

Лабораторная работа № 1

Разработка программы на языке Си

Вариант № 1

В программе определить структуру «Сотрудник» с полями: табельный номер, ФИО, адрес, телефон, пол, дата рождения, должность, оклад, наличие высшего образования.

Информацию о сотрудниках фирмы хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом сотруднике (табельный номер не должен повторяться); 2) поиска информации о сотрудниках, работающих на должность «менеджер» (табельный номер, ФИО, адрес, телефон); 3) поиска информации о сотрудниках с окладом 10 000-15 000 руб. и имеющих высшее образование (табельный номер, ФИО, должность, оклад); 4) отображения информации о сотрудниках фирмы в виде таблицы; 5) определения количества сотрудников женского пола с высшим образованием; 6) определения среднего возраста мужчин.

Вариант № 2

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) подсчета количества строк в файле; 3) определения количества слов в файле; 4) подсчета количества цифр в четных строках файла; 5) поиска самого длинного слова в файле с четным количеством гласных букв; 6) получения в другом файле того же текста, записанного прописными буквами; 7) редактирования текста в файле по следующему правилу: перед запятой не может быть пробела и после запятой должен быть один пробел; 8) вывода на экран содержимого текстового файла.

Вариант № 3

В программе создать структуру «Ученик» с полями: ФИО, адрес, телефон, класс, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
	1	2	3	...
Математика	1	4	1	...
Литература	0	0	3	...
Физика	4	1	5	...
...

где «1» – ученик присутствовал на занятии; 2..5 – полученная оценка, «0» – ученик отсутствовал на занятии).

Данные об учениках хранить в динамическом массиве.

Функционал программы: добавление информации о новом ученике; проставление успеваемости по предметам; поиск информации об отличниках из указанного класса; отображение информации обо всех учащихся; поиск информации об учащихся с указанной фамилией.

Вариант № 4

В программе определить структуру «Учитель» с полями: код, Фамилия, Имя, Отчество, пол, дата рождения, адрес, телефон, преподаваемый предмет, стаж работы.

Информацию об учителях школы хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом учителе (код не должен повторяться); 2) поиска информации об учителях, фамилии которых начинаются на букву «К» (ФИО, адрес, телефон); 3) поиска информации об учителях мужского пола с возрастом менее 40 лет (ФИО, преподаваемый предмет, стаж работы); 4) отображения информации об учителях в виде таблицы; 5) определения количества учителей со стажем работы более 10 лет; 6) определения среднего возраста учителей.

Вариант № 5

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) определения количества знаков препинания в файле; 3) определения чего в файле больше: букв или цифр; 4) определения количества гласных букв в нечетных строках файла; 5) поиска самого длинного слова в файле; 6) получения в другом файле того же текста, записанного строчными буквами; 7) редактирования текста файла по следующему правилу: после точки должен стоять один пробел, а следующая буква должна быть заглавной; 8) вывода на экран содержимого текстового файла.

Вариант № 6

В программе создать структуру «Студент» с полями: ФИО, адрес, телефон, группа, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
	1	2	3	...
Математика	1	4	1	...
Информатика	0	0	3	...
Программирование	4	1	5	...
...

где «1» – студент присутствовал на занятии; 2..5 – полученная оценка, «0» – студент отсутствовал на занятии).

Данные о студентах хранить в динамическом массиве.

Функционал программы: добавление информации о новом студенте; проставление успеваемости по предметам; поиск информации о студентах с успеваемостью выше средней; отображение информации обо всех студентах; отображение информации о студентах из указанной группы.

Вариант № 7

В программе определить структуру «Ученик» с полями: код, Фамилия, Имя, Отчество, пол, дата рождения, телефон, адрес, класс.

Информацию об учениках школы хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом ученике (код не должен повторяться); 2) поиска информации об учениках мужского пола с возрастом 10 лет; 3) отображения информации об учениках из указанного класса (ФИО, класс, дата рождения); 4) отображения информации об учениках в виде таблицы; 5) определения количества учеников женского пола с фамилией на букву «К»; 6) определения количества учеников указанного года рождения.

Вариант № 8

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) сложения всех цифр в файле; 3) определения количества согласных букв в файле; 4) получения в другом файле текста, состоящего из слов с 4-я буквами из исходного файла; 5) поиска самого короткого слова в файле; 6) редактирования текста в файле по следующему правилу: перед дефисом и после дефиса обязательно должны стоять по одному пробелу; 7) указать слово и посчитать сколько раз это слово встречается в текстовом файле; 8) вывода на экран содержимого текстового файла.

Вариант № 9

Учебный центр проводит курсы по программированию. На курсах учащийся изучает различные предметы по программированию («Язык Pascal», «Языке C», «Язык Java» и т.п.).

В программе создать структуру «Учащийся» с полями: ФИО, адрес, телефон, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
	1	2	3	...
Язык Pascal	1	1	1	...
Языке C	0	0	1	...
Язык Java	1	1	1	...
...

где «1» – учащийся присутствовал на занятии; «0» – учащийся отсутствовал на занятии).

Данные об учащихся хранить в динамическом массиве.

Функционал программы: добавление информации о новом учащемся; проставление посещаемости по предметам; поиск информации об учащихся, посетивших более 70% занятий; отображение информации обо всех учащихся; удаление информации об учащемся по ФИО.

Вариант № 10

В программе определить структуру «Книга» с полями: ISBN, название, авторы, издательство, год выпуска, количество страниц.

Информацию о книгах хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новой книге (ISBN не должны повторяться); 2) поиска информации о книгах, в названии которых присутствует слово «история» (название, авторы, издание, год выпуска); 3) поиска информации о книгах указанного издательства и года выпуска (название, авторы); 4) отображения информации о книгах в виде таблицы; 5) определения количества книг указанного года выпуска; 6) определения количества книг указанного издательства и автора.

Вариант № 11

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) замены в тексте файла всех точек на восклицательные знаки, а восклицательных знаков на точки; 3) определения количества цифр в файле; 4) определения количества слов в файле с четным количеством букв; 5) поиска самого короткого слова в файле; 6) проведения частотного анализ текста (указать в процентах сколько раз встречается та или иная буква). Результаты записать в отдельный файл; 7) вывода на экран содержимого текстового файла.

Вариант № 12

В программе создать структуру «Ученик» с полями: Фамилия, Имя, Отчество, адрес, телефон, класс, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
	1	4	1	...
<i>Химия</i>	0	0	3	...
<i>Информатика</i>	4	1	5	...
...

где «1» – ученик присутствовал на занятии; 2..5 – полученная оценка, «0» – ученик отсутствовал на занятии).

Данные об учениках хранить в динамическом массиве.

Функционал программы: добавление информации о новом ученике; проставление успеваемости по предметам; поиск информации об учениках с успеваемостью ниже средней; отображение информации обо всех учениках; поиск информации об учениках из указанного класса.

Вариант № 13

В программе определить структуру «Ученик» с полями: код, Фамилия, Имя, Отчество, пол, дата рождения, телефон, адрес, класс.

Информацию об учениках школы хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом ученике (код не должен повторяться); 2) поиска информации об учениках женского пола, родившихся в 2005 г.; 3) поиска информации об учениках, обучающихся в указанном классе (ФИО, адрес, телефон); 4) отображения информации об учениках в виде таблицы; 5) определения количества учеников с возрастом 10-12 лет; 6) определения среднего возраста учеников.

Вариант № 14

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) подсчета количества строк в файле; 3) подсчета количества знаков препинания в файле; 4) поиска самого длинного и самого короткого слов в файле; 5) редактирования текста файла по следующему правилу: в начале строки не должно быть знаков препинания, в конце строки обязательно должна стоять точка, после запятой обязательно должен стоять один пробел; 6) получения в другом файле слов, состоящих из 5 букв в исходном файле; 7) вывода на экран содержимого текстового файла.

Вариант № 15

В программе создать структуру «Студент» с полями: ФИО, адрес, телефон, группа, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	Номер занятия			
	1	2	3	...
Математика	1	4	1	...
Социология	0	0	3	...
Программирование	4	1	5	...
...

где «1» – студент присутствовал на занятии; 2..5 – полученная оценка, «0» – студент отсутствовал на занятии).

Данные о студентах хранить в динамическом массиве.

Функционал программы: добавление информации о новом студенте; проставление успеваемости по предметам; отображение информации о студентах в виде рейтинга по указанному предмету; отображение информации обо всех студентах.

Вариант № 16

В программе определить структуру «Товар» с полями: код, название, отдел, стоимость, описание.

Информацию о товарах хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом товаре (код не должен повторяться); 2) поиска информации о товарах, в названии которых присутствует слово «шоколад» (название, стоимость, отдел); 3) поиска информации о товарах со стоимостью 100-400 руб. из отдела «Сладости» (название, стоимость, описание); 4) отображения информации о товарах в виде таблицы; 5) определения количества товаров в указанном отделе; 6) определения количества товаров со стоимостью 100-200 руб.

Вариант № 17

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) определения количества строк в файле; 3) определения количества согласных букв в файле; 4) поиска самого короткого слова в файле; 5) указать слово и посчитать сколько раз это слово встречается в текстовом файле; 6) поиска всех слов в четных строках файла с 2 согласными буквами. Результаты записать в отдельный файл; 7) проверки правильности расстановки круглых скобок (т.е. находится ли правее каждой открывающейся скобки закрывающаяся и левее закрывающейся — открывающаяся); 8) вывода на экран содержимого текстового файла.

Вариант № 18

Учебный центр проводит курсы по программированию. На курсах учащийся изучает различные предметы по программированию («Язык Pascal», «Язык C», «Язык Java» и т.п.).

В программе создать структуру «Учащийся» с полями: код, ФИО, адрес, телефон, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
	1	2	3	...
Язык Pascal	1	1	1	...
Языке C	0	0	1	...
Язык Java	1	1	1	...
...

где «1» – учащийся присутствовал на занятии; «0» – учащийся отсутствовал на занятии).

Данные об учащихся хранить в динамическом массиве.

Функционал программы: добавление информации о новом учащемся; проставление посещаемости по предметам; поиск информации об учащихся, изучающих более 3-ёх предметов; поиск информации об учащихся с фамилией на указанную букву; отображение информации обо всех учащихся.

Вариант № 19

В программе определить структуру «Лекарство» с полями: код, название, стоимость, наличие рецепта, описание.

Информацию о лекарствах хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом лекарстве (код не должен повторяться); 2) поиска информации о лекарствах, в названии которых присутствует слово «витамин» (название, стоимость, наличие рецепта); 3) поиска информации о лекарствах со стоимостью 100-400 руб. и продаваемых без рецепта врача (название, стоимость, описание); 4) отображения информации о лекарствах в виде таблицы; 5) определения количества лекарств, отпускаемых без рецепта врача; 6) определения количества лекарств со стоимостью 100-200 руб.

Вариант № 20

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) определения количества строк в файле; 3) определения количества цифр в нечетных строках; 4) поиска самого длинного слова; 5) указать слово, которое необходимо заменить на слово «Мир», и получить в другом файле текст из исходного файла с заменённым словом; 6) редактирования текста в файле по следующему правилу: перед запятой не может быть пробела и после запятой должен быть обязательно один пробел; 7) вывода на экран содержимого текстового файла.

Вариант № 21

В программе создать структуру «Ученик» с полями: код, Фамилия, Имя, Отчество, адрес, телефон, класс, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
	1	4	1	...
Обществознание	1	4	1	...
Информатика	0	0	3	...
История	4	1	5	...
...

где «1» – ученик присутствовал на занятии; 2..5 – полученная оценка, «0» – ученик отсутствовал на занятии).

Данные об учениках хранить в динамическом массиве.

Функционал программы: добавление информации о новом ученике; проставление успеваемости по предметам; поиск информации об учениках, изучающих указанный предмет; удаление информации об ученике по коду; отображение информации обо всех учениках.

Вариант № 22

В программе определить структуру «Студент» с полями: номер зачетки, Фамилия, Имя, Отчество, пол, дата рождения, телефон, адрес, группа.

Информацию о студентах хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом студенте (номер зачетки не должен повторяться); 2) отображения информации о студентах женского пола (ФИО, группа, дата рождения); 3) поиска информации о студентах из указанной группы 1997 г.р. (Фамилия, Имя, Отчество, дата рождения, телефон); 4) отображения информации о студентах в виде таблицы; 5) определения количества студентов мужского пола с фамилией на букву «К»; 6) определения количества студентов указанного года рождения.

Вариант № 23

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) определения количества строк в файле; 3) определения количества букв в файле; 4) поиска в четных строках файла самого длинного слова; 5) задать букву и определить количество слов, начинающихся с указанной буквы; 6) указать слово, которое необходимо заменить на слово «информатика», и получить в другом файле текст из исходного файла с заменённым словом; 7) редактирования текста файла по следующему правилу: перед запятой не может быть пробела и после запятой должен быть обязательно один пробел; 8) вывода на экран содержимого текстового файла.

Вариант № 24

В программе создать структуру «Студент» с полями: номер зачетки, Фамилия, Имя, Отчество, адрес, телефон, группа, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
Математика	1	4	1	...
Информатика	0	0	3	...
Программирование	4	1	5	...
...

где «1» – студент присутствовал на занятии; 2..5 – полученная оценка, «0» – студент отсутствовал на занятии).

Данные о студентах хранить в динамическом массиве.

Функционал программы: добавление информации о новом студенте; проставление успеваемости по предметам; поиск информации о студентах с фамилией на указанную букву; удаление информации о студенте по номеру зачетки; отображение информации обо всех студентах.

Вариант № 25



В программе определить структуру «Участник олимпиады»: код, ФИО, пол, балл за задание №1, балл за задание №2, балл за задание №3, дата рождения, телефон.

Информацию об участниках олимпиады хранить в бинарном файле.

Разработать подпрограммы: 1) добавления информации о новом участнике олимпиады (код не должен повторяться); 2) отображения информации об участниках олимпиады (ФИО, балл за задание №1, балл за задание №2, балл за задание №3, общий балл); 3) поиска информации об участниках олимпиады, занявших 1-3 места (ФИО, общий балл); 4) отображения информации об участниках в виде таблицы; 5) определения количества участников мужского пола, родившихся в 2003 г.; 6) определения среднего возраста участников олимпиады.

Вариант № 26

В программе разработать меню для работы с текстовым файлом.

В программе разработать функции: 1) добавления текста в файл; 2) указать букву и посчитать сколько раз встречается указанная буква в файле; 3) подсчета количества знаков препинания в файле; 4) определения каких букв в тексте больше – гласных или согласных; 5) поиска в нечетных строках файла самого короткого слова; 6) задать букву и найти все слова, начинающиеся с указанной буквы. Результаты поиска записать в отдельный файл; 7) указать две буквы и поменять их местами в тексте файла; 8) вывода на экран содержимого текстового файла.

Вариант № 27

Учебный центр проводит курсы по программированию. На курсах учащийся изучает различные предметы по программированию («Язык Pascal», «Языке С», «Язык Java» и т.п.).

В программе создать структуру «Учащийся» с полями: код, ФИО, адрес, телефон, дата рождения, изучаемые предметы, успеваемость (в виде динамического двумерного массива вида:

Предмет \ Номер занятия	1	2	3	...
<i>Язык Pascal</i>	1	1	1	...
<i>Языке C</i>	0	0	1	...
<i>Язык Java</i>	1	1	1	...
...

где «1» – учащийся присутствовал на занятии; «0» – учащийся отсутствовал на занятии).

Данные об учащихся хранить в динамическом массиве.

Функционал программы: добавление информации о новом учащемся; проставление посещаемости по предметам; поиск информации об учащихся, изучающих предметы на указанную букву и родившихся в заданном году; поиск информации об учащихся с указанной фамилией; отображение информации обо всех учащихся.