

# Week 1

- 주의 사항: 부정행위 금지, STL 사용 금지, 인터넷 금지, 배열(array)을 이용하여 구현할 것.
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

## 문제 1

<i>Arr</i>	0	1	4	0	7	0	3	12	8	0	3	11
<i>index</i>	0	1	2									$N - 1$

정수  $Z$  ( $0 \leq Z \leq 10,000$ ) 를 저장할 수 있는 크기가  $N$  ( $5 \leq N \leq 10,000$ )인 배열 *Arr*을 생성해보자. 배열을 생성할 때 배열 내의 모든 값은 0으로 초기화된다. 배열을 생성한 후, 명령어를 입력 받고 명령어에 따른 기능을 실행한다. 이 때 입력 받을 명령어가 다음과 같이 주어진다.

- **at(*i*):** 배열 *Arr*의 인덱스 *i*에 저장된 값을 출력한다. (단,  $0 \leq i \leq N - 1$ )
- **add(*i*, *value*):** 배열 *Arr*의 인덱스 *i*에서  $N - 2$ 의 값을 오른쪽으로 한 칸씩 이동한다. 그리고 배열 *Arr*의 인덱스 *i*번째에 *value*를 삽입한다. (단,  $0 \leq i \leq N - 1$ )
- **remove(*i*):** 배열 *Arr*의 인덱스  $i + 1$ 에서  $N - 1$ 까지의 값을, 왼쪽으로 한 칸씩 이동시킨다. 그리고 배열 *Arr*의  $N - 1$ 번째 인덱스의 값을 0으로 할당한다. (단,  $0 \leq i \leq N - 1$ )
- **set(*i*, *value*):** 배열 *Arr*의 인덱스 *i*번째에 *value*를 할당하고 *value*를 출력한다. (단,  $0 \leq i \leq N - 1$ )
- **print():** 배열 *Arr*의 모든 값을 공백으로 구분지어 차례대로 출력한다.
- **find\_max():** 배열 *Arr*의 모든 값 중 가장 큰 수를 출력한다.

## 입력

첫 번째 줄에 배열 *Arr*의 크기  $N$  ( $5 \leq N \leq 10,000$ )과 테스트 케이스 수  $T$  ( $1 \leq T \leq 1,000$ )가 차례대로 주어진다. 두 번째 줄부터  $T$ 개의 줄에 명령어가 하나씩 주어진다.

## 출력

출력해야 하는 명령어가 주어질 때마다 그 결과를 한 줄씩 출력한다.

## 예제 입출력

예제 입력	예제 출력
9 25 find_max print add 0 10 add 1 20 add 2 30 print add 1 40 print add 8 80 remove 7 add 6 50 add 5 60 at 0 remove 2 print add 5 15 add 8 19 at 4 add 5 17 add 7 21 find_max print set 3 333 print find_max	0  0 0 0 0 0 0 0 0 10 20 30 0 0 0 0 0 10 40 20 30 0 0 0 0 10 10 40 30 0 60 0 50 0 60 60 10 40 30 0 60 17 15 21 0 333 10 40 30 333 60 17 15 21 0 333         