

# Лабораторная работа J: Хеширование

## Лабораторная работа J: Хеширование

### Хеш-функции

Вариант 10: CRC16 (1)

Вариант 11: CRC32 (1)

### Хеш-таблицы

Вариант 20: Разрешение коллизий с помощью цепочек (2)

Вариант 21: Открытая адресация с линейной последовательностью проб (2)

Вариант 22: Открытая адресация с квадратичной последовательностью проб (2)

Вариант 23: Двойное хеширование (2)

Вариант 24: Двойное хеширование с изменениями Брента (2)

Вариант 25: Удаление элемента из хеш-таблицы с открытой адресацией (2)

### Внешний поиск

Вариант 30: Использование цепочек с отдельными списками (3)

Вариант 31: Линейное хеширование (3)

### Разное

Вариант 40: Фильтр Блума (2)

Вариант 41: Хеширование в двоичном дереве поиска (3)

Вариант 42: Алгоритм Рабина-Карпа (3)

Вариант 43: Соединение хешированием (2)

### TODO

## Хеш-функции

---

### Вариант 10: CRC16 (1)

---

- Оцените возможность использования функции CRC16 в качестве хеш-функции, если исходными данным выступают ASCII-Z строки
- Оцените равномерность хеширования произвольного набора данных.
- Найдите коллизию (две ASCII-Z строки с одинаковым хеш-значением)

### Вариант 11: CRC32 (1)

---

- Оцените возможность использования функции CRC32 в качестве хеш-функции, если исходными данным выступают ASCII-Z строки
- Оцените равномерность хеширования произвольного набора данных.
- Найдите коллизию (две ASCII-Z строки с одинаковым хеш-значением)

## Хеш-таблицы

---

### Вариант 20: Разрешение коллизий с помощью цепочек (2)

---

- Реализуйте хеш-таблицу элементами которой являются ASCII-Z строки. В качестве метода разрешения коллизий выберете метод цепочек.

### Вариант 21: Открытая адресация с линейной последовательностью проб (2)

---

- Реализуйте хеш-таблицу элементами которой являются ASCII-Z строки. В качестве метода разрешения коллизий выберете открытую адресацию с линейной последовательностью проб.

## **Вариант 22: Открытая адресация с квадратичной последовательностью проб (2)**

---

- Реализуйте хеш-таблицу элементами которой являются ASCII-Z строки. В качестве метода разрешения коллизий выберете открытую адресацию с квадратичной последовательностью проб.

## **Вариант 23: Двойное хеширование (2)**

---

- Реализуйте хеш-таблицу элементами которой являются ASCII-Z строки. В качестве метода разрешения коллизий выберете двойное хеширование.

## **Вариант 24: Двойное хеширование с изменениями Брента (2)**

---

- Реализуйте хеш-таблицу элементами которой являются ASCII-Z строки. В качестве метода разрешения коллизий выберете двойное хеширование с изменениями Брента.

## **Вариант 25: Удаление элемента из хеш-таблицы с открытой адресацией (2)**

---

- Реализуйте алгоритм удаления из хеш-таблицы с открытой адресацией.

## **Внешний поиск**

---

### **Вариант 30: Использование цепочек с отдельными списками (3)**

---

Часто размер хеш-таблицы настолько велик, что она может храниться только во внешней памяти (например, на жестком диске, или распределено по сети). В этом случае алгоритмы работы с хеш-таблицами претерпевают некоторые изменения. Чаще всего пытаются минимизировать количество обращений во внешнюю память, т.к. это обычно довольно медленная операция. Одним из способов реализации хеш-таблицы во внешней памяти является следующий:

- Хеш-таблица состоит из  $M$  блоков по  $b$  записей.
- Отдельно есть область переполнения (например, в конце файла).
- Хешируемое значение попадает в один из  $M$  блоков. Если этот блок уже содержит  $b$  записей, то значение записывается в область переполнения.
- Область переполнения общая для всех блоков, однако мы можем ожидать, что переполнения будут случаться редко (иначе логичнее увеличить параметр  $b$ ) Реализуйте хеш-таблицу во внешней памяти, указанным выше способом.

### **Вариант 31: Линейное хеширование (3)**

---

Другим (см. упр 30) способом представления хеш-таблицы во внешней памяти является [линейное хеширование](#). Реализуйте алгоритм линейного хеширования.

## **Разное**

---

### **Вариант 40: Фильтр Блума (2)**

---

Реализуйте алгоритм [Фильтра Блума](#)

## Вариант 41: Хеширование в двоичном дереве поиска (3)

---

При построении двоичного дерева поиска может возникнуть ситуация когда дерево сильно не сбалансировано. В частности это может произойти если добавлять в дерево упорядоченную последовательность ключей. Чтобы избежать этого в двоичном дереве поиска мы можем сравнивать элементы не по значению (ключу), а по хеш-функции от ключа. Если хеш-функция выбрана правильно, то упорядоченная последовательность не будет нарушать свойства сбалансированности.

- Реализуйте двоичное дерево поиска по ключу, и с помощью хеш-функции.
- Каким образом надо обрабатывать коллизии?
- Хеш-функция не гарантирует нам поддержания сбалансированности дерева, она лишь делает последовательность элементов, приводящих к несбалансированному дереву, менее очевидной. Как модифицировать алгоритм, чтобы при многократных запусках вашей программы среднее время поиска элемента на дереве стремилась бы к логарифмическому.
- Приведите примеры, когда алгоритм с хеш-функцией работает быстрее чем простой, медленнее чем простой. Покажите, что модифицированный алгоритм в среднем работает лучше немодифицированного.

## Вариант 42: Алгоритм Рабина-Карпа (3)

---

Вам даны два текстовых файла, в первом из них есть текст, во втором множество строк. Найдите все вхождения строк из второго файла в тексте.

- Используйте алгоритм Рабина-Карпа для ускорения поиска.

## Вариант 43: Соединение хешированием (2)

---

При работе с базами данных часто приходится выполнять операцию соединения для двух и более таблиц. Реализуйте алгоритм соединения двух таблиц хешированием. Исходные таблицы считывайте из файла. Результат конкатенации запишите в файл.

## TODO

---

TODO: Хеш функция Дженкинса, [MurmurHash2?](#) , [cityhash](#)

-- [OlegTalalov](#) - 20 Aug 2010

---

Topic revision: r6 - 23 May 2014 - 07:14:45 - [OlegTalalov](#)

ComputerScience.LabJ moved from ComputerScience.LabC on 23 Sep 2010 - 05:59 by [OlegTalalov](#) - [put it back](#)

Copyright © by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

Ideas, requests, problems regarding TWiki? [Send feedback](#)

