

## 집합 연산 (3)

두 집합  $A, B$ 가 있을 때,  $A$  와  $B$ 에 모두 속하는 원소의 집합을  $A$  와  $B$ 의 교집합이라고 하고,  $A \cap B$ 로 나타낸다. 또한 집합  $A$ 에 속하거나  $B$ 에 속하는 원소의 집합을  $A$  와  $B$ 의 합집합이라고 하고,  $A \cup B$ 로 나타낸다. 그리고, 집합  $A$ 에는 속하지만 집합  $B$ 에는 속하지 않는 모든 원소의 집합을  $A$  와  $B$ 의 차집합이라고 하고,  $A - B$ 로 나타낸다.

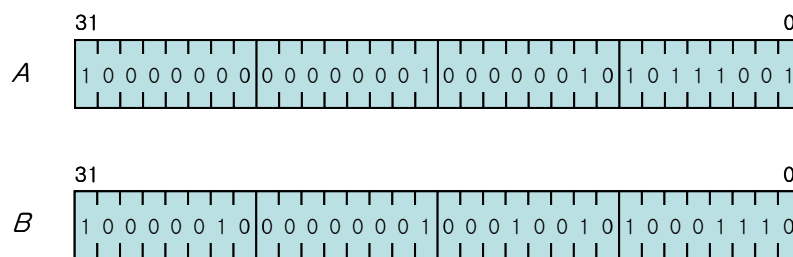
자연수  $n$ 에 대하여,  $0 \sim (n-1)$  사이의 정수를 원소로 하는 집합을 표현하는 방법 중의 하나는  $n$ -비트 배열(혹은  $n$ -비트 벡터)을 사용하는 것이다. 정수  $k$  ( $0 \leq k < n$ )가 어떤 집합에 속할 경우에는 그 집합을 표현하는 비트 배열의  $k$ -번째 비트를 1로 만들고, 그렇지 않는 경우에는  $k$ -번째 비트가 0이 되도록 만든다. 이렇게 집합의 원소를 한 개의 비트로 표현하는 비트 배열은 매우 작은 메모리를 사용하고서도 집합을 표현할 수 있는 장점이 있으며, 그 집합에 원소를 새로 추가하거나 제거하는 연산을 특정 비트의 값을 1로 만들거나 0으로 만드는 연산을 통하여 쉽게 구현할 수 있다. 또한, 비트 배열로 표현된 두 집합의 교집합, 합집합 등의 집합 연산을 프로그래밍 언어에서 제공하는 비트 연산자를 사용하여 쉽게 구현할 수 있다.

예를 들어,  $0 \sim 31$  사이의 정수를 원소로 하는 집합  $A$  와  $B$ 가 다음과 같이 주어졌을 때,

$$A = \{0, 3, 4, 5, 7, 9, 16, 31\}$$

$$B = \{1, 2, 3, 7, 9, 10, 12, 16, 25, 31\}$$

이 두 집합을 32-비트 배열로 표현하면 다음과 같다. 여기에서 32-비트의 가장 오른쪽(가장 낮은 자리) 비트를 0-번째 비트로 정하고, 가장 왼쪽(가장 높은 자리) 비트를 31-번째 비트로 정한다.



이와 같은 32-비트 배열은 C 프로그래밍 언어에서는 “unsigned int” 데이터 타입으로 정의할 수 있으며, 32-비트 배열로 표현된 두 집합의 교집합과 합집합의 연산은 각각 비트 연산자 ‘&’와 ‘|’로 수행할 수 있다. 또한, 32-비트 이상의 비트가 필요한 집합은 C 프로그래밍 언어에서 “unsigned int” 데이터 타입의 배열로서 구현할 수 있다.

$0 \sim 131$  사이의 정수를 원소로 하는 두 개의 집합이 주어졌을 때, 이 집합의 교집합, 합집합, 차집합을 비트 배열을 사용하여 구현하는 프로그램을 작성하시오.

## 입력

입력 파일의 이름은 “input.txt” 이다. 입력은  $t$  개의 테스트 케이스로 주어진다. 입력 파일의 첫 번째 줄에 테스트 케이스의 개수를 나타내는 정수  $t$  가 주어진다. 두 번째 줄부터 두 줄마다 한 개의 테스트 케이스에 해당하는 데이터가 입력되는데, 각 줄에는 한 개의 집합을 나타내는 데이터가 입력된다. 각 줄에서 첫 번째로 입력되는 정수  $m$  ( $0 \leq m \leq 132$ )은 집합에 속하는 원소의 개수를 나타내는 정수이다. 그 다음에는 그 집합에 속하는  $m$  개의 정수가 입력되는데, 이 정수들은 모두  $0 \sim 131$  사이의 정수이며, 같은 정수는 두 번 이상 입력되지 않는다. 각 정수들 사이에는 한 개의 공백이 있으며, 잘못된 데이터가 입력되는 경우는 없다.

## 출력

출력은 표준출력(standard output)을 사용한다. 입력되는 테스트 케이스의 순서대로 다음 줄에 이어서 각 테스트 케이스의 결과를 출력한다. 각 테스트 케이스의 첫 번째 줄에 입력되는 집합을  $A$  로 나타내고, 두 번째 줄에 입력되는 집합을  $B$  로 나타낼 때, 각 테스트 케이스에 해당하는 출력의 첫 줄에는 입력되는 두 집합의 교집합( $A \cap B$ )을, 두 번째 줄에는 합집합( $A \cup B$ )을, 세 번째 줄에는 차집합( $A - B$ )에 관한 데이터를 출력한다. 각 줄에 출력되는 집합을 나타내는 데이터에서 첫 번째로는 그 집합에 속하는 원소의 개수를 나타내는 정수를 출력하고, 그 다음에는 그 집합에 속하는 원소인 모든 정수를 출력한다. 단, 각 집합에 속하는 원소들은 작은 수부터 큰 수까지 오름차순으로 출력한다. 각 정수들 사이에는 한 개의 공백을 둔다.

## 입력과 출력의 예

입력	출력
4	3 3 7 9
5 5 3 7 9 4	8 1 2 3 4 5 7 9 10
6 1 9 7 2 3 10	2 4 5
5 2 4 6 8 10	0
5 9 7 5 3 1	10 1 2 3 4 5 6 7 8 9 10
10 10 9 8 7 6 5 4 3 2 1	5 2 4 6 8 10
5 1 2 3 4 5	5 1 2 3 4 5
5 1 2 3 4 5	10 1 2 3 4 5 6 7 8 9 10
0	5 6 7 8 9 10
	0
	5 1 2 3 4 5
	5 1 2 3 4 5