Хэш-таблицы. Хэш-функции.

Горденко М.К.

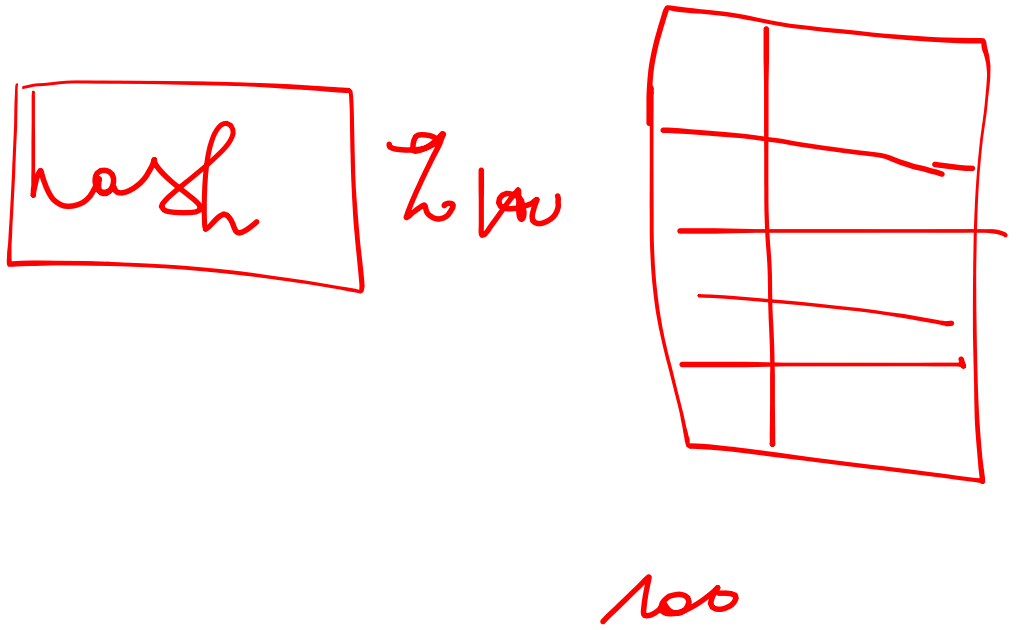
Асимптотика

Контейнер \ операция	insert	remove	find
Array	O(N)	O(N)	O(N)
List	O(1)	O(1)	O(N)
Sorted array	O(N)	O(N)	O(logN)
Бинарное дерево поиска	O(logN)	O(logN)	O(logN)
Хеш-таблица	O(1)	O(1)	O(1)

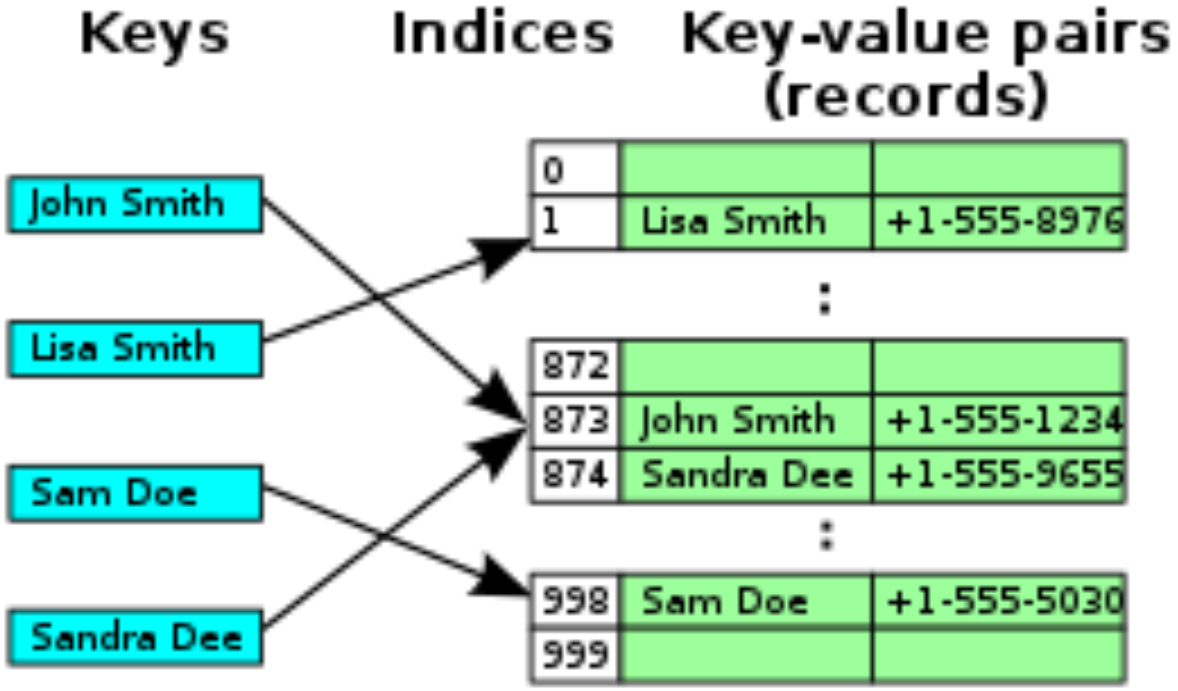
Хэш-функция

- Хеш-функция – отображает различные элементы из множества объектов на множество ключей за O(1) времени в худшем случае.

- операция добавления новой пары,
- операция поиска,
- операция удаления пары по ключу.



Хэш-таблица



len(apple)
len(hat)
len(a)
hello

O(1)

Парадокс дней рождений

- В группе, состоящей из 23 или более человек, вероятность совпадения дней рождения (число и месяц) хотя бы у двух людей превышает 50 %. Например, если в классе 23 ученика или более, то более вероятно то, что у какой-то пары одноклассников дни рождения придутся на один день, чем то, что у каждого будет свой неповторимый день рождения.

Лемма. Вероятность коллизий при вставке в хеш-таблицу превышает 50%

Que – 1. Given the following input (4322, 1334, 1471, 9679, 1989, 6171, 6173, 4199) and the hash function x mod 10, which of the following statements are true? (GATE CS 2004)

i. 9679, 1989, 4199 hash to the same value

ii. 1471, 6171 hash to the same value

iii. All elements hash to the same value

iv. Each element hashes to a different value

2 8 3 2 2 3 5 5

Que – 2. The keys 12, 18, 13, 2, 3, 23, 5 and 15 are inserted into an initially empty hash table of length 10 using open addressing with hash function h(k) = k mod 10 and linear probing. What is the resultant hash table?

0	
1	
2	2
3	23
4	
5	15
6	
7	
8	18
9	

(A)

0	
1	
2	12
3	13
4	
5	5
6	
7	
8	18
9	

(B)

0	
1	
2	12
3	13
4	2
5	3
6	23
7	5
8	18
9	15

(C)

0	
1	
2	12, 2
3	13, 3, 23
4	
5	5, 15
6	
7	
8	18
9	

(D)

Que – 3. A hash table of length 10 uses open addressing with hash function h(k)=k mod 10, and linear probing. After inserting 6 values into an empty hash table, the table is as shown below.

Which one of the following choices gives a possible order in which the key values could have been inserted in the table?

(A) 46, 42, 34, 52, 23, 33

(B) 34, 42, 23, 52, 33, 46

(C) 46, 34, 42, 23, 52, 33

(D) 42, 46, 33, 23, 34, 52

0	
1	
2	42
3	23
4	34
5	52
6	46
7	33
8	
9	

Хеширование в языках программирования

- Почти во всех современных языках присутствуют классы, реализующие хеширование. Рассмотрим некоторые из них.
- Java
  - HashMap — реализация интерфейса ассоциативного массива с использованием хеш-таблицы,
  - HashSet — реализация интерфейса множества с использованием хеш-таблицы,
  - LinkedHashMap — потомок класса HashMap. Позволяет просматривать значения в том порядке, в котором они были добавлены.
- C++
  - unordered_map — реализация интерфейса ассоциативного массива с использованием хеш-таблицы,
  - unordered_set — реализация интерфейса множества с использованием хеш-таблицы.
- Python (CPython)
  - dict — реализация интерфейса ассоциативного массива с использованием хеш-таблицы,
  - set — реализация интерфейса множества с использованием хеш-таблицы.

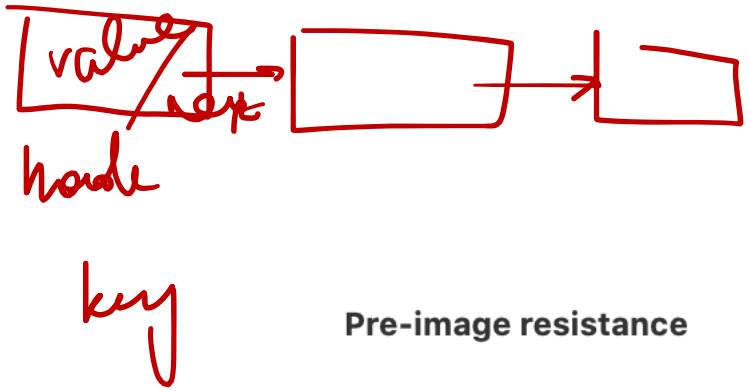
Идеальная хэш-функция

- Идеальная хеш-функция – отображает без коллизий различные элементы из множества объектов на множество ключей за O(1) времени в худшем случае.

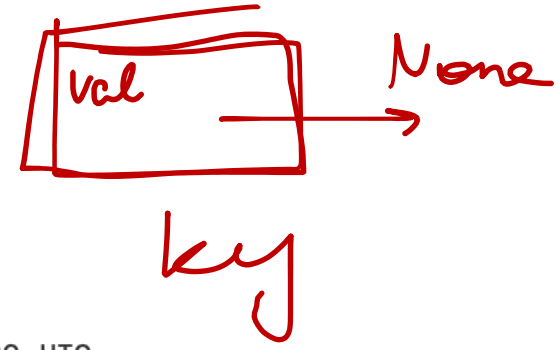
- Для идеальной хеш-функции выполняются следующие условия:

a) хеш-функция является детерминированной, то есть одно и то же сообщение приводит к одному и тому же хеш-значению
b) значение хеш-функции быстро вычисляется для любого сообщения
c) невозможно найти сообщение, которое дает заданное хеш-значение
d) невозможно найти два разных сообщения с одинаковым хеш-значением
e) небольшое изменение в сообщении изменяет хеш настолько сильно, что новое и старое значения кажутся некоррелирующими

https://neerc.ifmo.ru/wiki/index.php?title=Идеальное_хеширование



Свойства



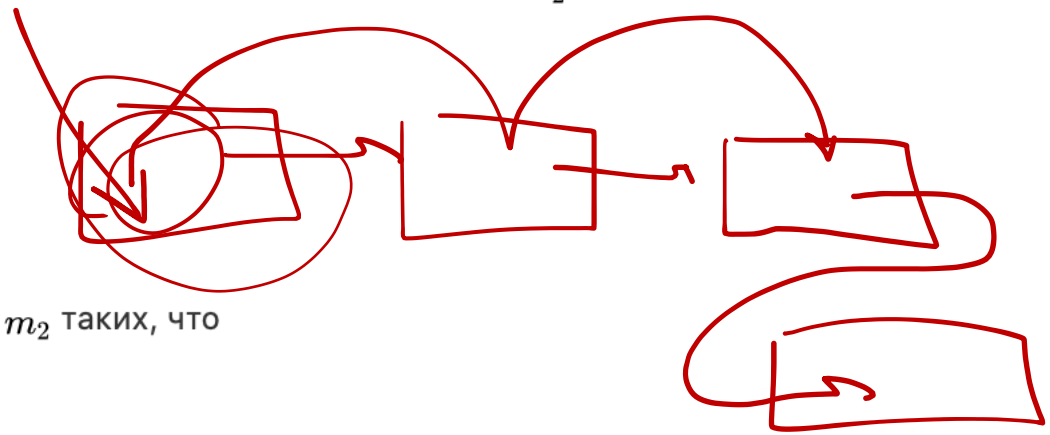
Pre-image resistance

Имея заданное значение *h*, должно быть сложно найти любое сообщение *m* такое, что
$h = hash(m)$

Second pre-image resistance

Имея заданное входное значение m_1 , должно быть сложно найти другое входное значение m_2 такое, что

$$hash(m_1) = hash(m_2)$$



Collision resistance

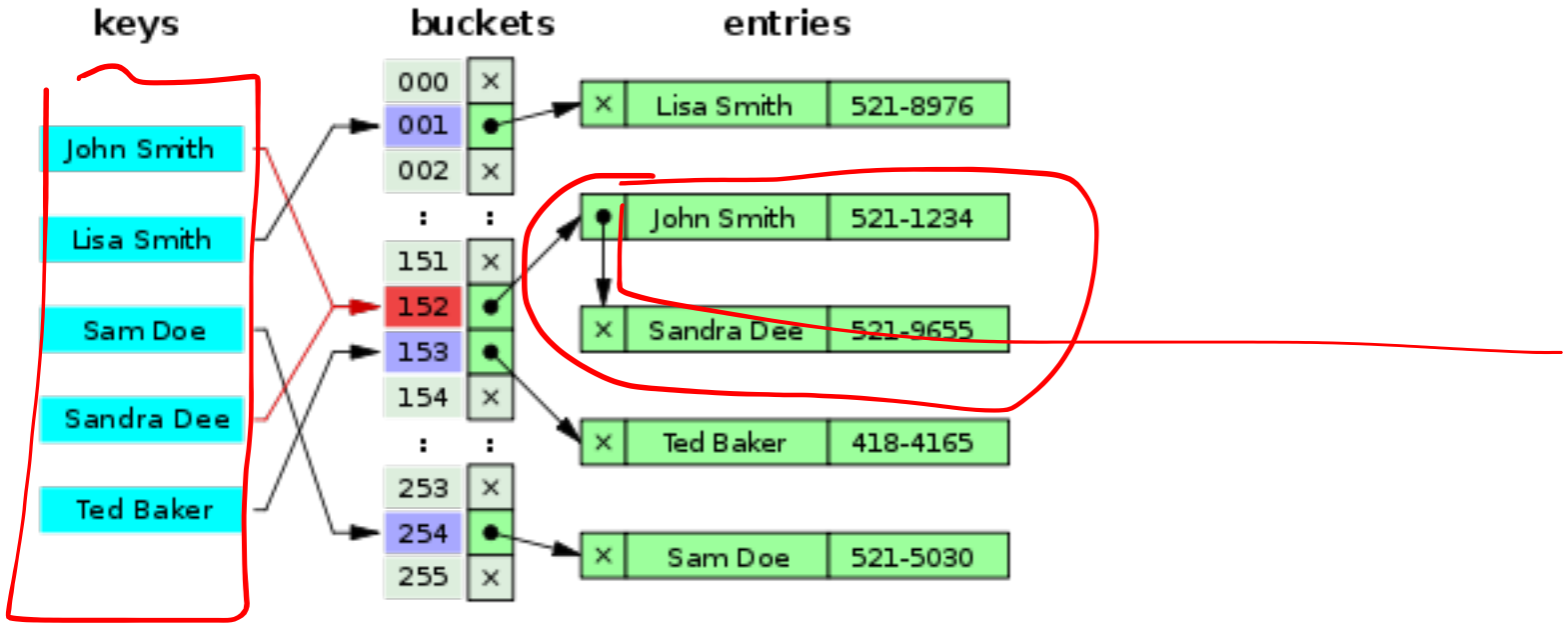
Должно быть сложно найти два различных сообщения m_1 *и* m_2 таких, что

$$hash(m_1) = hash(m_2)$$

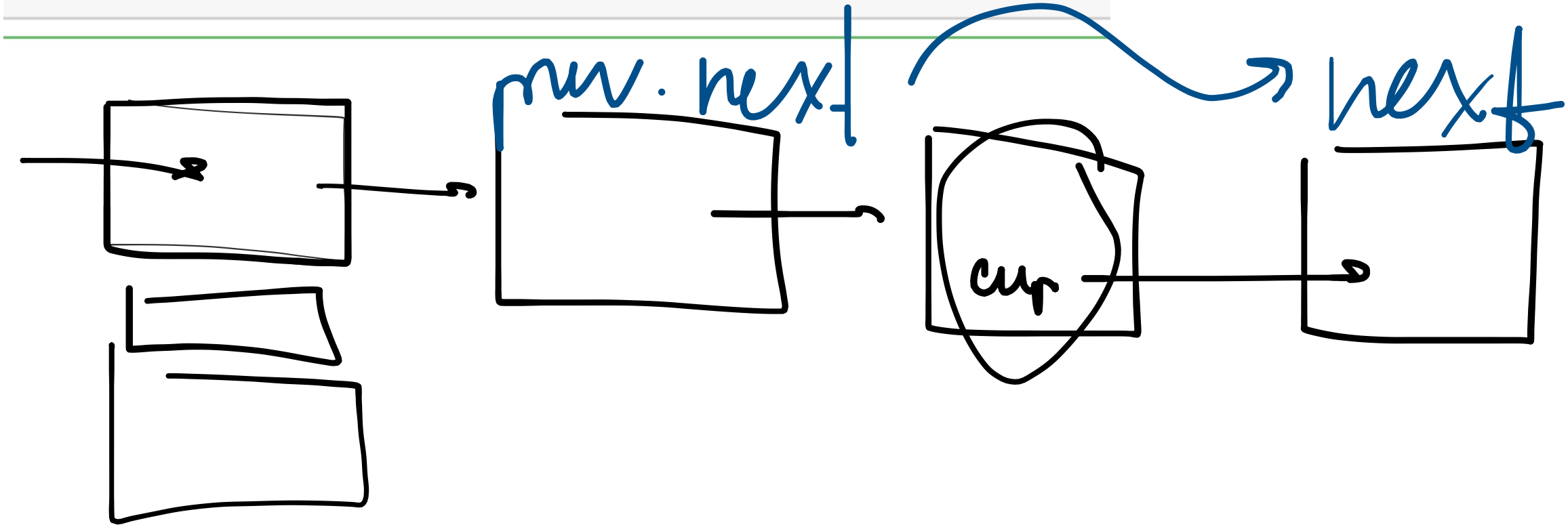
Такая пара сообщений m_1 *и* m_2 называется коллизией хеш-функции

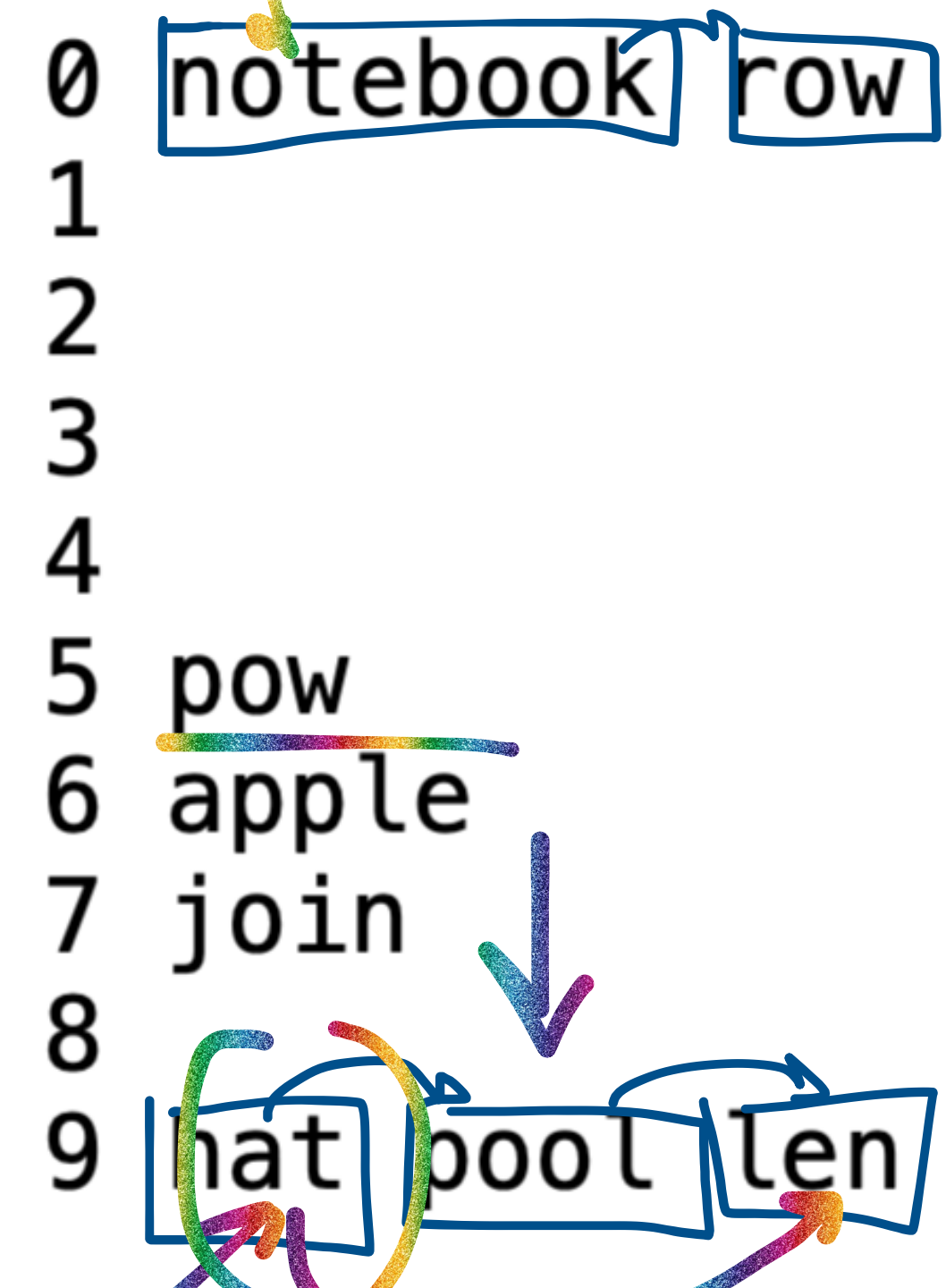
Разрешение коллизий

- Метод цепочек



```
  current = self.table[index]
  while current.Next: #None
      current = current.Next
  current.Next = Node(key, value)
  self.size += 1
```




0 notebook row
1
2
3
4
5 pow
6 apple
7 join
8
9 hat pool len

Разрешение коллизий

- Открытая адресация