

Ввод&вывод информации в текстовый файл

Подготовил: студент 4 курса
Кулик Д.А.

План презентации

1. Типы файлов.
2. Поточковый способ обработки данных.
3. Алгоритмы ввода&вывода информации в файл.
4. Примеры реализации на **C++**.
5. Рекомендуемая литература.

Основные типы файлов

- **Двоичные** – файлы, в которых информация считывается и записывается в виде блоков определенного размера (в них могут храниться данные любого вида и структуры)
- **Текстовые** – файлы, состоящие из любых символов

Потоковый способ обработки данных из файлов

Данный способ основан на применении определенного типа данных – потоков. Поток – абстракция, используемая для чтения или записи файлов, сокетов и т. п. в единой манере.

Потоки в C++:

- **ifstream (чтение),**
- **ofstream (запись),**
- **fstream (чтение&запись).**

В программах на C++ при работе с текстовыми файлами необходимо подключать библиотеки **iostream** и **fstream**.

Алгоритм записи в текстовый файл

- 1) Описать переменную типа **ofstream**.
- 2) Открыть файл с помощью функции **open**.
- 3) Вывести информацию в файл.
- 4) Закрыть файл.

Алгоритм считывания из текстового файла

- 1) Описать переменную типа **ifstream**.
- 2) Открыть файл с помощью функции **open**.
- 3) Считать информацию из файла, при считывании каждой порции данных необходимо проверять, достигнут ли конец файла.
- 4) Закрывать файл.

Примеры реализации записи в файл в C++

//первый способ

ofstream F;

F.open("D:\\WORK\\test.txt", ios::out);

//второй способ, режим ios::out является режимом по умолчанию

//для потока ofstream

ofstream F;

F.open("D:\\WORK\\test.txt");

//третий способ объединяет описание переменной и типа поток

//и открытие файла в одном операторе

ofstream F ("D:\\WORK\\test.txt", ios::out);

Пояснение

Файл может быть открыт в одном из следующих режимов:

- ***ios::in*** — открыть файл в режиме чтения данных, этот режим является режимом по умолчанию для потоков ***ifstream***;
- ***ios::out*** — открыть файл в режиме записи данных (при этом информация о существующем файле уничтожается), этот режим является режимом по умолчанию для потоков ***ofstream***;
- ***ios::app*** — открыть файл в режиме записи данных в конец файла;
- ***ios::ate*** — передвинуться в конец уже открытого файла;
- ***ios::trunc*** — очистить файл, это же происходит в режиме ***ios::out***;
- ***ios::nocreate*** — не выполнять операцию открытия файла, если он не существует;

Примеры реализации считывания из файла в C++

// Единичное считывание

```
ifstream F;
```

```
F.open("D:\\game\\noobs.txt", ios::in);
```

```
F>>a; /* считывание первого элемента  
текстового файла */
```

// Считывание целого файла

```
while (!F.eof()) // цикл прервется, когда
```

```
{ // достигнет конца
```

```
a // файла
```

```
F>>a;
```

```
}
```

Рекомендуемая литература

1. Лафоре Р. Объектно-ориентированное программирование в С++ // Р. Лафоре, изд. 4-ое. – Санкт-Петербург, ПИТЕР, 2004. – 450 с.
2. Паппас К. Х., Мюррей III У. Х. - Отладка в С++. Руководство для разработчиков // К. Х. Паппас, У. Х. Мюррей III – Москва, Бином, McGraw-Hill Companies, 2009. – 700 с.
3. Страуструп Бьерн. Язык программирования С++ // Бьерн Страуструп, изд. 2 – Москва, Бином-Пресс, 2005. – 200 с.