**Белорусский государственный технологический университет**

**Факультет информационных технологий**

**Кафедра ПИ**

Лабораторная работа № 19

По дисциплине «Основы алгоритмизации и программирования»

На тему «Работа с файлами на языке С»

Выполнила:  
Студентка 1 курса, 10 группа  
Кириленко Ангелина Васильевна  
Преподаватель: асс. Харланович А.В

2024 г. Минск

|  |  |
| --- | --- |
| **Задание** | **Выполнение задания** |
| 1. Изучить *форматированный* ввод и вывод при работе с *текстовыми* файлами на языке С, выполнив программу, записанную в данном пункте. Дополнить программу операторами чтения файла и вывода на экран его содержимого. | #include <stdio.h>  int main()  {  setlocale(LC\_ALL, "");  int a; errno\_t err;  FILE\* f;  err = fopen(&f, "a.txt", "w");  if (err != 0)  {  perror("Невозможно создать файл\n");  return EXIT\_FAILURE;  }  for (a = 0; a < 70; a += 3)  {  fprintf(f, "%d, ", a);  }  printf("Данные записаны в файл test.txt\n");  fclose(f);  return 0;  } |
| 2. Изучить *строковый* ввод и вывод при работе с *текстовыми* файлами на языке С, выполнив программу, записанную в данном пункте. Дополнить программу операторами чтения файла и вывода на экран его содержимого.  3. Выполнить программу, записанную в данном пункте, которая осуществляет запись *блока* информации в файл. Дополнить программу операторами чтения из файла и вывода на экран его содержимого. | #include <stdio.h> #include <locale.h>  #include <string.h>  int main() {  setlocale(LC\_ALL, "Russian");  FILE\* mf;  char str[50];  errno\_t err;  printf("Введите текст: ");  fgets(str, sizeof(str), stdin);  str[strcspn(str, "\n")] = '\0';  err = fopen\_s(&mf, "tst.txt", "w");  if (err != 0) {  printf("Ошибка открытия файла для записи\n");  return -1;  }  fputs(str, mf);  fputs("\n", mf);  fclose(mf);  err = fopen\_s(&mf, "tst.txt", "r");  if (err != 0) {  printf("Ошибка открытия файла для чтения\n");  return -1;  }  printf("\nСчитаны строки:\n");  while (fgets(str, sizeof(str), mf) != NULL) {  puts(str);  }  if (feof(mf)) {  printf("\nЧтение файла закончено\n");  }  else {  printf("\nОшибка чтения из файла\n");  fclose(mf);  return -1;  }  if (fclose(mf) == EOF) {  printf("Ошибка закрытия файла\n");  }  return 0;  }  #include <iostream> #include <stdio.h>  int main(void) {  setlocale(LC\_CTYPE, "Russian");  FILE\* fp;  errno\_t err;  const char\* st = "привет";  err = fopen\_s(&fp, "a.bin", "wb");  if (err != 0) {  perror("Ошибка открытия файла a.bin");  return EXIT\_FAILURE;  }  fwrite(st, strlen(st), 1, fp);  printf("Записан элемент\n");  fclose(fp);  err = fopen\_s(&fp, "a.bin", "rb");  if (err != 0) {  perror("Ошибка открытия файла a.bin");  return EXIT\_FAILURE;  }  char buffer[256];  size\_t bytesRead = fread(buffer, 1, sizeof(buffer) - 1, fp);  if (bytesRead > 0) {  buffer[bytesRead] = '\0';  printf("\nСодержимое файла:\n%s\n", buffer);  }  else {  printf("\nФайл пуст или произошла ошибка чтения\n");  }  fclose(fp);  return 0;  } |
| 4. Выполнить программу, записанную в правой части, которая реализует операции *позиционирования* в файле и *блоковый* вывод. Изменить программу так, чтобы выводились на экран подряд все предложения с 1 по **n**.  Задание 5. Вариант 8  **Дополнительные задания**  Вариант 3  Вариант 13 | #include <iostream> #include <cstdio>  #include <cstdlib>  #include <locale.h>  using namespace std;  int main() {  setlocale(LC\_ALL, "Russian");  FILE\* fd;  int number = 0, pr = 0;  long fsize = 0;  char pd;  errno\_t err = fopen\_s(&fd, "D:\\учеба\\ОАИП лр\\лр19+\\b.txt", "r");  if (err != 0 || fd == nullptr) {  perror("ошибка открытия файла b.txt");  return EXIT\_FAILURE;  }  fseek(fd, 0L, SEEK\_END);  fsize = ftell(fd);  fseek(fd, 0L, SEEK\_SET);  printf("Введите номер предложения (от 1 до n): ");  scanf\_s("%d", &number);  if (number < 1) {  printf("Номер предложения должен быть положительным числом.\n");  fclose(fd);  return EXIT\_FAILURE;  }  printf("\nВывод предложений с 1 по %d:\n", number);  int current\_sentence = 0;  long start\_pos = 0;  for (long k = 0; k < fsize; k++) {  fread(&pd, sizeof(char), 1, fd);  if (pd == '.') {  current\_sentence++;  if (current\_sentence <= number) {  fseek(fd, start\_pos, SEEK\_SET);  printf("%d-е предложение: ", current\_sentence);  for (long pos = start\_pos; pos <= k; pos++) {  fread(&pd, sizeof(char), 1, fd);  printf("%c", pd);  }  printf("\n");  start\_pos = k + 1;  fseek(fd, start\_pos, SEEK\_SET);  }  else {  break;  }  }  }  if (number > current\_sentence) {  printf("\nВ файле содержится только %d предложений.\n", current\_sentence);  }  fclose(fd);  return 0;  }    Примечание**:** Не выводятся русские символы, хотя locale.h и setlocale подключено. Если возможно, отправьте ответное сообщение с помощью разобраться с этой проблемой.  #define \_CRT\_SECURE\_NO\_DEPRECATE #include <stdio.h>  #include <stdlib.h>  #include <locale.h>  void range(int\* array, int size) {  int\* temp = (int\*)malloc(size \* sizeof(int));  if (!temp) {  printf("Ошибка выделения памяти\n");  return;  }  int j = 0;  for (int i = 0; i < size; ++i) {  temp[j++] = array[i % 2 == 0 ? i / 2 : size - 1 - i / 2];  }  for (int i = 0; i < size; ++i) {  array[i] = temp[i];  }  free(temp);  }  int main() {  setlocale(LC\_ALL, "Russian");  int N;  printf("Введите размер массива: ");  scanf("%d", &N);  int\* A = (int\*)malloc(N \* sizeof(int));  if (!A) {  printf("Ошибка выделения памяти\n");  return 1;  }  printf("Введите элементы массива:\n");  for (int i = 0; i < N; i++) {  scanf("%d", &A[i]);  }  range(A, N);  printf("Массив после перестановки:\n");  for (int i = 0; i < N; i++) {  printf("%d ", A[i]);  }  printf("\n");  free(A);  return 0;  }    #define \_CRT\_SECURE\_NO\_DEPRECATE #include <stdio.h>  #include <stdlib.h>  #include <locale.h>  void matrix(int\* A, int\* B, int\* C, int n, int m, int p) {  for (int i = 0; i < n; i++) {  for (int j = 0; j < p; j++) {  C[i \* p + j] = 0;  for (int k = 0; k < m; k++) {  C[i \* p + j] += A[i \* m + k] \* B[k \* p + j];  }  }  }  }  int main() {  setlocale(LC\_ALL, "Russian");  FILE\* file1 = fopen("D:\\учеба\\ОАИП лр\\лр19+\\matrix1.txt", "r");  FILE\* file2 = fopen("D:\\учеба\\ОАИП лр\\лр19+\\matrix2.txt", "r");  FILE\* file3 = fopen("D:\\учеба\\ОАИП лр\\лр19+\\matrix3.txt", "w");  if (!file1 || !file2 || !file3) {  printf("Ошибка открытия файла\n");  return 1;  }  int n, m, p;  fscanf(file1, "%d", &m);  fscanf(file2, "%d", &p);  fscanf(file2, "%d", &n);  int\* A = (int\*)malloc(n \* m \* sizeof(int));  int\* B = (int\*)malloc(m \* p \* sizeof(int));  int\* C = (int\*)malloc(n \* p \* sizeof(int));  if (!A || !B || !C) {  printf("Ошибка выделения памяти\n");  return 1;  }  for (int i = 0; i < n \* m; i++) {  fscanf(file1, "%d", &A[i]);  }  for (int i = 0; i < m \* p; i++) {  fscanf(file2, "%d", &B[i]);  }  matrix(A, B, C, n, m, p);  fprintf(file3, "%d %d\n", n, p);  for (int i = 0; i < n; i++) {  for (int j = 0; j < p; j++) {  fprintf(file1, "%d ", C[i \* p + j]);  }  fprintf(file2, "\n");  }  fclose(file1);  fclose(file2);  fclose(file3);  free(A);  free(B);  free(C);  printf("Произведение матриц записано в файл matrix3.txt\n");  return 0;  }          #define \_CRT\_SECURE\_NO\_DEPRECATE #include <iostream>  #include <stdio.h>  using namespace std;  int main()  {  setlocale(LC\_ALL, "RU");  FILE\* fileA = fopen("D:\\учеба\\ОАИП лр\\лр19+\\A1.txt", "r");  if (fileA == NULL) {  printf("Не удалось открыть файл A1.txt\n");  return 1;  }  else {  printf("Файл A1.txt успешно открыт\n");  }  FILE\* fileB = fopen("D:\\учеба\\ОАИП лр\\лр19+\\B1.txt", "w");  if (fileB == NULL) {  printf("Не удалось открыть файл B1.txt\n");  return 1;  }  else {  printf("Файл B1.txt успешно открыт\n");  }  int n;  fscanf(fileA, "%d", &n);  double\*\* matrix = new double\* [n];  for (int i = 0; i < n; ++i) {  matrix[i] = new double[n];  }  for (int i = 0; i < n; ++i) {  for (int j = 0; j < n; ++j) {  fscanf(fileA, "%lf", &matrix[i][j]);  }  }  for (int i = 0; i < n; ++i) {  for (int j = 0; j < n; ++j) {  fprintf(fileB, "%lf ", matrix[j][i]);  }  fprintf(fileB, "\n");  }  for (int i = 0; i < n; ++i) {  delete[] matrix[i];  }  delete[] matrix;  fclose(fileA);  fclose(fileB);  printf("Транспонированная матрица записана в файл B1.txt\n");  return 0; }        #define \_CRT\_SECURE\_NO\_DEPRECATE #include <iostream>  #include <stdio.h>  using namespace std;  int main()  {  setlocale(LC\_ALL, "RU");  int a, b, c;  FILE\* fileA = fopen("D:\\учеба\\ОАИП лр\\лр19+\\A.txt", "r");  if (fileA == NULL) {  printf("Не удалось открыть файл A.txt\n");  }  else {  printf("Файл A.txt успешно открыт\n");  }  FILE\* fileB = fopen("D:\\учеба\\ОАИП лр\\лр19+\\B.txt", "r");  if (fileB == NULL) {  printf("Не удалось открыть файл B.txt\n");  }  else {  printf("Файл B.txt успешно открыт\n");  }  FILE\* fileC = fopen("D:\\учеба\\ОАИП лр\\лр19+\\C.txt", "r");  if (fileC == NULL) {  printf("Не удалось открыть файл C.txt\n");  }  else {  printf("Файл C.txt успешно открыт\n");  }  FILE\* fileD = fopen("D:\\учеба\\ОАИП лр\\лр19+\\D.txt", "w");  if (fileD == NULL) {  printf("Не удалось открыть файл D.txt\n");  }  else {  printf("Файл D.txt успешно открыт\n");  }  while (true) {  int readA = fscanf(fileA, "%d", &a);  int readB = fscanf(fileB, "%d", &b);  int readC = fscanf(fileC, "%d", &c);  if (readA == EOF || readB == EOF || readC == EOF) {  break;  }  fprintf(fileD, "%d %d %d ", a, b, c);  }  fclose(fileA);  fclose(fileB);  fclose(fileC);  fclose(fileD);  printf("Значения в указанном порядке записаны в файл D.txt\n");  return 0; }            #define \_CRT\_SECURE\_NO\_DEPRECATE #include <stdio.h>  #include <stdlib.h>  #include <locale.h>  #define MAX 1000  void processFileA(const char\* fileA, const char\* fileB) {  int numbers[MAX] = {0};  int count[MAX] = {0};  int size = 0;  FILE\* input = fopen("D:\\учеба\\ОАИП лр\\лр19+\\fileA.txt", "r");  if (!input) {  printf("Ошибка открытия файла %s\n", fileA);  return;  }  int num;  while (fscanf(input, "%d", &num) == 1) {  int found = 0;  for (int i = 0; i < size; i++) {  if (count[i] == num) {  numbers[i]++;  found = 1;  break;  }  }  if (!found) {  count[size] = num;  numbers[size] = 1;  size++;  }  }  fclose(input);  FILE\* output = fopen(fileB, "w");  if (!output) {  printf("Ошибка открытия файла %s\n", fileB);  return;  }  for (int i = 0; i < size; i++) {  if (numbers[i] > 2) {  fprintf(output, "%d\n", count[i]);  }  }  fclose(output);  printf("Файл %s успешно создан.\n", fileB);  }  int main() {  setlocale(LC\_ALL, "RU");  processFileA("fileA.txt", "fileB.txt");  return 0;  }        #define \_CRT\_SECURE\_NO\_DEPRECATE #include <stdio.h>  #include <stdlib.h>  #include <locale.h>  #define MAX 1000  void processFileF(const char\* fileF, const char\* fileG) {  int seen[MAX] = {0};  int size = 0;  FILE\* input = fopen("D:\\учеба\\ОАИП лр\\лр19+\\fileF.txt", "r");  if (!input) {  printf("Ошибка открытия файла %s\n", fileF);  return;  }  FILE\* output = fopen("D:\\учеба\\ОАИП лр\\лр19+\\fileG.txt", "w");  if (!output) {  printf("Ошибка открытия файла %s\n", fileG);  fclose(input);  return;  }  int num;  while (fscanf(input, "%d", &num) == 1) {  int found = 0;  for (int i = 0; i < size; i++) {  if (seen[i] == num) {  found = 1;  break;  }  }  if (!found) {  seen[size++] = num;  fprintf(output, "%d\n", num);  }  }  fclose(input);  fclose(output);  printf("Файл %s успешно создан.\n", fileG);  }  int main() {  setlocale(LC\_ALL, "RU");  processFileF("f.txt", "g.txt");  return 0;  } |