std::getline(iss, owner.lastName, ';');

std::getline(iss, owner.firstName, ';');

std::getline(iss, owner.middleName, ';');

std::getline(iss, owner.address, ';');

std::getline(iss, owner.phoneNumber, ';');

owners.push\_back(owner);

}

file.close();

return owners;

}

void writeOwners(const std::vector<Owner>& owners, const std::string& outputPath) {

std::ofstream outFile(outputPath);

if (outFile.is\_open()) {

for (const auto& owner : owners) {

if (owner.phoneNumber.find("720") == 0) {

outFile << owner.lastName << ';' << owner.firstName << ';' << owner.middleName << ';'

<< owner.address << ';' << owner.phoneNumber << '\n';

}

}

outFile.close();

}

}

int main() {

setlocale(LC\_ALL, "ru");

std::string input;

cout << "Введите путь до исходнго файла ";

cin >> input;

std::string output;

cout << "Введите путь до результирующего файла ";

cin >> output;

std::vector<Owner> allOwners = getOwners(input);

if (!allOwners.empty()) {

writeOwners(allOwners, output);

}

return 0;

}

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, Шрифт

Автоматически созданное описание

Задание 6: создать текстовый файл в текстовом редакторе. Организовать просмотр содержимого файла и выполнения действий в соответствии с условием индивидуального задания. Обеспечить сохранение всех полученных результатов в новый файл.

Дан символьный файл f. Получить копию файла в файле g.

Код программы:

#include <iostream>

#include <fstream>

#include <string>

using namespace std;

int main() {

std::ifstream inputFile("f.txt", std::ios::binary);

std::ofstream outputFile("g.txt", std::ios::binary);

if (inputFile.is\_open() && outputFile.is\_open()) {

string line;

while (getline(inputFile, line)) {

outputFile << line;

}

inputFile.close();

outputFile.close();

}

return 0;

}

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Задание 7: представленные задания оформить в виде текстового файла.

В файле хранится k матриц размерности n× n. Для каждой матрицы из файла вычислить сумму её диагональных элементов. Все матрицы с нечетными суммами записать в другой файл, заменив их в исходном файле транспонированными матрицами. Вывести на экран содержимое первого и второго файлов.

Код программы:

#include <iostream>

#include <fstream>

#include <vector>

using namespace std;

struct Matrix {

vector<vector<int>> data;

};

vector<Matrix> readMatrices(const string& filePath, int& n) {

ifstream file(filePath);

file >> n;

vector<Matrix> matrices;

int k;

file >> k;

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

Matrix matrix;

matrix.data.resize(n, vector<int>(n));

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

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

file >> matrix.data[row][col];

}

}

matrices.push\_back(matrix);

}

file.close();

return matrices;

}

int calculateDiagonalSum(const Matrix& matrix) {

int sum = 0;

int n = matrix.data.size();

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

sum += matrix.data[i][i];

if (i != n / 2) {

sum += matrix.data[i][n - i - 1];

}

}

return sum;

}

void transposeMatrix(Matrix& matrix) {

int n = matrix.data.size();

vector<vector<int>> transposedData(n, vector<int>(n));

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

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

transposedData[j][i] = matrix.data[i][j];

}

}

matrix.data = transposedData;

}

void writeMatrices(const string& filePath, const vector<Matrix>& matrices) {

ofstream file(filePath, ios::app);

for (const auto& matrix : matrices) {

for (const auto& row : matrix.data) {

for (int element : row) {

file << element << " ";

}

file << endl;

}

file << endl;

}

file.close();

}

int main() {

setlocale(LC\_ALL, "ru");

int n;

vector<Matrix> matrices = readMatrices("T7\_1.txt", n);

if (matrices.empty()) {

return 1;

}

vector<Matrix> transposedMatrices;

for (const auto& matrix : matrices) {

int diagonalSum = calculateDiagonalSum(matrix);

if (diagonalSum % 2 != 0) {

Matrix transposedMatrix = matrix;

transposeMatrix(transposedMatrix);

transposedMatrices.push\_back(transposedMatrix);

}

}

cout << "Содержимое первого файла:" << endl;

for (const auto& matrix : matrices) {

for (const auto& row : matrix.data) {

for (int element : row) {

cout << element << " ";

}

cout << endl;

}

cout << endl;

}

writeMatrices("T7\_2.txt", transposedMatrices);

cout << "Содержимое второго файла:" << endl;

for (const auto& matrix : transposedMatrices) {

for (const auto& row : matrix.data) {

for (int element : row) {

cout << element << " ";

}

cout << endl;

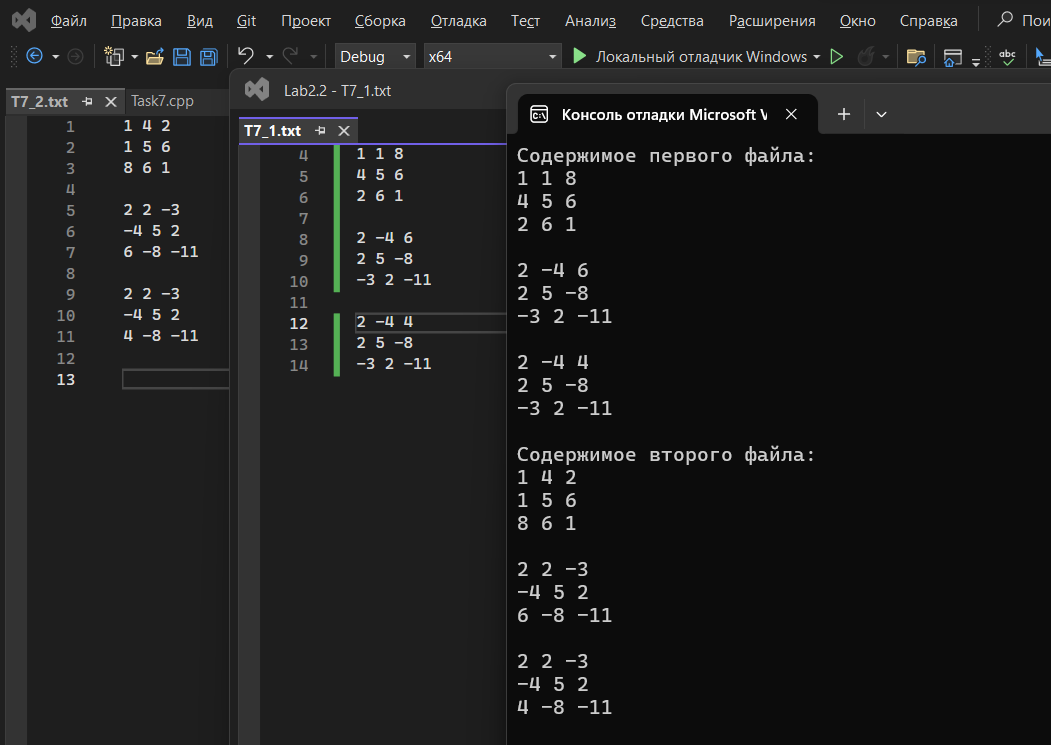
}

cout << endl;

}

return 0;

}



Задание 9: представленные задания оформить в виде бинарного файла

Ввести в Memo или в StringGrid некоторое количество целых чисел и записать их в бинарный файл. Переписать бинарный файл так, чтобы каждое значение, записанное в файле, последовательно от числа к числу было увеличено в два раза. Распечатать числа до и после изменения данных в файле.

Код программы:

#include <iostream>

#include <fstream>

#include <vector>

using namespace std;

void writeToFile(const string& fileName, const vector<int>& numbers) {

ofstream file(fileName, ios::binary);

for (int number : numbers) {

file.write((const char\*)(&number), sizeof(int));

}

file.close();

}

void modifyFile(const string& fileName) {

ifstream fileIn(fileName, ios::binary);

ofstream fileOut("modified\_" + fileName, ios::binary | ios::ate);

int number;

vector<int> modifiedNumbers;

cout << "Числа до изменения:" << endl;

while (fileIn.read((char\*)(&number), sizeof(int))) {

cout << number << endl;

modifiedNumbers.push\_back(number \* 2);

}

cout << "Числа после изменения:" << endl;

for (int modifiedNumber : modifiedNumbers) {

cout << modifiedNumber << endl;

fileOut.write((const char\*)(&modifiedNumber), sizeof(int));

}

fileIn.close();

fileOut.close();

}

int main() {

setlocale(LC\_ALL, "ru");

vector<int> numbers;

int num;

cout << "Введите целые числа (для завершения введите 0):" << endl;

do {

cin >> num;

numbers.push\_back(num);

} while (num != 0);

numbers.pop\_back();

writeToFile("T9\_1.bin", numbers);

modifyFile("T9\_1.bin");

return 0;

}

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Задание 10: Представленные задания оформить в виде бинарного файла.

Создать бинарный файл, компонентами которого является структура, содержащая следующие поля: − фамилия и инициалы клиентов кабельной сети; − стоимость оплаты услуг кабельной сети за месяц; − количество месяцев, за которые заплатил клиент; − количество месяцев, за которые заплатил клиент вперед. Переписать бинарный файл так, чтобы стоимость услуг кабельной сети была бы снижена на 7%, если клиент хочет оплатить вперед за 5 месяцев.

Код программы:

#include <iostream>

#include <fstream>

#include <cstring>

using namespace std;

struct ClientInfo {

char name[50];

float monthlyPayment;

int paidMonths;

int monthsAhead;

};

void createBinaryFile(const string& fileName, const ClientInfo clients[], int numClients) {

ofstream fileOut(fileName, ios::binary);

fileOut.write((char\*)&numClients, sizeof(int));

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

fileOut.write((char\*)&clients[i], sizeof(ClientInfo));

}

fileOut.close();

}

void modifyFile(const string& fileName) {

ifstream fileIn(fileName, ios::binary);

int numClients;

fileIn.read((char\*)&numClients, sizeof(int));

ofstream fileOut("modified\_" + fileName, ios::binary);

fileOut.write((char\*)&numClients, sizeof(int));

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

ClientInfo client;

fileIn.read((char\*)&client, sizeof(ClientInfo));

if (client.monthsAhead == 5) {

client.monthlyPayment \*= 0.93;

}

fileOut.write((char\*)&client, sizeof(ClientInfo));

}

fileIn.close();

fileOut.close();

}

int main() {

ClientInfo clients[] = {

{"Иванов И.И.", 50.0, 3, 2},

{"Петров П.П.", 60.0, 6, 0},

};

int numClients = sizeof(clients) / sizeof(ClientInfo);

createBinaryFile("T10.bin", clients, numClients);

modifyFile("T10.bin");

return 0;

}