

ЗАДАНИЕ

на лабораторную работу №2

В программу «Построение хеш-таблицы» (лабораторная работа №1) добавить функции, позволяющие:

1. Сгенерировать или ввести в интерактивном режиме новые элементы.
2. Осуществить поиск элемента в таблице.

Проверить, имеется ли такой элемент в таблице.

Если есть — вывести его индекс, если нет-сообщение: «Элемент не найден».

3. Добавить элемент в таблицу.

Проверить, имеется ли такой элемент в таблице.

Если есть — вывести его индекс, если нет — добавить в хеш-таблицу и вывести сообщение, в какой индекс добавлен элемент.

4. Удалить элемент из таблицы.

Проверить, имеется ли такой элемент в таблице.

Если есть — удалить и вывести сообщение, с какого индекса удален, если нет-сообщение: «Элемент не найден».

4. Ввести в интерактивном режиме:

- элемент, который следует заменить на другое значение,
- элемент, который требуется занести в таблицу.

Осуществить замену.

5. Произвести анализ следующих параметров полученной хеш-таблицы:

- коэффициент заполнения таблицы α , который равен отношению занятой памяти ко всей имеющейся или, другими словами, отношению количества заполненных ячеек к размеру таблицы;

- среднее число шагов необходимых для размещения некоторого ключа в таблице.

Порядок выполнения работы

1. Запустить программу построения хеш-таблицы, разработанную на первой лабораторной работе.

Полученная хеш-таблица является входной информацией для второй работы.

Входными данными также будут переменные, обозначающие элементы, которые требуется найти, добавить, удалить, заменить.

Пользователю предлагается работать с уже готовой хеш-таблицей.

2. Разработать интерфейс, который предоставляет пользователю выбрать нужную функцию по работе с хеш-таблицей, например,

cout << "\n Выберите нужный вариант : \n 1- найти элемент \n 2- удалить элемент \n 3- добавить новый элемент \n 4- Заменить элемент \n";

3. В результате выбора и активации нужного варианта программа должна выполнить необходимую операцию.

4. После завершения работы каждой функции напечатать соответствующее сообщение и обновленную хеш-таблицу.

5. В первую очередь реализовать функцию поиска элемента, так как на основе от полученного результата формируется сообщение и дальнейшие действия.

Функция поиска элемента

Ввести элемент с клавиатуры.

Вычислить хеш-адрес для этого элемента.

Проверить

Если ячейка по этому адресу пуста, например, равна 0,

выдать сообщение «Элемента в таблице нет».

Иначе сравнить значение по полученному хеш-адресу с введенным элементом.

Если значения равны, то элемент найден, вывести элемент и его индекс.

Если значения не равны, проверить по адресам, формируемым методом устранения коллизий до тех пор, пока либо не будет найден искомый элемент, либо пустая ячейка, либо максимальное количество итераций.

Выдать соответствующее сообщение.

6. Реализовать функцию **удаления**, позволяющую отследить коллизии по индексу с исключаемым элементом.

Например, если таблица инициализировалась значением **0**, то при удалении заполнить ячейку значением **-1**.

7. Реализовать функцию добавления элементов либо введенных с клавиатуры, либо сгенерированных случайным образом.

Функция добавления элемента

1. С помощью функции поиска проверить наличие элемента в таблице.

Если в таблице такой элемент уже есть, выдать сообщение.

Иначе добавить в ячейку, значение которой равно **0** или **-1**.

8. Реализовать функцию замены элементов.

Функция замены элемента

1. Ввести значение удаляемого элемента.

2. С помощью функции поиска проверить его наличие в таблице.

Если элемент отсутствует, выдать сообщение,

Иначе запустить функцию удаления, напечатать «элемент **k** удален из **индекс**».

3. Ввести значение удаляемого элемента.

2. С помощью функции поиска проверить его наличие в таблице.

Если элемент существует, выдать сообщение,

Иначе запустить функцию добавления.

Формы отчетности

1. На экране дисплея должны отображаться:
 - Начальная таблица и коэффициенты,
 - Диалог вызова функций,
 - Ввод значений элементов,
 - Сообщения о результате работы соответствующей функции,
 - Обновленная хеш-таблица,
 - Рассчитанные параметры.
2. Электронный вариант программы.
3. Отчет о проделанной лабораторной работе, содержащий следующие пункты:
 - задание на лабораторную работу, содержащее используемые хеш-функцию, метод устранения коллизий;
 - описание идентификаторов, задающих входные и выходные данные;

Например,

Входные данные:

- Table[] – исходная хеш-таблица;
- Size – размерность хеш-таблицы Table[];
- Occurancy – количество элементов в таблице;
- Key1 – первое значение, введенное с клавиатуры;
- key2 – второе значение, введенное с клавиатуры.

Выходные данные:

- Table[] – итоговая хеш-таблица;
- Occurancy – итоговое количество элементов в таблице;
- CoefOccurancy – коэффициент заполнения таблицы;
- AverageCountAttempt – среднее число проб.
- Описание шагов, отражающих:
 - ✓ разработку интерфейса работы с хеш-таблицей,

- ✓ алгоритмы поиска, удаления, добавления, замены с учетом заданных хеш-функции и метода устранения коллизий, после описания каждого шага – кусочек кода, реализующий данный шаг.
- формулы расчета требуемых параметров хеш-таблицы
- примеры работы программы, демонстрирующие все варианты работы программы.