Секция «Вычислительная математика и кибернетика»

Интерфейс на основе шаблонных классов C++ для работы с большими типизированными файлами.

Ефимов Александр Юрьевич
Студент
Московский государственный университет имени М.В. Ломоносова, Факультет вычислительной математики и кибернетики, Москва, Россия
E-mail: efimov.alexander@gmail.com

При обработке больших объёмов данных часто встаёт задача хранения и быстрого доступа к последовательностям одинаковых элементов. Такие данные удобно хранить в типизированных файлах — бинарных файлах, состоящих из последовательностей элементов одного типа. Структура такого файла похожа на обычный массив, но в отличие от него, объём данных, хранящихся в файле, может быть больше объёма доступной программе оперативной памяти.

При создании ПО приходится каждый раз заново писать код, отвечающий за работу с типизированными файлами. Выделение такого кода в отдельную библиотеку с простым интерфейсом могло бы упростить и ускорить процесс разработки. Также, при добавлении возможности работы с большими объёмами данных, хранящимися в файлах, в уже имеющемся ПО, работающим с массивами, приходится значительно переписывать код, отвечающий за обработку этих данных. Можно решить обе проблемы, реализовав аналог вектора из стандартной библиотеки шаблонов (STL), работающий с данными хранящимися в файле.

К описанному вектору предъявляется несколько требований: небольшой расход памяти, большая скорость обработки запросов, интерфейс, схожий с интерфейсом стандартного вектора. Так как ускорить сам процесс чтения данных из файла не представляется возможным, необходимо кэшировать прочитанные данные. В реализованной системе используется программный частично ассоциативный кэш. Для поиска выталкиваемых элементов при переполнении кэша применяется алгоритм LRU[2].

Подстраивая параметры кэша можно добиться необходимого расхода памяти, удовлетворяющего ограничениям, наложенным на программу. С помощью механизма перегрузки операторов C++ реализован интерфейс, аналогичный интерфейсу вектора из STL. Возникает также проблема определения многомерных векторов. Эта задача решается с помощью механизма частичного определения шаблонов C++[1].

В рамках данной работы была реализована система, позволяющая свести обработку файлов к работе с векторами, таким образом упрощающая модернизацию существующего ПО, разработку нового и улучшая читаемость кода. Данное решение показало более высокую скорость работы, чем наивное чтение требуемых структур из файла без применения алгоритмов кэширования. Результаты экспериментов показали, что данное средство может быть использовано даже в алгоритмах с интенсивными обращениями к данным (например, сортировка), не помещающимся в основную память целиком.

Литература

1. Страуструп Б. Язык Программирования C++. Специальное издание. Бином, 2011. С. 417-420.

2. LRU. http://en.wikipedia.org/wiki/Cache_algorithms#Least_Recently_Used