Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 18

По дисциплине «Основы алгоритмизации и программирования»

На тему «Работа с файлами на языке С»

Выполнил:

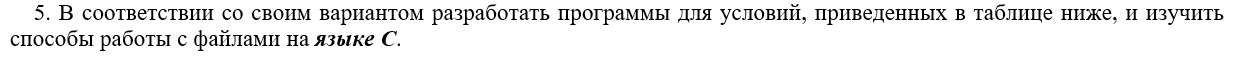
Студент 1 курса 6 группы

Романов Игорь Вячеславович

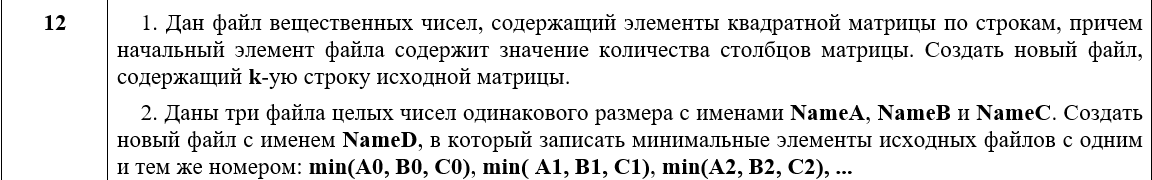
Преподаватель: асс. Андронова М.В.

2023, Минск

Задание 5



Вариант 12 (основной)



1.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>+

int main() {

setlocale(LC\_ALL, "Russian");//локализация консоли

FILE\* inputFile, \*outputFile; //создание исходного и выходных файлов

int nCols, rowNumber, i, j; //количество столбцов, вводимая строка; переменные итераторы

errno\_t err; //создание переменной, определяющей ошибку открытия файла

err = fopen\_s(&inputFile, "input.txt", "r"); //открытие файла в режиме чтения

if (err != 0 || inputFile == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

fscanf\_s(inputFile, "%d", &nCols); //чтение числа столбцов из файла

printf("Введите номер строки для извлечения: "); //ввод номера строки, которую нужно извлечь

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

if (rowNumber <= 0 || rowNumber > nCols) { //проверка валидности номера строки

printf("Некорректный номер строки.\n");

fclose(inputFile);

return -1;

}

err = fopen\_s(&outputFile, "output.txt", "w"); //создание пустого файла для записи строки

if (err != 0 || outputFile == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(inputFile);

return -1;

}

//перемещение указателя файла на нужную строку

for (i = 1; i < rowNumber; i++) {

for (j = 0; j < nCols; j++) {

float temp;

fscanf\_s(inputFile, "%f", &temp);

}

}

//чтение и запись строки в новый файл

for (i = 0; i < nCols; i++) {

float element;

fscanf\_s(inputFile, "%f", &element);

fprintf(outputFile, "%.2f ", element); //запись элемента в файл с двумя знаками после запятой

}

printf("Строка %d записана в файл 'output.txt'\n", rowNumber); //вывод сообщений об успешной записи строки

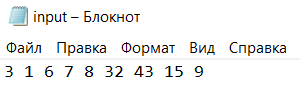
// Закрытие файлов

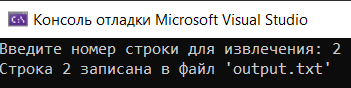
fclose(inputFile);

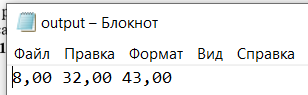
fclose(outputFile);

return 0;

}







2.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>+

//функция для нахождения минимального элемента

int findMin(int numA, int numB, int numC) {

int i\_min;

if (numA < numB) {

if (numA < numC) {

i\_min = numA;

}

else {

i\_min = numC;

}

}

else {

if (numB < numC) {

i\_min = numB;

}

else {

i\_min = numC;

}

}

return i\_min;

}

//главная функция

int main() {

setlocale(LC\_ALL, "Russian");//локализация консоли

FILE \*NameA, \*NameB, \* NameC, \* NameD; //создание исходных и выходного файлов

errno\_t errA, errB, errC, errD; //создание переменных, определяющих ошибку открытия файла

//открытие файлов в режиме чтения

errA = fopen\_s(&NameA, "NameA.txt", "r");

errB = fopen\_s(&NameB, "NameB.txt", "r");

errC = fopen\_s(&NameC, "NameC.txt", "r");

if (errA != 0 || NameA == NULL || errB != 0 || NameB == NULL || errC != 0 || NameC == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

//создание пустого файла NameD

errD = fopen\_s(&NameD, "NameD.txt", "w"); //создание пустого файла для записи строки

if (errD != 0 || NameD == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(NameA);

fclose(NameB);

fclose(NameC);

return -1;

}

//нахождения минимального элемента

int numA, numB, numC; //ввод чисел соответствующих файлов

//пока строки файлов не закончатся

while (fscanf\_s(NameA, "%d", &numA) != EOF && fscanf\_s(NameB, "%d", &numB) && fscanf\_s(NameC, "%d", &numC)) {

int i\_min = findMin(numA, numB, numC); //нахождение минимального элемента

fprintf(NameD, "%d ", i\_min); //запись минимального элемента в файл NameD

}

// Закрытие файлов

fclose(NameA);

fclose(NameB);

fclose(NameC);

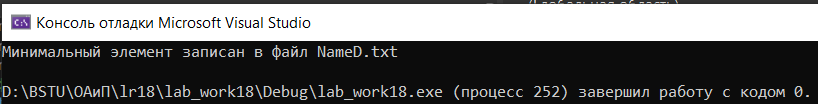
fclose(NameD);

//вывод сообщения об успешной записи элементов в файл

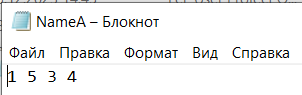
printf("Минимальный элемент записан в файл NameD.txt\n");

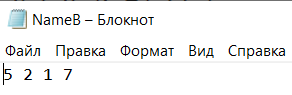
return 0;

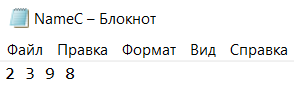
}



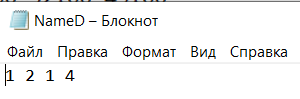
Входные файлы:





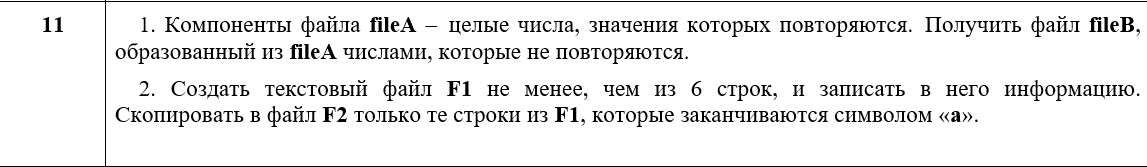


Выходной файл:



Доп. задания

Вариант 11



1.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

//главная функция

int main() {

setlocale(LC\_ALL, "Russian");

FILE \*fileA, \*fileB; //создание исходного и выходного файлов

errno\_t errA, errB; //создание переменных, определяющих ошибку открытия файла

//открытие файлов в режиме чтения

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

if (errA != 0 || fileA == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

int i, unique\_counter = 0, number; //переменная итератора, счетчик неповторяющихся чисел, число из файла

int unique\_numbers[100]; //создание массива для хранения неповторяющихся чисел

while (fscanf\_s(fileA, "%d", &number) != EOF) { //пока не закончитcя файл

int isDublicate = 0; //флажок, определяющий, есть ли в массиве выбранная переменная

for (int i = 0; i < unique\_counter; i++) { //проверка вхождения уникальных чисел

if (unique\_numbers[i] == number) { //если найдено повторное вхождение

isDublicate = 1; //то установить флажку значение истина

break; //выйти из подцикла

}

}

if (!isDublicate) { //добавить уникальное число в массив

unique\_numbers[unique\_counter] = number;

unique\_counter++;

}

}

//создание пустого файла fileB

errB = fopen\_s(&fileB, "fileB.txt", "w"); //создание пустого файла для записи строки

if (errB != 0 || fileB == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(fileA);

return -1;

}

for (int i = 0; i < unique\_counter; i++) { //запись уникальных элементов в файл fileB

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

}

//закрытие файлов

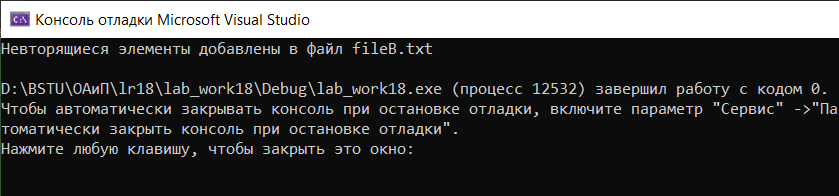
fclose(fileB);

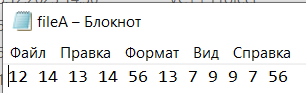
//вывод сообщения об успешной записи элементов в файл

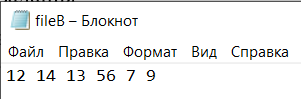
printf("Неповторяющиеся элементы добавлены в файл fileB.txt\n");

return 0;

}







2.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

//главная функция

int main() {

setlocale(LC\_ALL, "Russian");

FILE \*F1, \*F2; //создание исходного и выходного файлов

errno\_t err1, err2; //создание переменных, определяющих ошибку открытия файла

//открытие файлов в режиме чтения

err1 = fopen\_s(&F1, "F1.txt", "r");

if (err1 != 0 || F1 == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

//создание пустого файла F2

err2 = fopen\_s(&F2, "F2.txt", "w"); //создание пустого файла для записи строки

if (err2 != 0 || F2 == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(F1);

return -1;

}

char str[100]; //массив для хранения символьной строки

while (fgets(str, sizeof(str), F1) != NULL) { //чтение из файла одной строки текста полностью

if (str[strlen(str) - 2] == 'a' || str[strlen(str) - 1] == 'a') { //если строка заканчивается на a

fputs(str, F2); //то записать ее в файл

}

}

//закрытие файлов

fclose(F1);

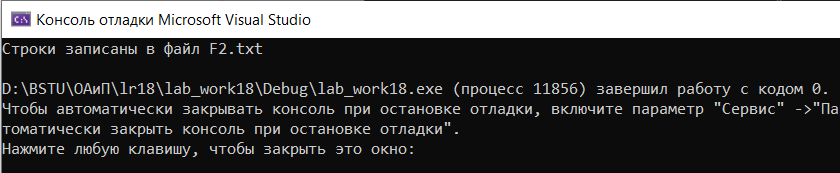
fclose(F2);

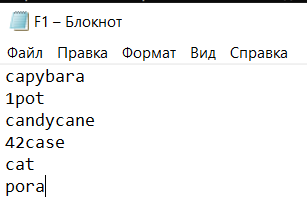
//вывод сообщения об успешной записи элементов в файл

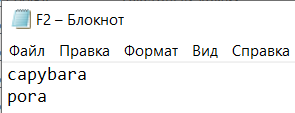
printf("Строки записаны в файл F2.txt\n");

return 0;

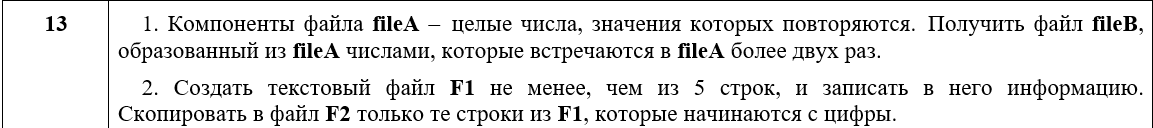
}







Вариант 13



1.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

//главная функция

int main() {

setlocale(LC\_ALL, "Russian");

FILE \*fileA, \*fileB; //создание исходного и выходного файлов

errno\_t errA, errB; //создание переменных, определяющих ошибку открытия файла

//открытие файлов в режиме чтения

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

if (errA != 0 || fileA == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

//создание пустого файла fileB

errB = fopen\_s(&fileB, "fileB.txt", "w"); //создание пустого файла для записи строки

if (errB != 0 || fileB == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(fileA);

return -1;

}

int i, j, index = 0, num; //переменные итератора; индексы чисел массива; число строки

int numbers[100]; //создание массива для хранения чисел из файла

while (fscanf\_s(fileA, "%d", &num) != EOF) { //пока не закончитcя файл

numbers[index++] = num;

}

for (i = 0; i < index; i++) {

int num\_counter = 0; //счетчик переменной

int current = numbers[i]; //текущее число

for (j = 0; j < index; j++) { //если текущее число повторяется, то увеличить счетчик

if (numbers[j] == current) {

num\_counter++;

}

}

if (num\_counter > 2) { //если счетчик больше 2, то добавить число в файл

fprintf(fileB, "%d ", current);

}

}

//закрытие файлов

fclose(fileA);

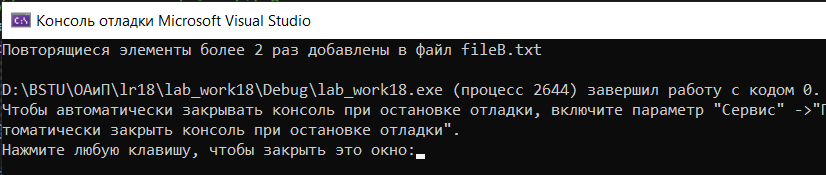
fclose(fileB);

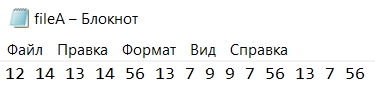
//вывод сообщения об успешной записи элементов в файл

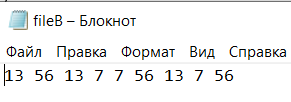
printf("Повторяющиеся элементы более 2 раз добавлены в файл fileB.txt\n");

return 0;

}







2.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

//главная функция

int main() {

setlocale(LC\_ALL, "Russian");

FILE \*F1, \*F2; //создание исходного и выходного файлов

errno\_t err1, err2; //создание переменных, определяющих ошибку открытия файла

//открытие файлов в режиме чтения

err1 = fopen\_s(&F1, "F1.txt", "r");

if (err1 != 0 || F1 == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

//создание пустого файла F2

err2 = fopen\_s(&F2, "F2.txt", "w"); //создание пустого файла для записи строки

if (err2 != 0 || F2 == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(F1);

return -1;

}

char str[100]; //массив для хранения символьной строки

while (fgets(str, sizeof(str), F1) != NULL) { //чтение из файла одной строки текста полностью

if (\*str > '0' && \*str < '9') { //если первый символ строки цифра

fprintf(F2, "%s", str); //то записать ее в файл

}

}

//закрытие файлов

fclose(F1);

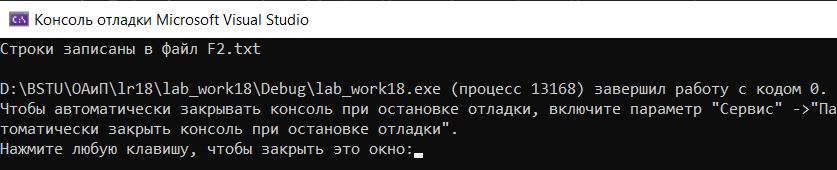
fclose(F2);

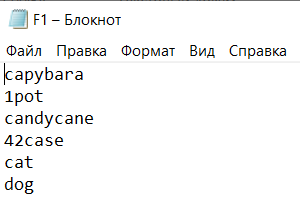
//вывод сообщения об успешной записи элементов в файл

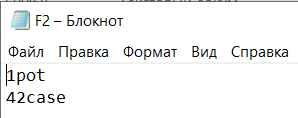
printf("Строки записаны в файл F2.txt\n");

return 0;

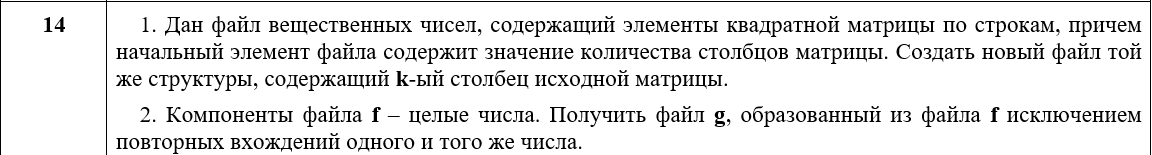
}







Вариант 14



1.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>+

int main() {

setlocale(LC\_ALL, "Russian");//локализация консоли

FILE\* inputFile, \* outputFile; //создание исходного и выходных файлов

int nCols, k, i, j; //количество столбцов, вводимая строка; переменные итераторы

errno\_t err1, err2; //создание переменной, определяющей ошибку открытия файла

err1 = fopen\_s(&inputFile, "input.txt", "r"); //открытие файла в режиме чтения

if (err1 != 0 || inputFile == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

fscanf\_s(inputFile, "%d", &nCols); //чтение числа столбцов из файла

printf("Введите номер столбца для извлечения: "); //ввод номера столбца, которую нужно извлечь

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

if (k <= 0 || k > nCols) { //проверка валидности номера столбца

printf("Некорректный номер строки.\n");

fclose(inputFile);

return -1;

}

err2 = fopen\_s(&outputFile, "output.txt", "w"); //создание пустого файла для записи столбца

if (err2 != 0 || outputFile == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(inputFile);

return -1;

}

float arr[100][100]{}; //создание двумерного массива

for (i = 0; i < nCols; i++) {

for (j = 0; j < nCols; j++) {

fscanf\_s(inputFile, "%f", &arr[i][j]); //запись чисел в массива

}

}

//чтение и запись строки в новый файл

for (i = 0; i < nCols; i++) {

fprintf(outputFile, "%.2f\n", arr[i][k - 1]); //запись элемента в файл с двумя знаками после запятой

}

printf("Столбец %d записан в файл 'output.txt'\n", k); //вывод сообщений об успешной записи столбца

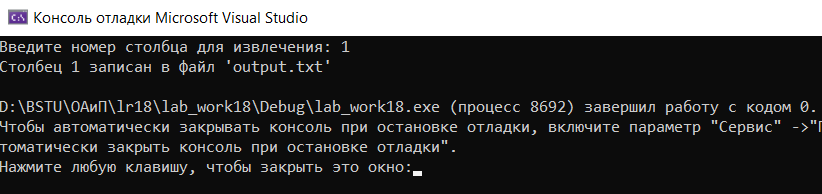
//закрытие файлов

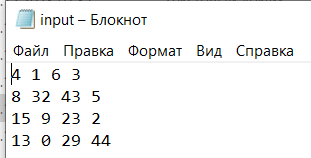
fclose(inputFile);

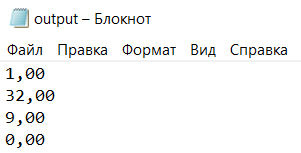
fclose(outputFile);

return 0;

}







2.

#include <stdio.h>

#include <stdlib.h>

#include <iostream>

//главная функция

int main() {

setlocale(LC\_ALL, "Russian");

FILE\* f, \* g; //создание исходного и выходного файлов

errno\_t errf, errg; //создание переменных, определяющих ошибку открытия файла

//открытие файлов в режиме чтения

errf = fopen\_s(&f, "f.txt", "r");

if (errf != 0 || f == NULL) { //если не удалось открыть файл

printf("Не удалось открыть файл с входными данными.\n"); //то вывести соответствующее сообщение

return -1;

}

int i, unique\_counter = 0, number; //переменная итератора, счетчик неповторяющихся чисел, число из файла

int unique\_numbers[100]; //создание массива для хранения неповторяющихся чисел

while (fscanf\_s(f, "%d", &number) != EOF) { //пока не закончится файл

int isDublicate = 0; //флажок, определяющий, есть ли в массиве выбранная переменная

for (int i = 0; i < unique\_counter; i++) { //проверка вхождения уникальных чисел

if (unique\_numbers[i] == number) { //если найдено повторное вхождение

isDublicate = 1; //то установить флажку значение истина

break; //выйти из подцикла

}

}

if (!isDublicate) { //добавить уникальное число в массив

unique\_numbers[unique\_counter] = number;

unique\_counter++;

}

}

//создание пустого файла g

errg = fopen\_s(&g, "g.txt", "w"); //создание пустого файла для записи строки

if (errg!= 0 || g == NULL) { //если не удалось открыть файл

printf("Не удалось создать файл для записи.\n"); //то вывести соответствующее сообщение

fclose(f);

return -1;

}

for (int i = 0; i < unique\_counter; i++) { //запись уникальных элементов в файл g

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

}

//закрытие файлов

fclose(g);

//вывод сообщения об успешной записи элементов в файл

printf("Неповторяющиеся элементы добавлены в файл g.txt\n");

return 0;

}

