

# Week 12 - Hashing

- 주의 사항: 부정행위 금지(채점서버 외 인터넷 사용금지), STL 사용금지 (string, vector는 사용 가능)
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

## 문제 1

해시 테이블의 배열에 저장된 엔트리( $E_{key}$ ,  $E_{value}$ )의 값( $E_{value}$ )을 차례대로 출력해보자. 배열의 크기가  $P$  ( $2 \leq P \leq 1,000$ )인 해시 테이블을 작성하고, 해시 함수를 통해 해시 테이블에 엔트리를 삽입한다. 그리고 해시 테이블의 배열의 처음부터 끝까지 모든 엔트리의 값( $E_{value}$ )을 차례대로 출력한다. 이 때 이 해시 테이블은 다음 조건을 만족한다.

1. 충돌이 일어날 경우, 선형 조사법(Linear Probing)을 사용하여 충돌을 처리한다.
2. 해시 함수는 그림 1의 해시 함수를 사용하기로 한다. 그림 1의 해시 함수는 입력 받은 자연수  $key$  ( $1 \leq key \leq 1,000$ )를 배열의 크기  $P$ 로 나머지 연산하는 함수이다.
3. 해시 테이블 생성 시 모든 배열의 엔트리의 키( $E_{key}$ )와 값( $E_{value}$ )은 -1로 초기화한다.

```
int hashfunc(int key)
{
    return key % P;
}
```

그림 1. 해시 함수

# 입력

- 1. 첫 번째 줄에는 테스트 케이스의 수  $T$ 가 주어진다. ( $1 \leq T \leq 1,000$ )
- 2. 두 번째 줄부터, 다음이  $T$ 번 반복된다.
  - 1) 해시 테이블의 배열의 크기  $P$  ( $2 \leq P \leq 1,000$ )가 주어진다.
  - 2) 해시 테이블에 삽입할, 엔트리의 키( $E_{key}$ )와 값( $E_{value}$ )이 될 자연수  $key$  ( $1 \leq key \leq 1,000$ )의 개수  $Q$  ( $1 \leq Q < P$ )가 주어진다. (따라서,  $key = E_{key} = E_{value}$ )
  - 3)  $Q$ 개의 자연수  $key$ 가 빈칸을 사이에 두고 주어진다. (단, 값은 중복되지 않는다.)

# 출력

매 테스트 케이스마다 주어진 해시 테이블의 배열의 처음부터 끝까지 각 배열의 엔트리의 값( $E_{value}$ )을 빈 칸을 두고 차례대로 출력한다.

## 예제 입출력

예제 입력	예제 출력
2	-1 1 -1 -1 -1 -1 6 7 20 33 8 -1 12
13	-1 1 -1 -1 55 -1 -1 -1 42 26 10 27 9 -1 31 -1 16
7	
7 20 33 1 6 12 8	
17	
9	
26 10 27 42 9 16 31 55 1	