

Практика №3-4. Разработка и программная реализация задач поиска данных в таблицах с применением механизма хеширования.

Тема. Применение хеш-таблицы для поиска данных в двоичном файле с записями фиксированной длины

Цель. Получить навыки по разработке хеш-таблиц и их применению при поиске данных в других структурах данных (файлах).

Задание

Разработать приложение, которое использует хеш-таблицу для организации прямого доступа к записям двоичного файла, реализованного в практической работе 2.

Метод разрешения коллизии представлен в вашем варианте задания в таб.1.

Требования к выполнению

1. Создать приложение и включить в него три заголовочных файла: управление хеш-таблицей, управление двоичным файлом (практическая работа 2), управление двоичным файлом посредством хеш-таблицы. Имена заголовочным файлам определите сами. Подключите заголовочные файлы к приложению.
2. Для обеспечения прямого доступа к записи в файле элемент хеш-таблицы должен включать обязательные поля: ключ записи в файле, номер записи с этим ключом в файле. Элемент может содержать другие поля, требующиеся методу (указанному в вашем варианте), разрешающему коллизию.
3. Управление хеш-таблицей.
 - 1) Определить структуру элемента хеш-таблицы и структуру хеш-таблицы в соответствии с методом разрешения коллизии, указанным в варианте. Определения разместить в соответствующем заголовочном файле. Все операции управления хеш-таблицей размещать в этом заголовочном файле.
 - 2) Тестирование операций выполнять в функции main приложения по мере их реализации.

После тестирования всех операций, создать в заголовочном файле функцию с именем testHeshT переместить в нее содержание функции main, проверить, что приложение выполняется.

Разработать операции по управлению хеш-таблицей.

- 3) Разработать хеш-функцию (метод определить самостоятельно), выполнить ее тестирование, убедиться, что хеш (индекс элемента таблицы) формируется верно.
- 4) Разработать операции: вставить ключ в таблицу, удалить ключ из таблицы, найти ключ в таблице, рехешировать таблицу. Каждую операцию тестируйте по мере ее реализации.
- 5) Подготовить тесты (последовательность значений ключей), обеспечивающие:
 - вставку ключа без коллизии
 - вставку ключа и разрешение коллизии
 - вставку ключа с последующим рехешированием
 - удаление ключа из таблицы
 - поиск ключа в таблице

Примечание. Для метода с открытым адресом подготовить тест для поиска ключа, который размещен в таблице после удаленного ключа, с одним значением хеша для этих ключей.

- 6) Выполнить тестирование операций управления хеш-таблицей. При тестировании операции вставки ключа в таблицу предусмотрите вывод списка индексов, которые формируются при вставке элементов в таблицу.

4. Управление двоичным файлом.

Операции управления двоичным файлом: создание двоичного файла из текстового, добавить запись в двоичный файл, удалить запись с заданным ключом из файла, прочитать запись файла по заданному номеру записи.

Примечание. Эти операции должны быть отлажены в практической работе 2, или уже в этой работе, если их пока нету.

Структура записи двоичного файла и все операции, по управлению файлом, должны быть размещены в соответствующем заголовочном файле.

Выполнить тестирование операций в main приложения, и содержание функции main переместить в соответствующую функцию заголовочного файла с именем testBinF.

5. Управление файлом посредством хеш-таблицы.

В заголовочный файл управления файлом посредством хеш-таблицы подключить заголовочные файлы: управления хеш-таблицей, управления двоичным файлом. Реализовать поочередно все перечисленные ниже операции в этом заголовочном файле, выполняя их тестирование из функции main приложения. После разработки всех операций выполнить их комплексное тестирование.

Разработать и реализовать операции.

- 1) Прочитать запись из файла и вставить элемент в таблицу (элемент включает: ключ и номер записи с этим ключом в файле, и для метода с открытой адресацией возможны дополнительные поля).
- 2) Удалить запись из таблицы при заданном значении ключа и соответственно из файла.
- 3) Найти запись в файле по значению ключа (найти ключ в хеш-таблице, получить номер записи с этим ключом в файле, выполнить прямой доступ к записи по ее номеру)..

4) Подготовить тесты для тестирования приложения:

Заполните файл небольшим количеством записей.

- Включите в файл записи как не приводящие к коллизиям, так и приводящие.
- Обеспечьте включение в файл такого количества записей, чтобы потребовалось рехеширование.

Заполните файл большим количеством записей (до 1 000 000).

- Определите время чтения записи с заданным ключом: для первой записи файла, для последней и где-то в середине. Убедитесь (или нет), что время доступа для всех записей одинаково.

6. Составить отчет

Структура отчета представлена в приложении 1.

7. Ответы на вопросы

1. Расскажите о назначении хеш-функции.

2. Что такое коллизия?
3. Что такое «открытый адрес» по отношению к хеш-таблице?
4. Как в хеш-таблице с открытым адресом реализуется коллизия?
5. Какая проблема, может возникнуть после удаления элемента из хеш-таблицы с открытым адресом и как ее устранить?
6. Что определяет коэффициент нагрузки в хеш-таблице?
7. Что такое «первичный кластер» в таблице с открытым адресом?
8. Как реализуется двойное хеширование?

Таблица 1. Варианты заданий к практической работе

№	Тип хеш-таблицы (метод разрешения коллизии)	Структура записи двоичного файла
1	С открытой адресацией (смещение на 1)	Читательский абонемент: номер читательского билета – целое пятизначное число, ФИО, адрес.
2	С открытой адресацией (смещение на номер выполняемого подбора)	Счет в банке: номер счета – 7 разрядное число, ФИО, адрес.
3	С открытой адресацией (двойное хеширование)	Владелец телефона: номер телефона – последовательность символов, адрес, ФИО.
4	Цепное хеширование	Владелец автомобиля: номер машины, марка, сведения о владельце, сведения об угоне (логического типа).
5	Цепное хеширование	Пациент поликлиники: номер карточки, код хронического заболевания, фамилия лечащего врача.
6	Цепное хеширование	Товар: название, код – шестизначное число, завод изготовитель, цена, страна (название).
7	Цепное хеширование	Специализация вуза: код

		специальности, название вуза, название специальности.
8	Открытый адрес(двойное хеширование)	Книга: ISBN – двенадцатизначное число, автор, название, год издания.
9	Цепное хеширование	Страховой полис: номер, компания, фамилия владельца.
10	Открытый адрес(смещение на 1)	Англо-русский словарь: английское слово, русское слово.
11	Открытый адрес(двойное хеширование)	Железнодорожная справка: номер поезда, пункт отправления, пункт назначения, время отправления.
12	Цепное хеширование	Регистрация малого предприятия: номер лицензии (текстовое значение), название, учредитель, признак действия лицензии (0 действует, 1 отозвана).
13	Открытый адрес(двойное хеширование)	Студент: номер зачетной книжки, номер группы, ФИО.
14	Цепное хеширование	Справочная межгорода: код города, название города, страна.
15	Открытый адрес (смещение на 1)	Найти и поздравить друга: <u>дата</u> <u>рождения</u> , имя
16	Цепное хеширование	Расписание занятий группы: номер группы, название дисциплины, номер пары, номер недели, номер дня недели, вид занятия, номер аудитории.
17	Открытый адрес (смещение на 1)	Частотный словарь: слово, количество вхождений в текст.
18	Открытый адрес (двойное хеширование)	Читательский билет: номер, инвентарный номер книги, дата

		выдачи, дата возврата.
19	Цепное хеширование	Вызов такси: номер, фамилия водителя, время выезда, отметка о присутствии в гараже.
20	Открытый адрес(смещение на 1)	Продажи товаров: код товара, название, цена, дата продажи.
21	Открытый адрес(двойное хеширование)	Сотрудник: табельный номер, должность, оклад, количество детей.
22	Цепное хеширование	Расписание занятий группы: номер группы, название дисциплины, номер пары, номер недели, номер дня недели, вид занятия, номер аудитории.
23	Открытый адрес(двойное хеширование)	Нагрузка по дисциплине: <u>код дисциплины</u> , <u>код направления подготовки</u> , название дисциплины, номер семестра проведения дисциплины.
24	Цепное хеширование	Нагрузка по дисциплине: <u>код дисциплины</u> , <u>код направления подготовки</u> , название дисциплины, номера семестров проведения дисциплины (не более двух). <u>Подсказка</u> . Если только один семестр, то второе поле должно содержать 0.
25	Открытый адрес (смещение на 1)	Аэропорт (табло прибытия пассажирских авиарейсов сохраняет все данные в файле): пункт вылета, <u>номер рейса</u> , дата прилета, время прилета, информация о задержке прилета в часах.

26	Цепное хеширование	Учет заболеваний пациента. Структура записи о пациенте: <u>номер полиса</u> , фамилия, имя, отчество, код заболевания, дата установки диагноза, код врача.
27	Открытый адрес (смещение на 1)	Учет техосмотра автомобилей. Структура записи об автомобиле: <u>Номер</u> (код региона, цифровой код, буквенный код), Модель, Цвет, Сведения о владельце (Фамилия, Имя, Адрес), дата последнего техосмотра.
28	Открытый адрес (смещение на 1)	Учет нарушений ПДД. Структура записи о нарушении ПДД: <u>номер автомобиля</u> , фамилия и инициалы владельца, модель, дата нарушения, место нарушения (текстом), статья (КоАП), наказание (сумма штрафа).
29	Открытый адрес (двойное хеширование)	Справочник банков по городам страны. Об отдельном банке хранятся данные: наименование, <u>код банка</u> , адрес (город), форма собственности (коммерческий или государственный).
30	Цепное хеширование	Касса магазина. Структура записи операции по кассе: <u>номер кассы</u> , код товара, количество товара, цена товара, процентная скидка на товар, сумма за товар с учетом скидки.
31	Открытый адрес (смещение на 1)	Киноафиша города. Структура записи о сеансе: название кинотеатра, название фильма, дата, время начала, стоимость билета.

Приложение 1 Образец оформления отчета по разработке приложения

Условие задачи

Разработать приложение, которое использует хеш-таблицу для организации прямого доступа к записям файла, структура записи которого приведена в варианте.

Задание варианта

№	Метод хеширования (способ реализации коллизий)

1. Постановка задачи

Дано.

Файл двоичный с записями фиксированной длины.

Структура записи файла согласно варианту (изобразить в форме таблицы, указав названия полей).

Результат.

Хеш-таблица.

2. Подход к решению

1) Файл двоичный из записей фиксированного размера.

Указать:

Структура записи файла из кода.

Размер записи файла в байтах.

Прототипы операций по управления двоичным файлом с указанием предусловия и постусловия.

2) Хеш-таблица

Структура элемента таблицы: изобразить в виде строки таблицы, указать названия полей, их назначение.

Привести код элемента таблицы, реализацию структуры таблицы.

Описание алгоритмов операций:

- вставка ключа в таблицу
- поиск записи по ключу в таблице и возвращение номера записи в файле
- удаление элемента из хеш-таблицы

3) Алгоритм поиска записи с заданным ключом в файле посредством хеш-таблицы.

3. Код приложения

4. Тестирование операций управления файлом посредством хеш-таблицы
- Скриншоты результатов выполнения операций с хеш-таблицей в соответствии с требованиями задания п.3.
 - Скриншоты содержания файла и хеш-таблицы тестирования требований задания п.5.

5. Результат тестирования.

Заполните файл большим количеством записей (до 1 000 000).

Определите время чтения записи с заданным ключом: для первой записи файла, для последней и где-то в середине. Убедитесь (или нет), что время доступа для всех записей одинаково.