

## 1. Постановка задачи

Целью данной работы является реализация хеш-таблицы со следующими характеристиками:

- Тип ключа: строковый (вариант 1б)
- Метод хеширования: метод умножения (вариант 2б)
- Разрешение коллизий: метод внешних цепочек (вариант 3а)
- Размер таблицы: 16 элементов ( $2^4$ )

## 2. Теоретические сведения

### 2.1 Метод умножения для хеширования

Метод умножения использует формулу:  $h(k) = \lfloor M(kA \bmod 1) \rfloor$  где:

- $k$  - числовое представление ключа
- $A = (\sqrt{5}-1)/2 \approx 0.6180339887498949$  (золотое сечение)
- $M = 16$  (размер таблицы)

### 2.2 Преобразование строкового ключа

Для преобразования строкового ключа в число используется полиномиальная функция:  $k = \text{sum}(\text{ord}(\text{символ}) * (31^i))$  для каждого символа

### 2.3 Метод внешних цепочек

При возникновении коллизии новый элемент добавляется в связный список, хранящийся в соответствующей ячейке таблицы.

## 3. Практическая реализация

### 3.1 Результаты вставки ключей:

Ключ	Хеш-значение	Успех вставки	Коллизии
test1	7	Да	0
test2	13	Да	0
test3	2	Да	0
test4	8	Да	0
test5	14	Да	0
test6	4	Да	0
test7	10	Да	0

test1	7	Да	0
test8	0	Да	0
test2	13	Да	0

### 3.2 Тестирование поиска:

Ключ	Результат
test1	Найден по индексу 7, коллизий: 0
test2	Найден по индексу 13, коллизий: 0
test9	Не найден

## 4. Заключение

В ходе выполнения работы была реализована хеш-таблица с использованием метода умножения для хеширования и метода внешних цепочек для разрешения коллизий.

Тестирование показало, что:

- Реализованный алгоритм успешно обрабатывает как уникальные, так и повторяющиеся ключи
- Коллизии эффективно разрешаются с помощью метода внешних цепочек
- Поиск элементов работает корректно как для существующих, так и для отсутствующих ключей

Реализованная структура данных может быть использована для эффективного хранения и поиска данных со строковыми ключами.