

CSC212
Tutorial 12
Hashing

Problem 1: Use the hash function $H(\text{key}) = \text{key} \% 7$ to store the following sequence of keys in a hash table: **28, 71, 43, 11, 56, 17, 63, 6, 7**. If there is no place for some keys in the table indicate that the table is full.

1. Use linear rehashing (take $c = 1$), and show the number of probes
2. Use external chaining
3. Use coalesced chaining with a cellular size of 3. Show the final location of *epla*

Solution:

1.

Linear Rehashing

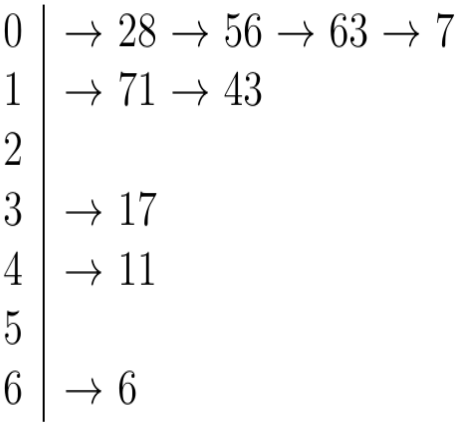
| | Hash | Probs |
|---|------|-------|
| 0 | 28 | 1 |
| 1 | 71 | 1 |
| 2 | 43 | 2 |
| 3 | 56 | 4 |
| 4 | 11 | 1 |
| 5 | 17 | 3 |
| 6 | 63 | 7 |

(No space for 6 and 7)

| Key | 28 | 71 | 43 | 11 | 56 | 17 | 63 |
|------------------|----|----|----|----|----|----|----|
| Position | 0 | 1 | 2 | 4 | 3 | 5 | 6 |
| Number of Probes | 1 | 1 | 2 | 1 | 4 | 3 | 7 |

2.

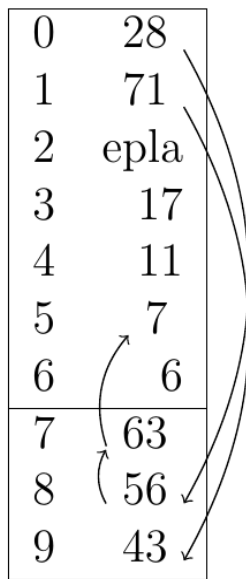
External Rehashing



| | | | | | | | | | |
|-------------------------------------|----|----|----|----|----|----|----|---|---|
| Key | 28 | 71 | 43 | 11 | 56 | 17 | 63 | 6 | 7 |
| Position of the key within the list | 1 | 1 | 2 | 1 | 2 | 1 | 3 | 1 | 4 |
| Index of the list | 0 | 1 | 1 | 4 | 0 | 3 | 0 | 6 | 0 |

3.

Coalesced chaining



| | | | | | | | | | |
|-----------------------|----|----|----|----|----|----|----|----|----|
| Key | 28 | 71 | 43 | 11 | 56 | 17 | 63 | 6 | 7 |
| Position | 0 | 1 | 9 | 4 | 8 | 3 | 7 | 6 | 5 |
| Index of next element | 8 | 9 | -1 | -1 | 7 | -1 | 5 | -1 | -1 |

Problem 2: Assuming the keys are integers, denoted by $d_n d_{n-1} \dots d_k \dots d_2 d_1$, where d_i is the i -th decimal digit in the key, d_n being the leftmost decimal digit.

The hash function $H(key)$ is given by: $H(key) = (d_1 d_2 + d_{n-1} d_n + d_k) \bmod 11$

where $d_1 d_2$ is a two-digit number (composed by swapping the rightmost two digits) $d_{n-1} d_n$ is also a two-digit number (composed by swapping the leftmost two digits), and $k = \lceil n/2 \rceil$

Example: $H(70934) = (43 + 07 + 9) \bmod 11 = 59 \bmod 11 = 4$

Assume the keys are: 1234, 519, 911, 7346, 0, 999, 99834, 54 and 40015.

- Compute $H(key)$ for each of the above keys
- Insert the above keys (in the same order) in a hash table with open addressing (linear rehashing)
- Find the number of probes required to search for keys 54 and 11 in the above hash table

Solution:

- $H(1234) = (43 + 21 + 3) \% 11 = 67 \% 11 = 1$
- $H(519) = (91 + 15 + 1) \% 11 = 107 \% 11 = 8$
- $H(911) = (11 + 19 + 1) \% 11 = 31 \% 11 = 9$
- $H(7346) = (64 + 37 + 4) \% 11 = 105 \% 11 = 6$
- $H(0) = (0 + 0 + 0) \% 11 = 0 \% 11 = 0$
- $H(999) = (99 + 99 + 9) \% 11 = 207 \% 11 = 9$
- $H(99834) = (43 + 99 + 8) \% 11 = 150 \% 11 = 7$
- $H(54) = (45 + 45 + 4) \% 11 = 94 \% 11 = 6$
- $H(40015) = (51 + 04 + 0) \% 11 = 55 \% 11 = 0$

| 0 | 0 | 1 |
|----|-------|---|
| 1 | 1234 | 1 |
| 2 | 54 | 8 |
| 3 | 40015 | 4 |
| 4 | empty | 0 |
| 5 | empty | 0 |
| 6 | 7346 | 1 |
| 7 | 99834 | 1 |
| 8 | 519 | 1 |
| 9 | 911 | 1 |
| 10 | 999 | 2 |

Number of probes to search for 54 is 8, and for 11 ($H(11) = (11+11+1)\%11 = 23\%11 = 1$) it's 4.

Problem 3: Develop a hashing function to convert a character key k of 15 characters into integers in the range of 0 to 999.

Solution:

```
public static int charHash(char[] k) {  
    int hash = 0;  
    for (char c : k)  
        hash += c;  
    return hash %= 1000;  
}
```