Лабораторная работа №6
по курсу «Высокоуровневое программирование»
«Обработка файлов произвольного доступа»

Оглавление

Цель: приобретение практических навыков создания и обработки файлов произвольного доступа

Задачи:

1. Познакомиться с организацией файлов произвольного доступа.
2. Изучить основные программные средства для произвольной обработки файлов указанного типа.

Содержание отчета:

1. Титульный лист.
2. Цель, задачи работы.
3. Формулировка общего задания лабораторной работы.
4. Блок-схемы созданных подпрограмм.
5. Блок-схема основной программы.
   ОБРАТИТЕ ВНИМАНИЕ: для лабораторной работы создаётся один проект, в основной программе реализуется пользовательское меню с применением созданных пользовательских подпрограмм.
6. Листинги пользовательских функций и основной программы.
7. Результаты работы: меню, каждого подпункта отдельно.
8. Выводы по работе в целом.

В начало

Основные теоретические сведения

Файлы произвольного доступа

Произвольный доступ означает возможность перемещения в любую позицию в файле вместо последовательного перемещения по нему. В этом случае необходимо использовать специальные функции, предназначенные для позиционирования в потоках C++:

Класс	Функция-член	Описание
`basic_istream<>`	`tellg()`	Возвращает позицию ввода (*read position*)
	`seekg(pos)`	Устанавливает позицию ввода как абсолютное значение
	`seekg(offset, rpos)`	Устанавливает позицию ввода как относительное значение
`basic_ostream<>`	`tellp()`	Возвращает позицию вывода (*write position*)
	`seekp(pos)`	Устанавливает позицию вывода как абсолютное значение
	`seekp(offset, rpos)`	Устанавливает позицию вывода как относительное значение

Данные функции различают позицию ввода и вывода. Функции для управления позицией ввода определены в классе `basic_istream<>`, а функции для управления позицией вывода — в классе `basic_ostream<>`. Прототипы `seekg()` имеют вид:

```
basic_istream<charT, traits>& seekg(pos_type);
basic_istream<charT, traits>& seekg(off_type, ios_base::seekdir);
```

Для типа `char` приведенные прототипы эквивалентны следующим:

```
istream & seekg(streampos);
istream & seekg(streamoff, ios_base::seekdir);
```

Первый прототип представляет позицию в файле, измеренную в байтах от начала файла (*абсолютная позиция*). Второй прототип представляет позицию в файле, измеренную как смещение в байтах от позиции, заданной вторым аргументом (*относительная позиция*).
Для работы с абсолютными значениями необходимо использовать функции `tellg()` и `tellp()`, которые возвращают абсолютное значение, представляющее текущую позицию в байтах, измеренную от начала файла.

```
std::streampos pos = file.tellg(); //сохраняем текущую позицию в файле
...
file.seekg(pos); // переходим к позиции, сохраненной в переменной pos
```

Для относительных значений смещение может определяться относительно начала файла, относительно текущей позиции или же конца файла. Для каждой позиции в классе `ios_base` определены соответствующие константы.

Константа	**Описание**
`beg`	Смещение задается относительно начала файла
`cur`	Смещение задается относительно текущей позиции
`end`	Смещение задается относительно конца файла

Константа `ios_base::beg` означает отсчет смещения от начала файла, константа `ios_base::cur` — от текущей позиции, а константа `ios_base::end` — от конца файла. Вот некоторые примеры вызовов, предполагающие, что `fin` — объект класса `ifstream`:

```
fin.seekg(30, ios_base::beg);    // 30 байт от начала
fin.seekg(-1, ios_base::cur);    // назад на 1 байт
fin.seekg(0, ios_base::end);     // перейти в конец файла
```

Во всех случаях позиция должна находиться в пределах файла. Если позиция предшествует началу или находится за концом файла, поведение программы не определено.
Следующий пример демонстрирует использование функции `seekg()`. В нем используется функция, которая дважды выводит содержимое файла:

```
#include <iostream>
#include <fstream>

void printFileTwice(const char* filename)
```

```
{
      // открываем файл
      std::ifstream file(filename);

      // выводим содержимое файла первый раз
      std::cout << file.rdbuf() ;

      // переходим в начало
      file.seekg(0);

      // выводим содержимое файла второй раз
      std::cout << file.rdbuf();
}

int main(int argc, char* argv[])
{
      // дважды выводим все файлы, передаваемые как аргумент командной строки
      for(int i = 1; i < argc; ++i){
            printFileTwice(argv[i]);
      }
}
```

Функция `file.rdbuf()` используется для вывода содержимого потока `file`. Таким образом, оператор применяется непосредственно к потоковому буферу и не может изменить состояние потока. Если бы содержимое файла `file` выводилось с помощью функций потокового интерфейса, например `read()`, пришлось бы вызывать функцию `clear()`, чтобы очистить состояние потока `file` перед началом работы с ним (включая изменение позиции ввода), поскольку данные функции после обнаружения конца файла устанавливают флаги `ios::eofbit` и `ios::failbit`.

Ниже приведен фрагмент кода, который открывает файл, переходит в его начало и отображает содержимое:

```
struct S
{
   char str[50];
   int a;
};
...
S s1;

fstream file;
file.open("file.dat", ios_base::in | ios_base::out | ios_base::binary);

if(file.is_open()){
   file.seek(0);

   while(file.read((char *) &s1, sizeof s1))
   {
      cout << s1.str << ": "
           << s1.a << endl;
   }

   if(file.eof())
```

```
      file.clear();     //очистить флаг eof
   else
   {
      cerr << "Error in reading";
      exit(EXIT_FAILURE);
   }
}
else
{
   cerr << "File could not be opened";
   exit(EXIT_FAILURE);
}
```

В начало

Задания

Вариант 1

Задача 1

Запись имеет вид: фамилия, пол, год рождения и рост. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Вывести данные о самом высоком спортсмене.

Задача 2

Файл записей переменной длины перед каждой записью содержит целое, определяющее ее длину. Написать функцию вывода записи в такой файл. Использовать функцию для работы с двумя файлами - строк и динамических массивов целых чисел.

В начало

Вариант 2

Задача 1

Запись имеет вид: название вуза, число студентов, количество факультетов. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Добавить в конец файла информацию о трех новых вузах и посчитать общее число студентов.

Задача 2

Файл записей переменной длины перед каждой записью содержит целое, определяющее ее длину. Написать функцию ввода записи в такой файл. Функция ввода (чтения) должна возвращать размер очередной прочитанной записи. Использовать функцию для работы с двумя файлами - строк и динамических массивов целых чисел.

В начало

Вариант 3

Задача 1

Запись имеет вид: название издания, газеты или журнала, стоимость одного экземпляра, количество экземпляров в год. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Вывести информацию о самом дешевом издании.

Задача 2

Программа создает в файле массив указателей фиксированной размерности на строки текста. Размерность массива находится вначале файла, сами строки также хранятся в файле в виде записей переменной длины. Написать функцию чтения строки из файла по заданному номеру.

В начало

Вариант 4

Задача 1

Запись имеет вид: фамилия студента, номер зачетной книжки, 4 оценки за экзамен. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Выводить информацию обо всех двоечниках и корректировать ее.

Задача 2

Программа создает в файле массив указателей фиксированной размерности на строки текста. Размерность массива находится вначале файла, сами строки также хранятся в файле в виде записей переменной длины. Написать функцию записи строки в файл по заданному номеру.

В начало

Вариант 5

Задача 1

Запись имеет вид: фамилия спортсмена, его номер, количество набранных очков. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Поменять местами в файле записи о первых двух спортсменах.

Задача 2

Упорядоченные по возрастанию строки хранятся в файле в виде массива указателей. Написать функцию включения строки в файл.

В начало

Вариант 6

Задача 1

Запись имеет вид: фамилия, номер телефона, дата рождения. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Внести в начало списка информацию о четырех новых знакомых.

Задача 2

Упорядоченные по возрастанию строки хранятся в файле в виде массива указателей. Написать функцию вывода упорядоченной последовательности строк (просмотр файла).

В начало

Вариант 7

Задача 1

Запись имеет вид: название инструмента, число, месяц и год изготовления. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Вывести на печать информацию об инструменте с самым большим сроком использования и выполнить корректировку этой записи.

Задача 2

Для произвольного текстового файла программа составляет файл записей фиксированной длины, содержащий файловые указатели на строки текстового файла. Программа производит логическое удаление строк, не меняя самого текстового файла.

В начало

Вариант 8

Задача 1

Запись имеет вид: номер читательского билета, автор книги, название, дата заказа. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Поменять местами первую и последнюю записи в файле.

Задача 2

Для произвольного текстового файла программа составляет файл записей фиксированной длины, содержащий файловые указатели на строки текстового файла. Программа производит логическую перестановку строк, не меняя самого текстового файла.

В начало

Вариант 9

Задача 1

Запись имеет вид: фамилия спортсмена, его номер, количество набранных очков. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Удалить из списка информацию о спортсмене с наименьшим количеством очков.

Задача 2

Для произвольного текстового файла программа составляет файл записей фиксированной длины, содержащий файловые указатели на строки текстового файла. Программа производит логическую сортировку строк, не меняя самого текстового файла.

В начало

Вариант 10

Задача 1

Запись имеет вид: фамилия, количество вещей, общий вес. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Удалите из файла сведения о багаже, общий вес вещей в котором меньше, чем 10 кг.

Задача 2

Создать файл, содержащий массив указателей на упорядоченные в алфавитном порядке строки, представленные записями переменной длины. Реализовать функцию поиска строки по строке-образцу, начало которой совпадает с искомой строкой.

В начало

Вариант 11

Задача 1

Запись имеет вид: название команды, количество набранных очков, фамилии капитанов. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Вывести список в порядке набранных мест.

Задача 2

Создать файл, содержащий массив указателей на строки, представленные записями переменной длины. В начале файла - целая переменная - размерность массива указателей. Реализовать функцию загрузки строки по логическому номеру.

В начало

Вариант 12

Задача 1

Запись имеет вид: марка смартфона, стоимость, количество. Создать файл из 10 записей, просмотреть файл, добавить в файл новую информацию. Вывести информацию об имеющихся в продаже смартфонах. При покупке их количество соответственно уменьшается. Предусмотреть удаление информации о товаре, количество которого равно нулю.

Задача 2

Создать файл, содержащий массив указателей на строки, представленные записями переменной длины. Последовательность указателей ограничена NULL-указателем. Реализовать функцию добавления строки по логическому номеру.

В начало

Контрольные вопросы

1. Каково назначение функций `seekg()`, `seekp()`?
2. Каково назначение функций `tellg()`, `tellp()`?
3. Как можно установить позицию файлового указателя в начало файла? в конец файла?
4. В каком случае функции `tellg()` и `tellp()` возвращают одно и то же значение? разные значения?

Список литературы

1. Курс лекций доцента кафедры ФН1-КФ Пчелинцевой Н.И.
2. Программирование на языке высокого уровня C/C++ [Электронный ресурс]: конспект лекций / – Электрон. текстовые данные. – М.: Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2016. – 140 с. – Режим доступа: http://www.iprbookshop.ru/48037.
3. Прата, С. Язык программирования C++. Лекции и упражнения, 6-е изд. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2012 – 1248 с.
4. Джосаттис, Николаи М. Стандартная библиотека C++: справочное руководство, 2-е изд. : Пер. с англ. – М. : ООО «И.Д. Вильямс», 2014 – 1136 с.

В начало