Федеральное государственное автономное

образовательное учреждение

высшего образования

«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий

Кафедра вычислительной техники

**ОТЧЕТ О ПРАКТИЧЕСКОЙ** **РАБОТЕ № 17**

Сортировка массивов  
 Вариант № 19

Преподаватель \_\_\_\_\_\_\_\_ Пушкарев К. В.

подпись, дата

Студент КИ18-09б, 031831293 \_\_\_\_\_\_\_\_ Овсянников В.А.

подпись, дата

Красноярск 2019

**1 Дополнительное упражнение № 2 (вариант 9)**

Разработать, отладить, продемонстрировать и защитить преподавателю графическую схему алгоритма и программу для решения следующей задачи: прочитать из текстового файла данные в массив структур, отфильтровать данные в соответствии с указаниями в таблице №17.1-3 и сохранить отфильтрованные данные в двоичном файле (имя файла вводится с клавиатуры). Вывести содержимое текстового и двоичного файла на монитор в виде таблицы.

Данные в текстовом файле: журналы: название, стоимость одного экземпляра, количество экземпляров в год, тематика (детский, научно-популярный, популярный, научный).

Способ фильтрации для записи в двоичный файл: данные о журналах в алфавитном порядке.

**2 Цель работы**

Получить практические навыки решения задач с использованием

структур.

**3 Графическая схема алгоритма**

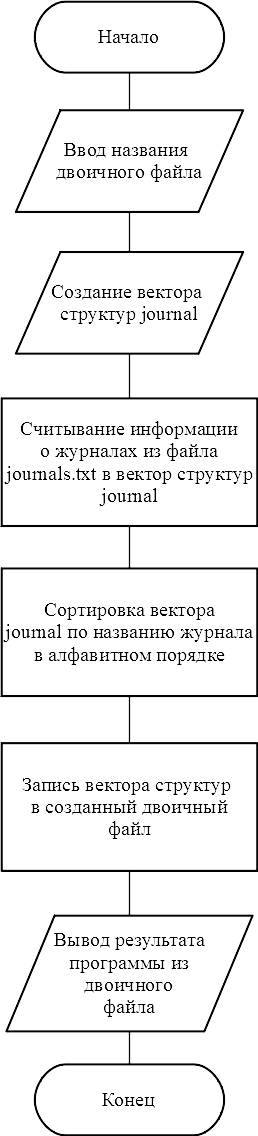


Рисунок 1 – Графическая схема алгоритма

**4 Код программы**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27 28 29 30 31 32 33 34 35 36 37 38 39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117 | #include "pch.h"  #include <iostream>  #include <fstream>  #include <clocale>  #include <iomanip>  #include <vector>  using namespace std;  enum topic\_num {  scientif,  popular,  popularscience,  children  };  struct journal {  char title[20];  float cost;  int number;  topic\_num topic;  };  void print\_journal(journal \*buf) {  cout.setf(ios::left);  cout << "\n " << setw(20) << buf->title;  cout << setw(10) << buf->cost << setw(5) << buf->number;  if (buf->topic == 0) {  cout << "научный" << endl;  }  else if (buf->topic == 1) {  cout << "популярный" << endl;  }  else if (buf->topic == 2) {  cout << "научно-популярный" << endl;  }  else {  cout << "детский" << endl;  }  }  void sort(vector<journal> &buf, int n) {  journal temp;  for (int i = 0; i < n; i++) {  for (int j = 0; j < n; j++) {  if (strcmp(buf[i].title, buf[j].title) < 0) {  temp = buf[i];  buf[i] = buf[j];  buf[j] = temp;  }  }  }  }  int main() {  setlocale(LC\_ALL, "");  ifstream fin("journals.txt");  char filename[255];  cout << "Введеите имя файла: ";  cin.getline(filename, 255);  fstream fout(filename, ios::binary | ios::in | ios::out | ios::trunc);  if (!fin.is\_open()) {  cout << "Не удалось открыть файл!" << endl;  return -1;  }  vector<journal> buf;  int n = 0;  //заполнение массива структур  cout << "\nДанные о журналах из текстового файла: " << endl;  journal buffer;  while (fin >> buffer.title) {  fin >> buffer.cost >> buffer.number;  char buf\_topic[20];  fin >> buf\_topic;  if (strcmp(buf\_topic, "научный") == 0) {  buffer.topic = scientif;  }  else if (strcmp(buf\_topic, "популярный") == 0) {  buffer.topic = popular;  }  else if (strcmp(buf\_topic, "научно-популярный") == 0) {  buffer.topic = popularscience;  }  else {  buffer.topic = children;  }  buf.push\_back(buffer);  print\_journal(&buffer);  n++;  }  sort(buf, n);//сортировка массива структур по алфавиту    //запись в двоичный файл сортированного списка  fout.seekp(ios::beg);  fout.write((char \*)&buf[0], sizeof(buf[0])\*n);  //вывод списка из двоичного файла  cout << "\nДанные о журналах из двоичного файла: " << endl;  fout.seekg(0, ios\_base::beg);    for (int i = 0; i < n; i++) {  fout.read((char \*)&buffer, sizeof(buffer));  print\_journal(&buffer);  }  fin.close();  fout.close();    return 0;  } |

**5 Результаты выполнения экспериментальной части работы**

Результаты запуска программы с различными входными значениями приведены на рисунке 2.

Рисунок 2 – Работа программы на тестовых наборах данных

