

# 블록 탑

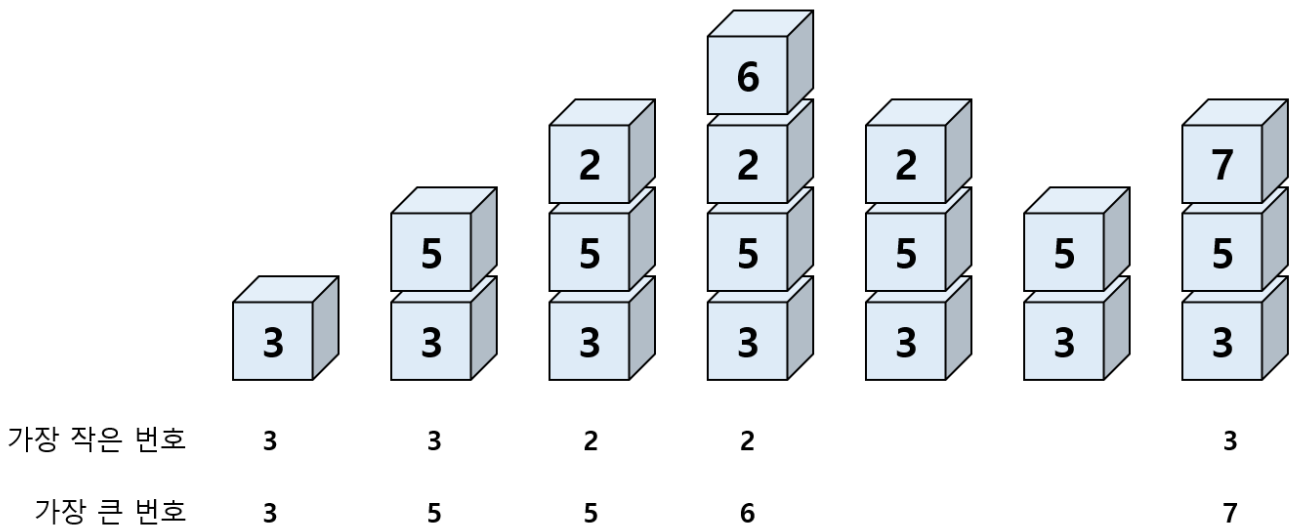
김인하는 블록을 높이 쌓아 탑을 만드는 것을 좋아한다. 김인하는 새로운 한 개의 블록을 블록 탑의 가장 위에 쌓거나, 블록 탑의 가장 위에 있는 한 개의 블록을 제거하며 블록 탑을 만든다. 이때, 각각의 블록에는 번호가 적혀있다.

어느 날 김인하는 블록 탑에서 가장 작은 블록의 번호와 가장 큰 블록의 번호가 무엇인지 궁금해졌다. 탑을 높이 쌓은 상태로 두 번호를 찾는 것은 힘들므로, 블록을 쌓을 때 마다 가장 작은 번호와 가장 큰 번호를 계속 작성해 두기로 했다. 예를 들어, [그림 1]과 같이 블록을 쌓는다고 하자. 김인하는 순서대로 3, 5, 2, 6 번 블록을 쌓은 후, 블록을 2 회 제거 하고, 7 번 블록을 쌓았다. 이때, 블록을 하나 쌓을 때 마다 가장 작은 블록의 번호와 가장 큰 블록의 번호를 그림과 같이 출력해야 한다.

김인하가 블록 탑을 쌓는 것은 아래의 질의로 주어진다.

- 1  $x$  : 블록 탑의 가장 위에 번호  $x$  ( $1 \leq x \leq 1,000$ )가 적힌 블록을 하나 쌓은 후, 현재의 블록 탑에서 가장 작은 블록의 번호와 가장 큰 블록의 번호를 출력한다.
- 2 : 블록 탑의 가장 위에 있는 블록을 하나 제거한다.

김인하를 도와, 블록을 쌓을 때 마다 가장 작은 번호와 가장 큰 번호를 출력하는 프로그램을 작성하시오.



[그림 1] 블록의 가장 작은 번호와 가장 큰 번호 출력 예시

※ 프로그램의 실행 시간은 1 초, 메모리 사용량은 512MB 를 초과할 수 없다.

사용할 수 있는 언어는 C, C++로 제한한다. C++의 경우 main 함수 내의 시작 지점에 다음 내용을 추가함으로써 cin, cout 의 입출력 속도를 개선할 수 있다.

```
ios_base::sync_with_stdio(false);
cin.tie(NULL);
cout.tie(NULL);
```

단, 위의 내용을 추가할 경우 cin, cout 만 사용해야 하며, scanf, printf 등 C 입출력을 혼용해서 사용하면 안된다. C++의 std::endl 의 경우 출력 속도가 느리므로, cout<<endl; 대신 cout<<"\n";을 사용하는 것을 권장한다.

# 입력

첫 번째 줄에 테스트 케이스 수  $T$  ( $1 \leq T \leq 100$ )가 주어진다.

각 테스트 케이스의 구성은 다음과 같다.

- 첫 번째 줄에 질의의 수  $N$  ( $1 \leq N \leq 100,000$ )이 주어진다.
- 이후  $N$ 개의 줄에 걸쳐, 위에서 설명한 질의들이 주어진다.

단, 블록 탐이 비어 있을 때 블록을 제거하는 질의(2 번 질의)가 주어지는 경우, 해당 질의는 무시한다.

# 출력

블록을 쌓는 질의(1 번 질의)가 주어질 때 마다, 질의를 수행한 후 블록 탐에 존재하는 가장 작은 블록의 번호와 가장 큰 블록의 번호를 공백으로 구분하여 한 줄에 출력한다.

## 예제 입출력

| 예제 입력 | 예제 출력  |
|-------|--------|
| 2     | 3 3    |
| 7     | 3 5    |
| 1 3   | 2 5    |
| 1 5   | 2 6    |
| 1 2   | 3 7    |
| 1 6   | 50 50  |
| 2     | 50 100 |
| 2     | 10 100 |
| 1 7   | 10 200 |
| 11    | 10 200 |
| 1 50  | 1 200  |
| 1 100 | 50 300 |
| 1 10  |        |
| 1 200 |        |
| 1 40  |        |
| 2     |        |
| 1 1   |        |
| 2     |        |
| 2     |        |
| 2     |        |
| 1 300 |        |