Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа № 2

По дисциплине «Основы алгоритмизации и программирования»

На тему «Работа с файлами на языке C»

Выполнил:

Кулешов Артём

Студент 1 курса 8 группы

Преподаватель: асс. Андронова М.В.

Минск, 2024

5. В соответствии со своим вариантом разработать программы для условий, приведенных в таблице ниже, и изучить способы работы с файлами на ***языке С***.

*Задание №1 (основное)*

1. Даны два файла целых чисел, содержащие элементы квадратных матриц с именами **A** и **B** по строкам, причем начальный элемент каждого файла содержит количество столбцов соответствующей матрицы. Создать файл той же структуры с именем **C**, содержащий произведение матриц **А** и **В**.

#include <stdio.h>

#include <iostream>

using namespace std;

int main()

{

FILE\* f, \* f2, \* f3;

fopen\_s(&f, "001.txt", "r"); //открытие первого файла для чтения//

if (f == NULL) { //проверка файла при открытии//

printf("error of opening first file\n");

return -1;

}

fopen\_s(&f2, "002.txt", "r"); //открытие второго файла для чтения//

if (f2 == NULL) { //проверка файла при открытии//

printf("error of opening second file\n");

return -1;

}

int columns, check;

fscanf\_s(f, "%d", &columns); //запись первых чисел в переменные//

fscanf\_s(f2, "%d", &check);

if (check != columns) { //проверка для умножения матриц//

printf("error of multiplication");

return -1;

}

rewind(f); //возвращение указателя//

rewind(f2);

int matrix1[100][100];

int matrix2[100][100];

for (int i = 0; i < columns; i++) { //запись первой матрицы//

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

fscanf\_s(f, "%d", &matrix1[i][j]);

}

}

for (int i = 0; i < columns; i++) { //запись второй матрицы//

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

fscanf\_s(f2, "%d", &matrix2[i][j]);

}

}

fclose(f); //закрытие файлов//

fclose(f2);

int matrix3[100][100];

matrix3[0][0] = 0;

for (int i = 0; i < columns; i++) { //вычисление произведения матриц//

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

matrix3[i][j] = matrix1[i][j] \* matrix1[i][j];

}

}

fopen\_s(&f3, "003.txt", "w"); //открытие третьего файла для записи//

if (f3 == NULL) { //проверка файла при открытии//

printf("error of opening last file\n");

return -1;

}

for (int i = 0; i < columns; i++) { //запись произведения матриц//

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

fprintf(f3, "%d", matrix3[i][j]);

fprintf(f3, " ");

}

fprintf(f3, "%c", '\n');

}

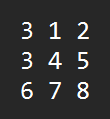
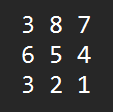
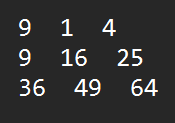
fclose(f3); //закрытие третьего файла//

printf("program run success");

return 0;

}



*Задание №2 (основное)*

2. Создать текстовый файл **F1** не менее, чем из 6 строк, и записать в него информацию. Скопировать в файл **F2** только те строки из **F1**, которые начинаются с буквы «**А**».

#include <stdio.h>

#include <iostream>

using namespace std;

int main() {

FILE\* f, \* f2;

fopen\_s(&f, "f.txt", "r"); //открытие первого файла для чтения//

if (f == NULL) { //проверка файла при открытиии//

printf("error of opening first file");

return -1;

}

fopen\_s(&f2, "f2.txt", "w"); //открытие первого файла для записи//

if (f2 == NULL) { //проверка файла при открытии//

printf("error of opening second file");

return -1;

}

char str[100];

while (fgets(str, sizeof(str), f) != NULL) { //чтение файла построчно//

if (str[0] == 'a') {

fprintf(f2, "%s", str); //запись во второй файл//

}

}

fclose(f); //закрытие файлов//

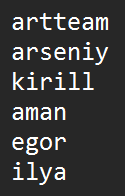
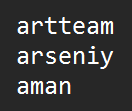
fclose(f2);

printf("program run success\n");

return 0;

}



Дополнительные задания

*Вариант 4 Задание №1*

1. Компоненты файла **fileA** – целые отличные от нуля положительные и отрицательные числа. Получить файл **fileB**, состоящий из положительных чисел.

#include <stdio.h>

#include <iostream>

int main()

{

FILE\* f, \* f2;

fopen\_s(&f, "f.txt", "r"); //открытие первого файла для чтения//

if (f == NULL) { //проверка открытия файла//

printf("error of opening first file\n");

return -1;

}

fopen\_s(&f2, "f2.txt", "w"); //открытие второго файла для записи//

if (f2 == NULL) //проверка открытия файла//

{

printf("error of opening second file\n");

fclose(f); //закрытие файла//

}

int num;

while (fscanf\_s(f, "%d", &num) != EOF) //чтение файла до его конца//

{

if (num > 0) { //проверка является ли число положительным//

fprintf(f2, "%d ", num); //запись во второй файл//

}

}

printf("program run success\n");

fclose(f); //закрытие файлов//

fclose(f2);

}





*Задание №2*

2. Компоненты файла **f** – целые двухзначные числа. Получить файл **g**, образованный из **f** включением только чисел больших некоторого числа, вводимого с клавиатуры.

#include <stdio.h>

#include <iostream>

using namespace std;

int main()

{

FILE\* f, \* g;

fopen\_s(&f, "f.txt", "r"); //открытие первого файла для чтения//

if (f == NULL) { //проверка открытия файла//

printf("error of opening first file\n");

return -1;

}

fopen\_s(&g, "g.txt", "w"); //открытие файла для записи//

if (g == NULL) {

printf("error of creating second file\n");

fclose(f); //закрытие первого файла//

}

int inputNum, num;

cout << "enter num: "; cin >> inputNum;

while (fscanf\_s(f, "%d", &num) != EOF) //чтение файла до его конца//

{

if (num > inputNum) {

fprintf(g, "%d ", num); //запись во второй файл//

}

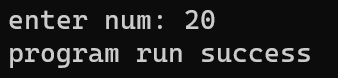
}

printf("program run success\n");

fclose(f); //закрытие файлов//

fclose(g);

}







*Вариант 9 Задание №1*

1. Компоненты файла **fileA** –  целые числа, значения которых повторяются.  Получить файл **fileB**, образованный из **fileA** исключением повторных вхождений одного и того же числа.

#include <iostream>

#include <stdio.h>

using namespace std;

int main() {

FILE\* fileA, \* fileB;

fopen\_s(&fileA, "fileA.txt", "r");

if (fileA == NULL) {

printf("error of opening first file\n");

return -1;

}

fopen\_s(&fileB, "fileB.txt", "w");

if (fileB == NULL) {

printf("error of opening second file\n");

return -1;

}

int num, unique\_num = 0;

int unique\_numbers[100];

while (fscanf\_s(fileA, "%d", &num) != EOF) {

int counter = 0;

for (int i = 0; i < unique\_num; i++) {

if (unique\_numbers[i] == num) {

counter = 1;

break;

}

}

if (counter == 0) {

unique\_numbers[unique\_num] = num;

unique\_num++;

}

}

for (int i = 0; i < unique\_num; i++) {

fprintf\_s(fileB, "%d", unique\_numbers[i]);

}

fclose(fileA);

fclose(fileB);

printf("program run success\n");

return 0;

}



*Задание №2*

2. Создать текстовый файл **F1** не менее, чем из 4 строк, и записать в него информацию. Скопировать из файла **F1** в файл **F2** строки, количество символов в которых больше, чем заданное число, которое вводится с клавиатуры.

#include <iostream>

#include <stdio.h>

using namespace std;

int main() {

FILE\* F1, \* F2;

fopen\_s(&F1, "F1.txt", "r");

if (F1 == NULL) {

printf("error of opening first file\n");

return -1;

}

fopen\_s(&F2, "F2.txt", "w");

if (F2 == NULL) {

printf("error of opening second file\n");

}

char string[100];

int number;

printf("enter num: ");

scanf\_s("%d", &number);

while (fgets(string, sizeof(string), F1) != NULL) {

if (strlen(string) > number) {

fprintf\_s(F2, "%s", string);

}

}

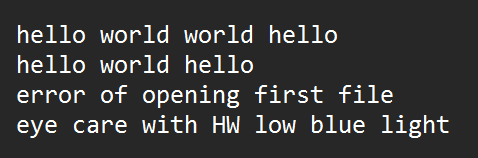
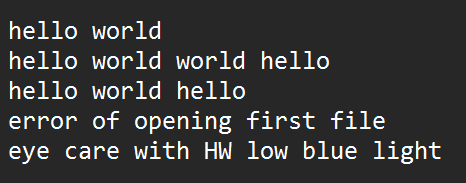
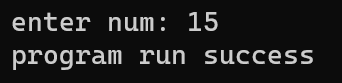
fclose(F1);

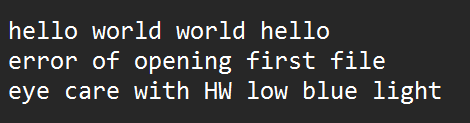
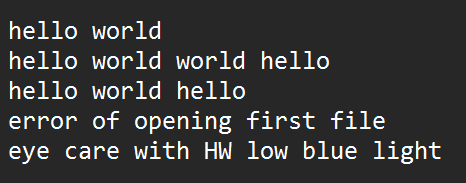
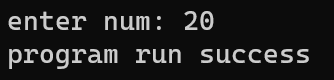
fclose(F2);

printf("program run success\n");

return 0;

}





*Вариант 12 Задание №1*

1. Дан файл вещественных чисел, содержащий элементы квадратной матрицы по строкам, причем начальный элемент файла содержит значение количества столбцов матрицы. Создать новый файл, содержащий **k**-ую строку исходной матрицы.

#include<stdio.h>

#include<string>

#include<iostream>

using namespace std;

void main()

{

int k;

string gg;

char row[50];

FILE\* f, \* f2;

fopen\_s(&f, "f.txt", "r"); //открытие файла только для чтения//

cout << "num of srt: ";

cin >> k;

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

{

fgets(row, 50, f); //пропуск строк до нужной//

}

fgets(row, 50, f);

fopen\_s(&f2, "f2.txt", "w"); //открытие файла только для записи//

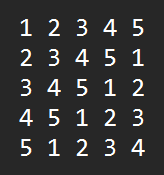
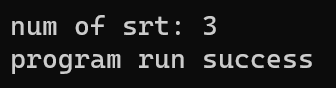
fprintf(f2, "%s", row);

fclose(f2);

fclose(f);

printf("program run success\n");

}



*Задание №2*

2. Даны три файла целых чисел одинакового размера с именами **NameA**, **NameB** и **NameC**. Создать новый файл с именем **NameD**, в который записать минимальные элементы исходных файлов с одним и тем же номером: **min(A0, B0, C0)**, **min( A1, B1, C1)**, **min(A2, B2, C2), ...**

**#include<iostream>**

#include<stdio.h>

usingnamespacestd;

voidmain()

{

#include<stdio.h>

setlocale(LC\_ALL, "rus");

int A[100], B[100], C[100], i, j, size, count, num, min;

charrow[50];

FILE\* file1, \* file2, \* file3, \* file4;

fopen\_s(&file1, "NameA.bin", "r");

fopen\_s(&file2, "NameB.bin", "r");

fopen\_s(&file3, "NameC.bin", "r");

fseek(file1, 0L, SEEK\_END);

size = ftell(file1);

fseek(file1, 0L, SEEK\_SET);

fgets(row, size, file1);

count = 0;

for (i = 0; i <size; i++)

{

if (row[i] == ' ') { count += 1; }

}

count += 1;

cout<<count<<endl;

fseek(file1, 0L, SEEK\_SET);

for (i = 0; i <count; i++)

{

fscanf\_s(file1, "%d", &num);

A[i] = num;

}

for (i = 0; i <count; i++)

{

fscanf\_s(file2, "%d", &num);

B[i] = num;

}

for (i = 0; i <count; i++)

{

fscanf\_s(file3, "%d", &num);

C[i] = num;

}

fclose(file1);

fclose(file2);

fclose(file3);

fopen\_s(&file4, "NameD.bin", "w+");

for (i = 0; i <count; i++)

{

min = A[i];

if (B[i] <min) min = B[i];

if (C[i] <min) min = C[i];

fprintf(file4, "%d ", min);

}

fclose(file4);

}

Входные данные:







Итог:

