

Практическая работа №3

Тема: «Хэш-таблицы»

Цель работы: изучить реализацию хэш-таблиц.

Хеш-таблица — это структура данных, реализующая интерфейс ассоциативного массива, а именно, она позволяет хранить пары (ключ, значение) и выполнять три операции: операцию добавления новой пары, операцию поиска и операцию удаления пары по ключу.

Создадим хэш-таблицу с реализацией метода открытой адрессации для простейшего телефона справочника. Для этого определим структуру контакта, которая представлена в листинге 1.

Листинг 1. Структура контакта.

Для одной ячейки таблицы определим следующую структуру, представленную в листинге 2.

Листинг 2. Структура ячейки таблицы.

Где empty – флаг, указывающий, что ячейка свободна, в независимости от содержащихся там данных.

visit – флаг, указывающий, что ячейка просматривалась.

Для вычисления значения хэша будем использовать следующую функцию, представленную в листинге 3.

Листинг 3. Хэш-функция.

Диаграмма деятельности для этой функции представлена на рисунке 1.

					АИСД.09.03.02.220000 ПР			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Третьяк И.Н.			Практическая работа №3 Хэш-таблицы	Лит.	Лист	Листов
Проверил		Береза А.Н.						
Реценз						ИСОиП (филиал) ДГТУ в г.Шахты ИСТ-Тб21		
Н. Контр.								
Утверд.								

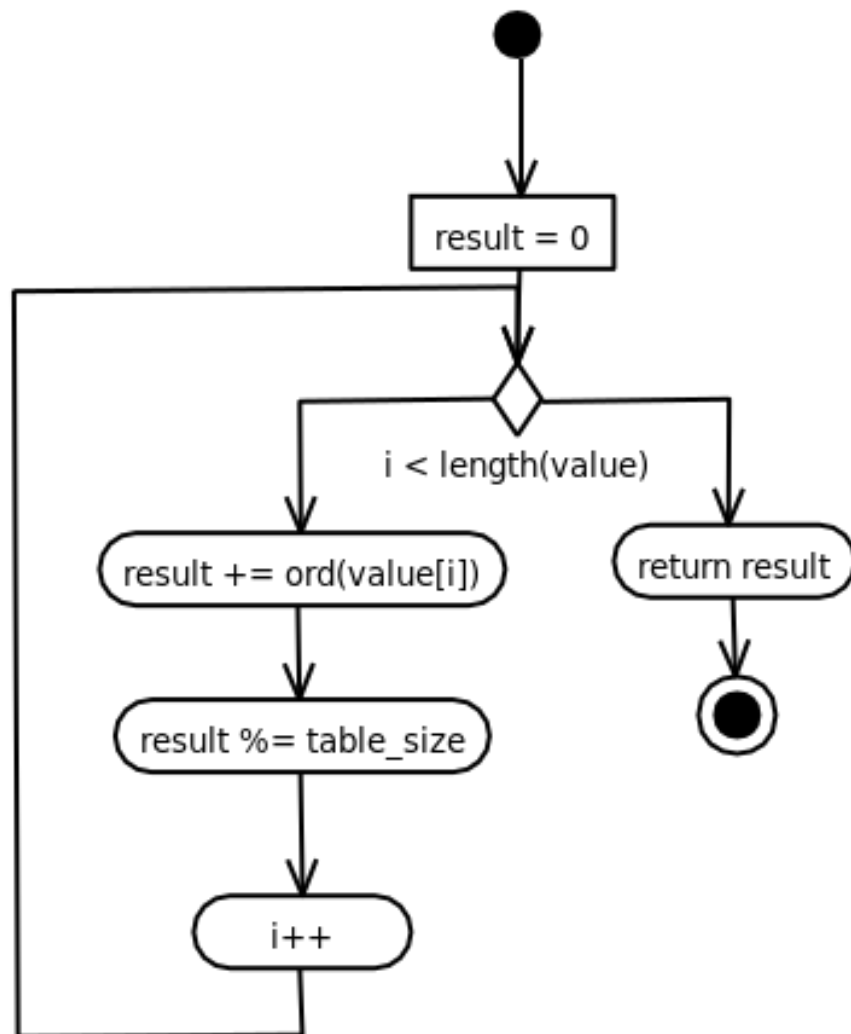


Рисунок 1 - Диаграмма деятельности для __hash_function

Листинг функции добавления элемента представлен в Листинге 4.
Листинг 4. Добавление элемента в хэш-таблицу.

Диаграмма деятельности для добавления элемента представлена на рисунке .

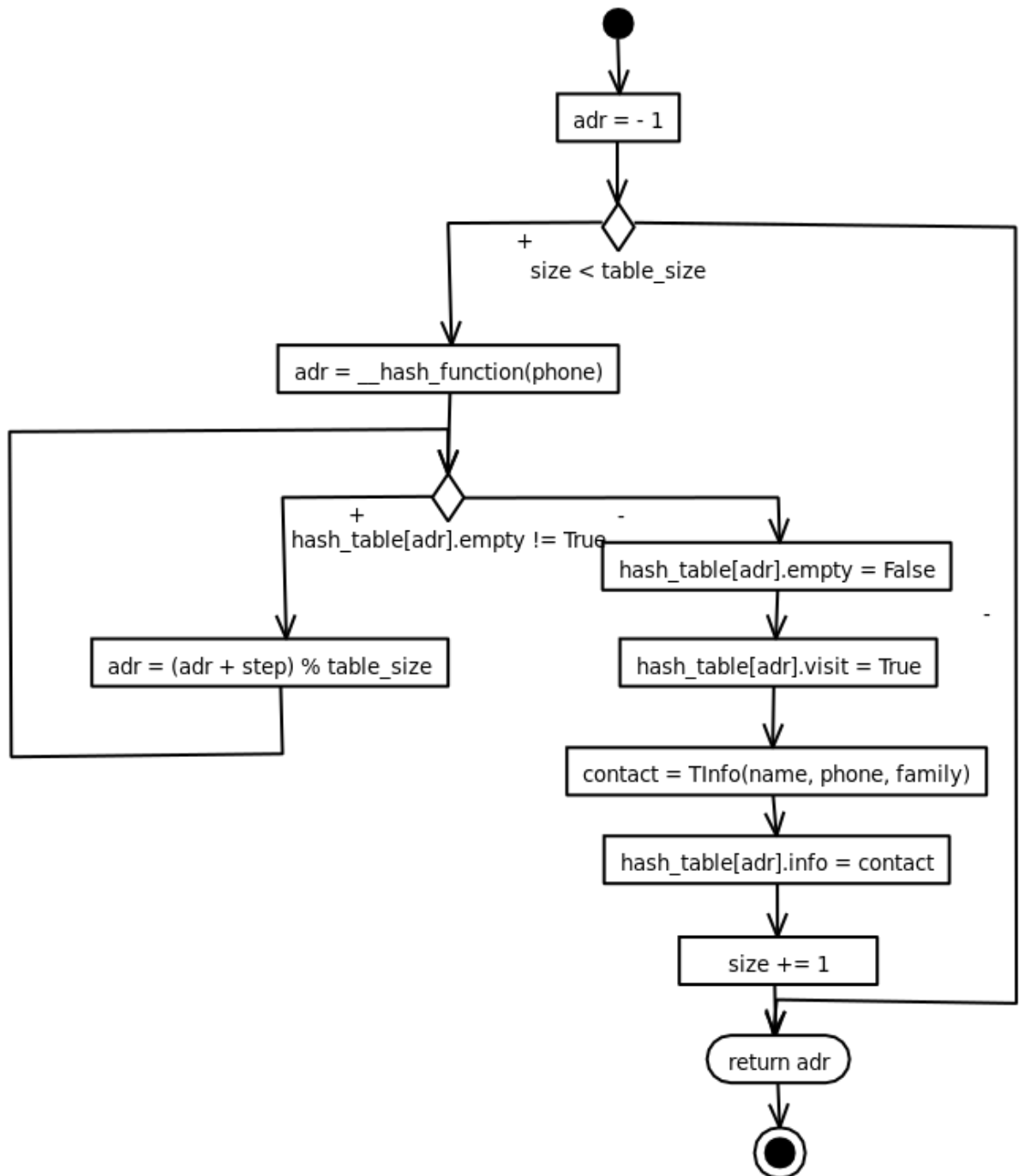


Рисунок 2 - Диаграмма деятельности для добавления элемента, в таблицу методом открытой адресации.

Для поиска элемента, надо убедиться, что флаги visit каждой ячейки сброшены к дефолтным значениям. Для этого мы используем функцию, код которой представлен в листинге 5.

Листинг 5. Сброс значений к дефолтным.

Диаграмма деятельности представлена для нее на рисунке 3.

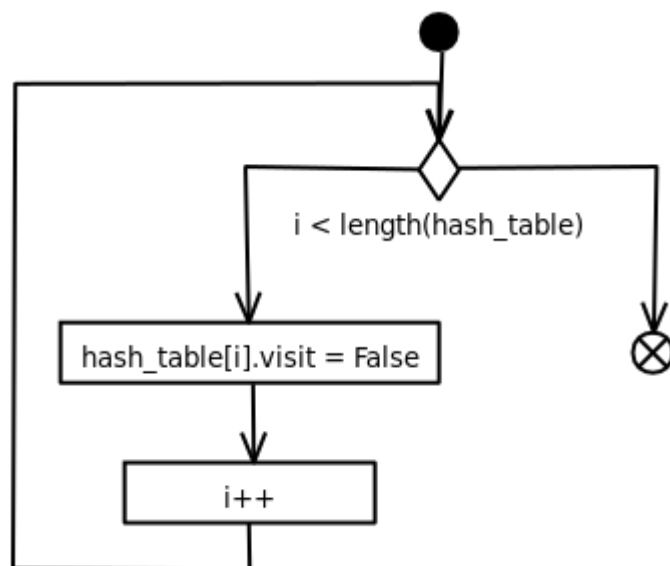


Рисунок 3 - Сброс флагов visit к дефолтным значениям

Функция поиска значения в таблице представлена в листинге 6.
Листинг 6. Поиск элемента в таблице.

Диаграмма деятельности для поиска элемента представлена на рисунке .

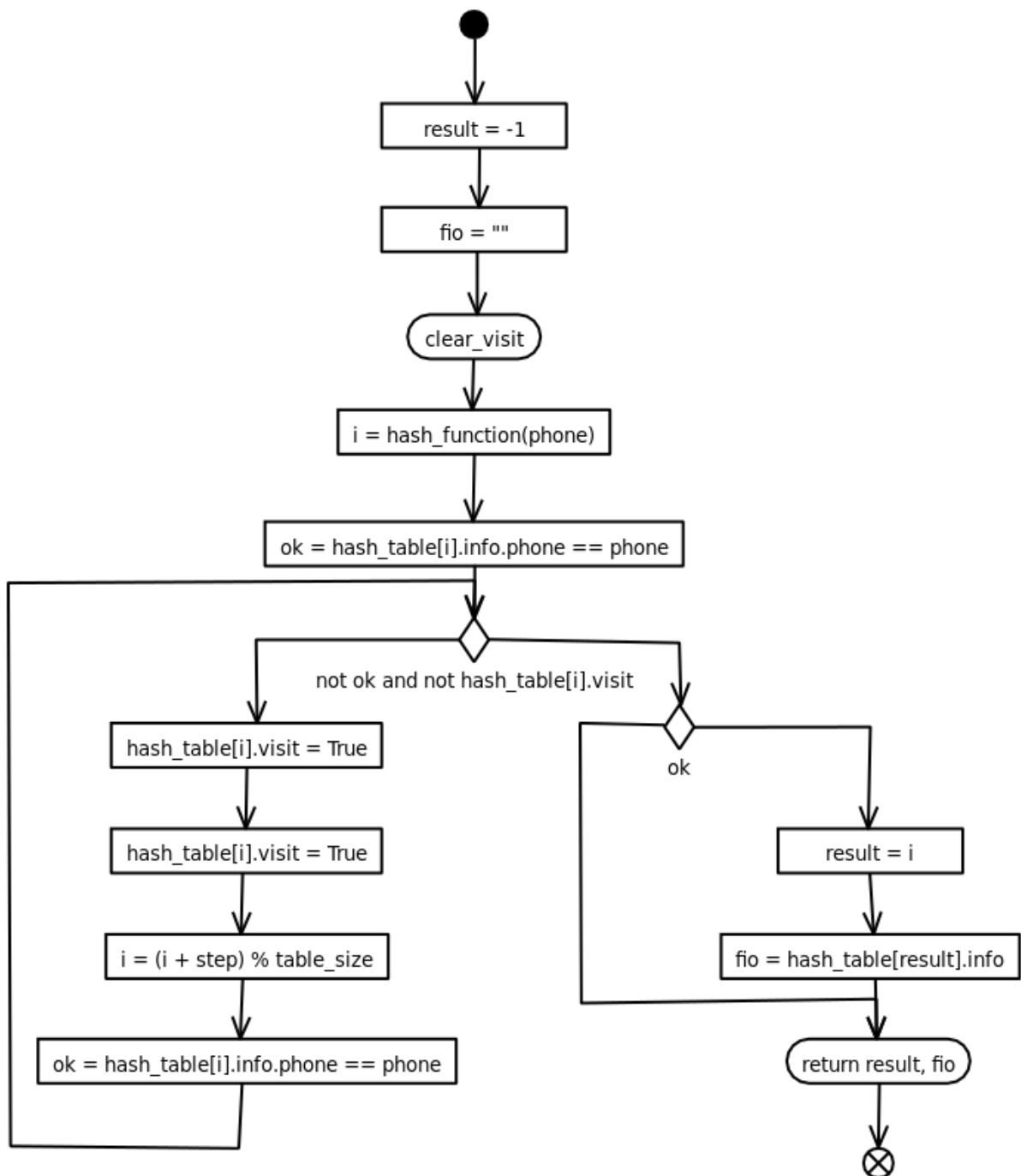


Рисунок 4 - Поиск элемента в хэш-таблице с открытой адресацией

Для удаления элемента реализован метод, код которого представлен в листинге . Действие кода сводится к нахождению нужного элемента и выставление флага empty в позицию True.

Листинг 7. Удаление элемента.

Диаграмма деятельности для этого представлена на рисунке 5.

					ИКСиС.09.03.02.220000 ПР	
						7

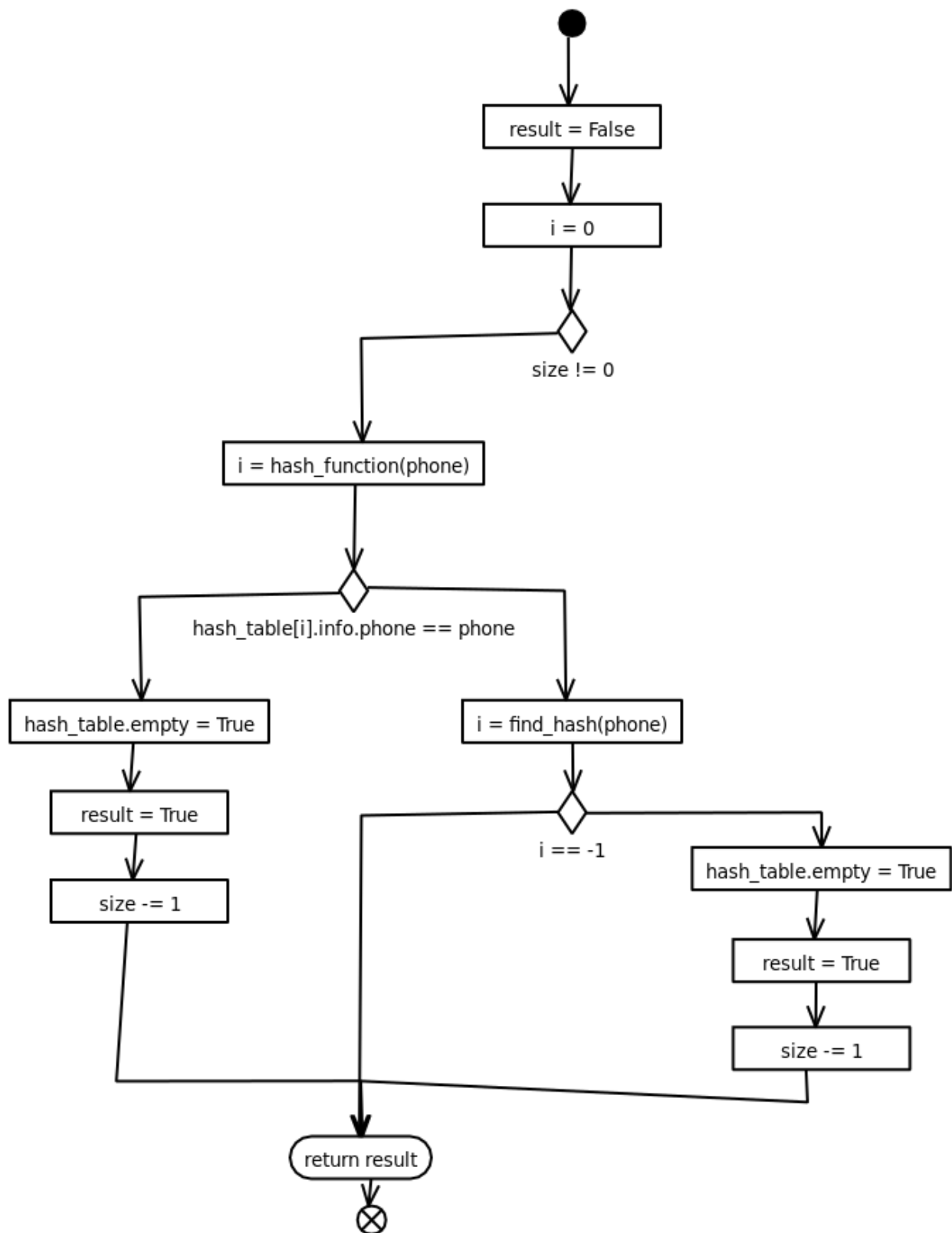


Рисунок 5 - Удаление элемента из хэш-таблицы

Так же реализуем хэш-таблиц по методу цепочек. Для этого определим классы данных, как в листинге 8.

Листинг 8. Классы данных для метода цепочек.

Реализацию функции для хэширования оставим без изменений.
Изменим функцию добавления нового значения (листинг 9) и ее диаграмма деятельности представлена на рисунке 6.

Листинг 9. Функция добавления новой записи в таблицу.

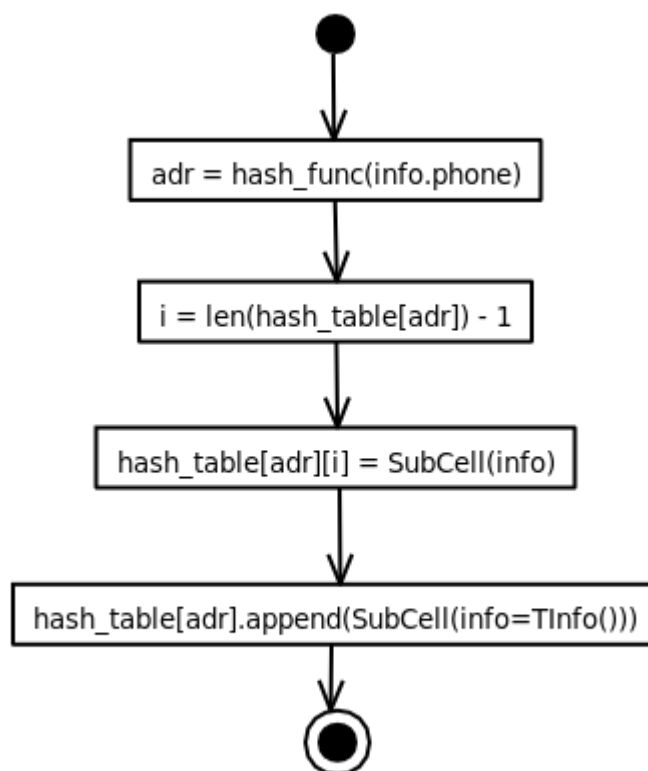


Рисунок 6 - Добавление нового элемента в таблицу

Функция удаления элемента представлена в листинге 10. Диаграмма деятельности для нее представлена на рисунке 7.

Листинг 10. Удаление элемента.

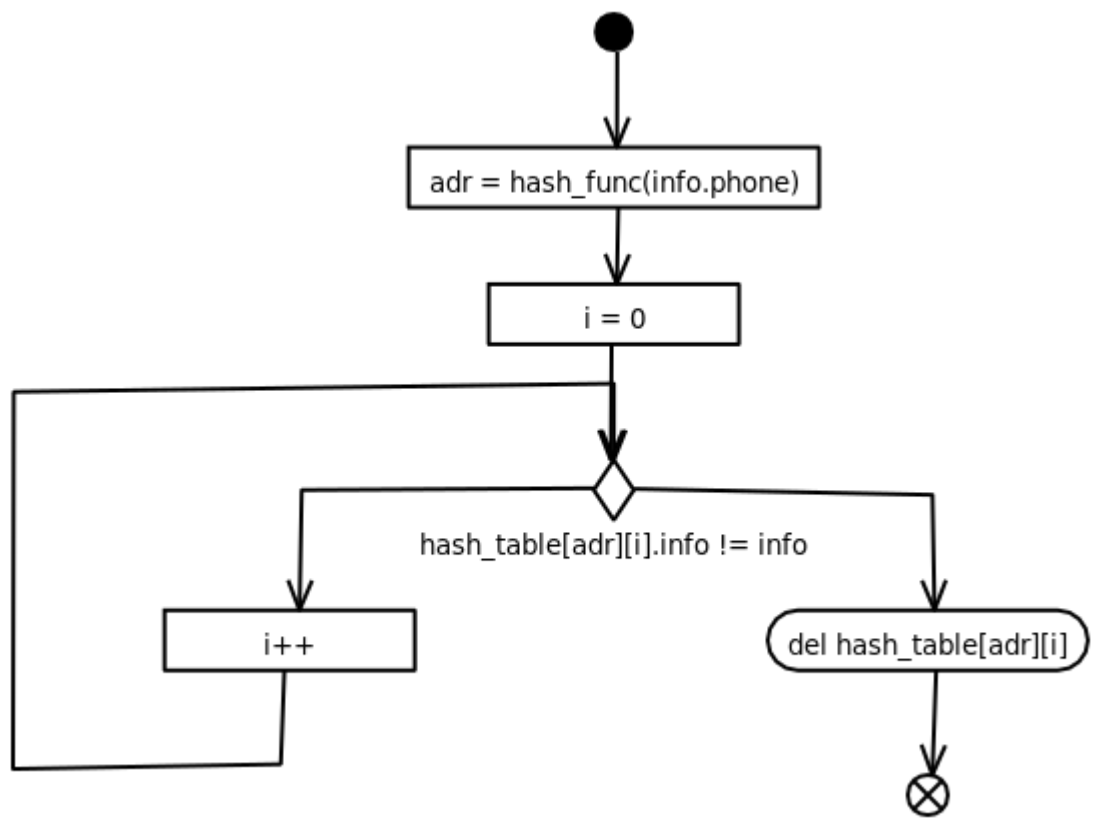


Рисунок 7 - Удаление элемента

Функция поиска элемента представлена в листинге 11. Диаграмма деятельности на рисунке 8.

Листинг 11. Функция поиска элемента.

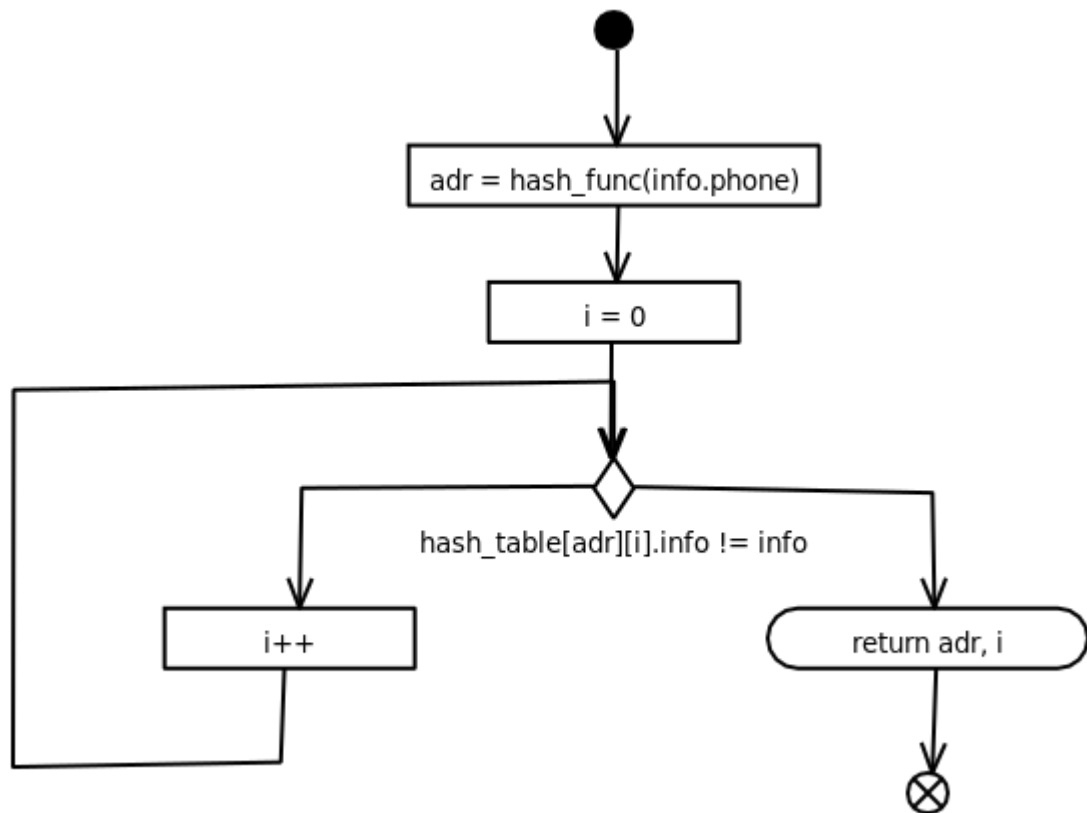


Рисунок 8 - Поиск элемента в хэш-таблице

Вывод: в ходе выполнения практической работы были изучены хэш-таблицы и методы их реализации на языке Python.