

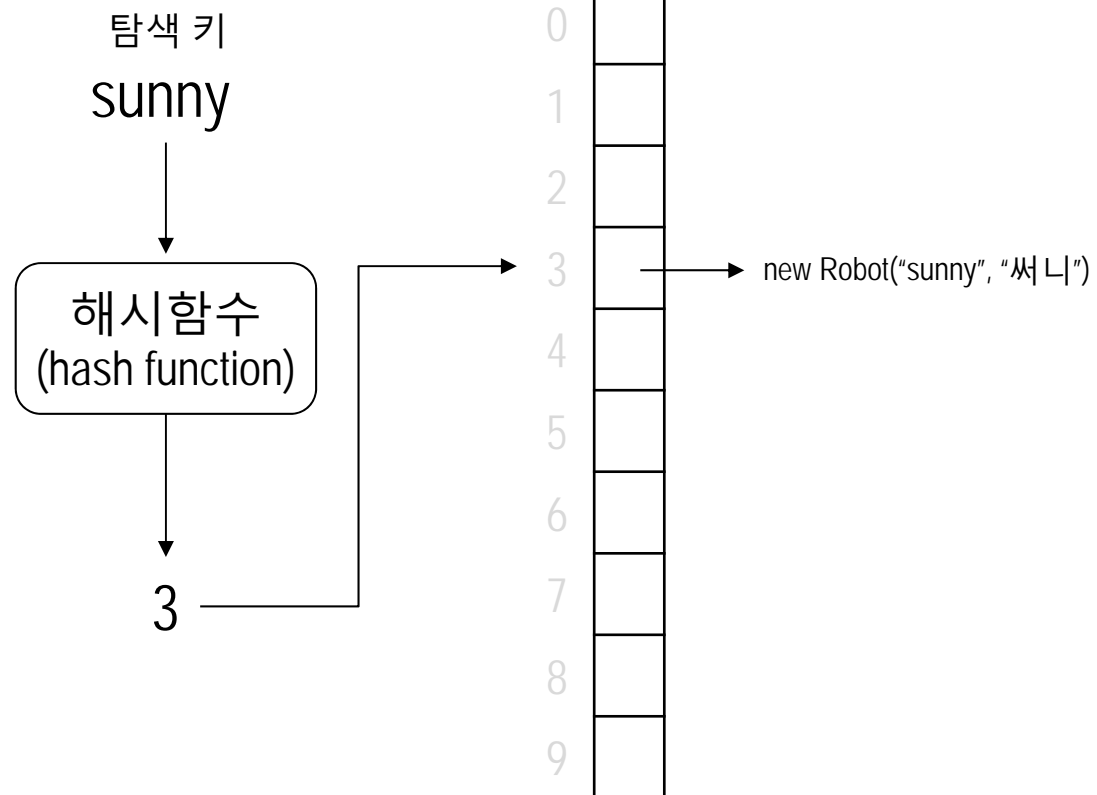
# 해싱

# 해시테이블 (hash table)

Robot

- id: sunny
- name: 써니

```
public class Robot {  
    String id;  
    String name;  
    public Robot(String id, String name) {  
        this.id=id;  
        this.name=name;  
    }  
}
```



hashTable[3]= new User("sunny", "써니")

# 해시 함수 (hash function)

ASCII code 십진수

- s: 115
- u: 117
- n: 110
- n: 110
- y: 121

탐색 키  
sunny

해시 함수 (hash function)

$$115 + 117 + 110 + 110 + 121 = 573$$
$$573 \% 10 = 3$$

3

해시 테이블  
(hash table)

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

# 해시 함수: 자바 hashCode()

```
public class Test {  
    public static void main(String[] args) {  
        String id;  
        id="sunny";  
        System.out.println(id.hashCode());  
        id="universe";  
        System.out.println(id.hashCode());  
    }  
}
```

실행결과  
109799703  
-290305467

탐색 키  
sunny

해시 함수 (hash function)

```
int hash(Object key) {  
    return key.hashCode() % 10;  
}
```

$109799703 \% 10 = 3$

3

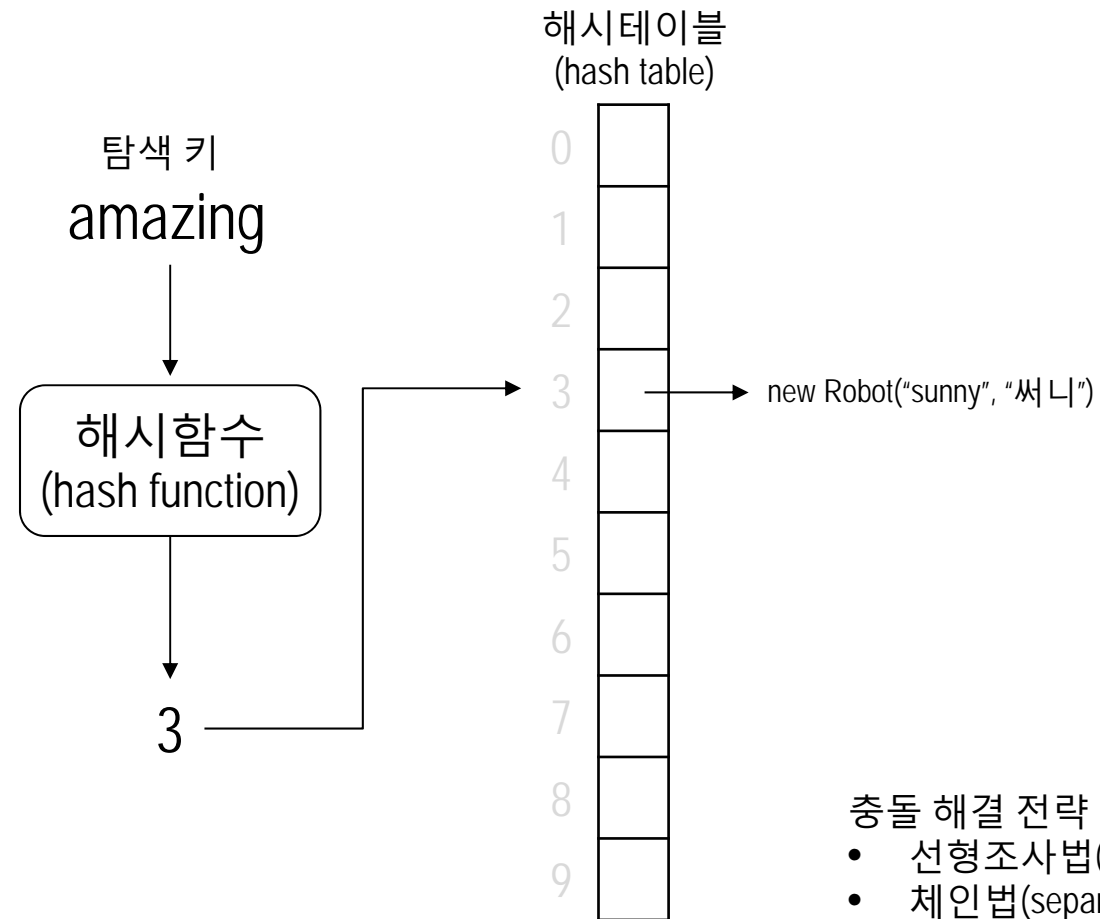
해시 테이블  
(hash table)

0	
1	
2	
3	
4	
5	
6	
7	
8	
9	

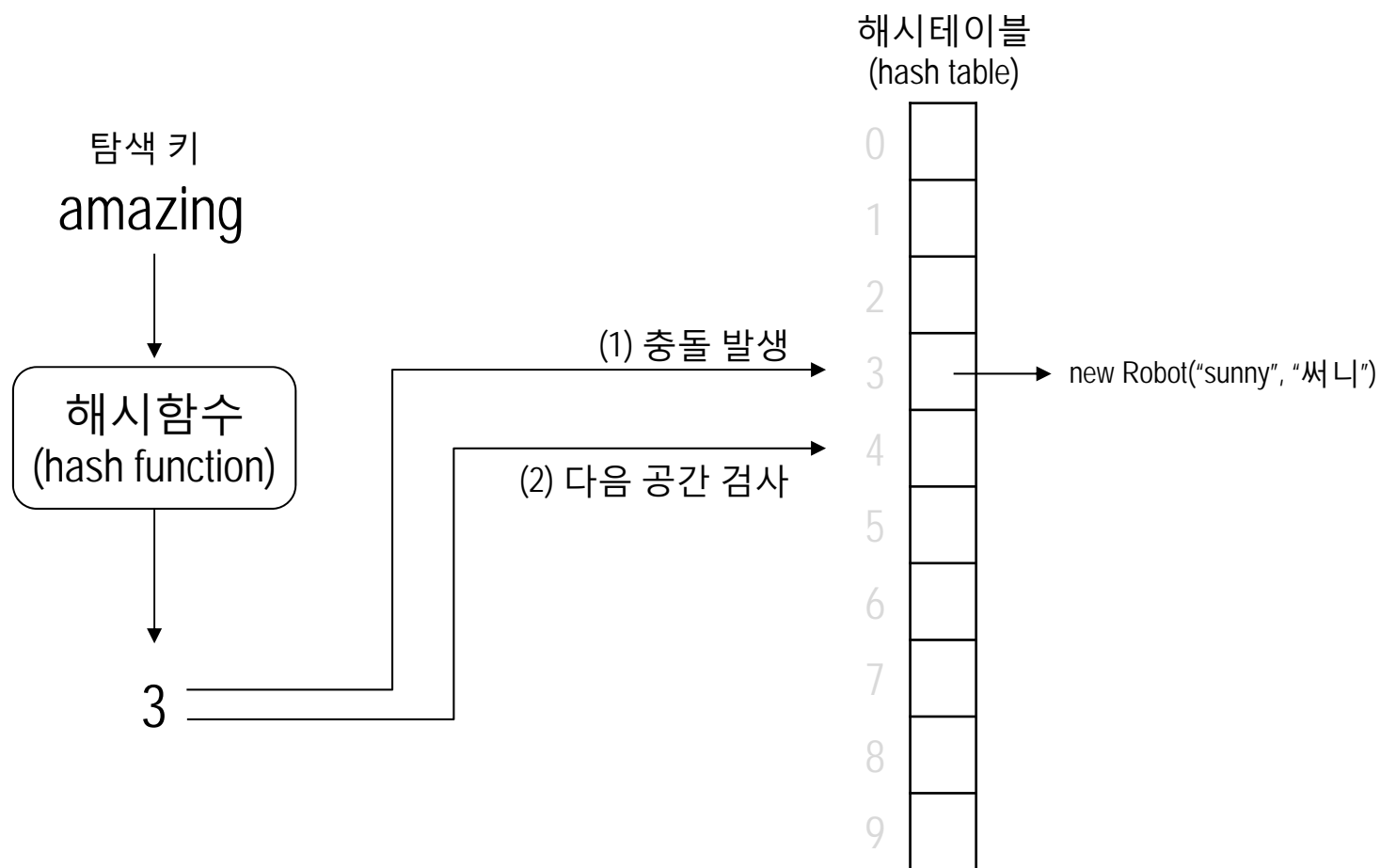
hashCode() 음수 처리 (예)

- $(\text{key.hashCode()} \& 0x7FFFFFFF) \% 10$
- Reference: <https://algs4.cs.princeton.edu/34hash/SeparateChainingHashST.java.html>

# 충돌 (collision)



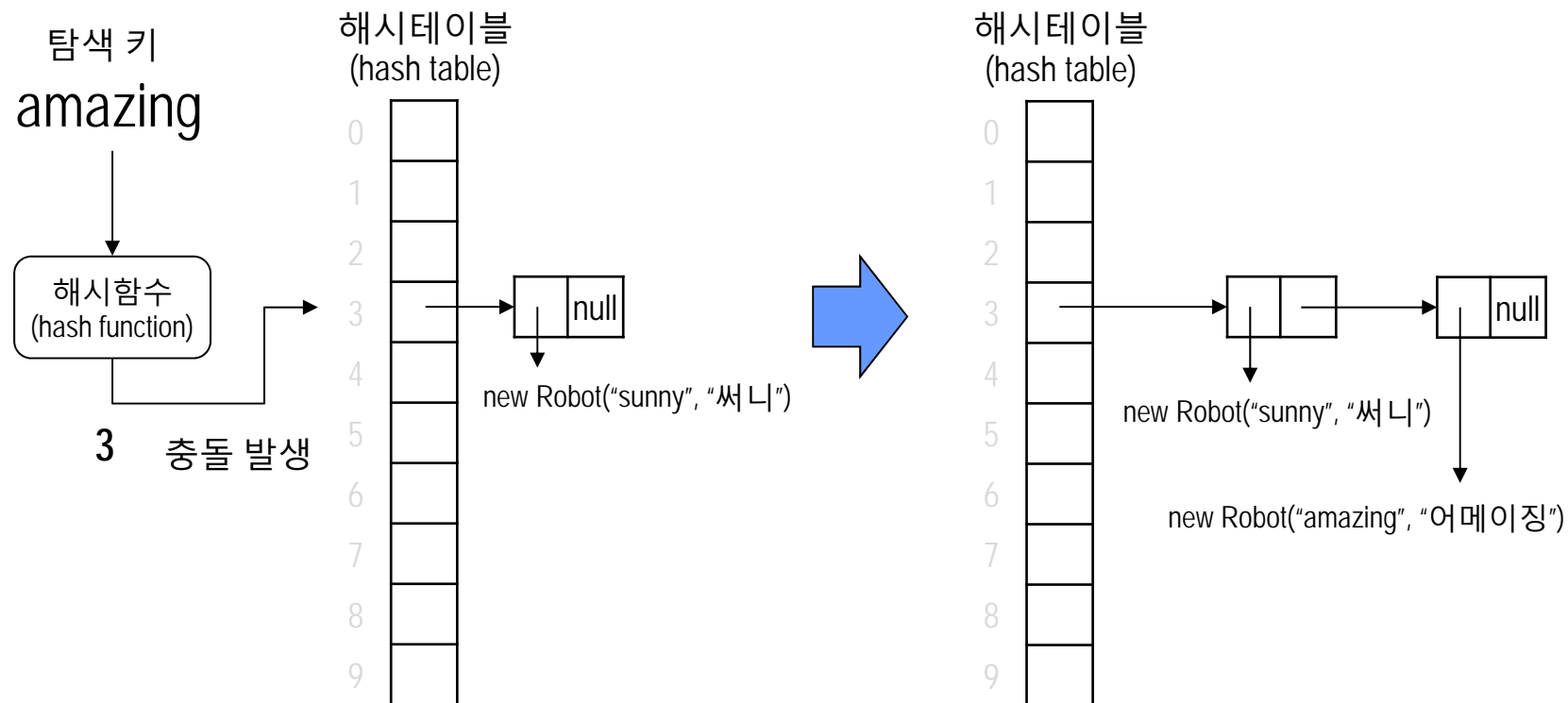
# 충돌 해결 전략: 선형조사법



# 충돌 해결 전략: 체인법

해시테이블의 각 버킷을 연결리스트로 만들어 두고  
충돌 발생 시 해당 연결리스트에 추가

```
LinkedList<Object> hashTable[] = new LinkedList[10];  
for (int i = 0; i < hashTable.length; i++) hashTable[i] = new LinkedList<>();
```



# 해시테이블 시간복잡도



Reference: [https://en.wikipedia.org/wiki/Hash\\_table](https://en.wikipedia.org/wiki/Hash_table)

## 해시테이블 시간복잡도

### 삽입, 삭제, 탐색 시간복잡도

◆ Average case:  $O(1)$

◆ Worst case:  $O(n)$



## References

- ✚ C로 쓴 자료구조론 (Fundamentals of Data Structures in C, Horowitz et al.). 이석호 역. 사이텍미디어. 1993.
- ✚ 쉽게 배우는 알고리즘: 관계 중심의 사고법. 문병로. 한빛아카데미. 2013.
- ✚ C언어로 쉽게 풀어 쓴 자료구조. 천인국 외 2인. 생능출판사. 2017.
- ✚ 프로그래밍 콘테스트 챌린징, Akiba 등 공저, 로드북, 2011.
- ✚ <https://introcs.cs.princeton.edu/>
- ✚ Introduction to Algorithms, Cormen et al., 3rd Edition (The MIT Press)