

Week 13

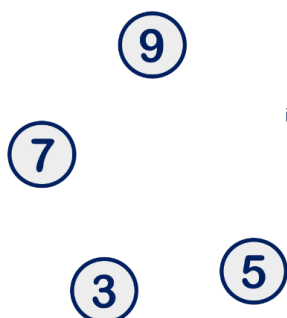
- 주의 사항: 부정행위 금지(채점서버 외 인터넷 사용금지), STL 사용금지 (string, vector는 사용 가능)
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

문제 1

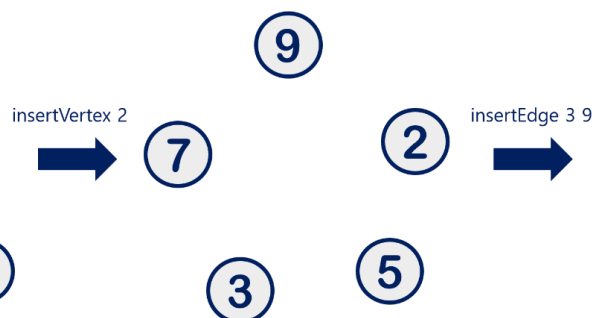
데이터를 입력 받아 인접행렬(Adjacent Matrix) 기반의 무향그래프를 생성하는 프로그램을 작성하시오. 입력으로 주어지는 명령어는 다음과 같이 총 5가지 이다.

- **insertVertex S**: 정점 $S(1 \leq S \leq 10,000)$ 를 그래프에 추가한다. 만약 같은 번호의 정점이 이미 존재한다면, 정점은 추가로 삽입하지 않고 "Exist"를 출력한다.
- **insertEdge S D**: 두 정점 S와 D를 잇는 간선을 삽입한다. 만약 간선이 이미 존재한다면, 간선은 추가로 삽입하지 않고 "Exist"를 출력한다. S와 D는 현재 그래프에서 존재하는 정점으로만 주어진다.
- **eraseVertex S**: 정점 S를 삭제한다. S는 현재 그래프에서 존재하는 정점으로만 주어진다.
- **eraseEdge S D**: 두 정점 S와 D를 잇는 간선을 삭제한다. 만약 간선이 존재하지 않으면 "None"을 출력한다. S와 D는 현재 그래프에서 존재하는 정점으로만 주어진다.
- **isAdjacent S D**: 정점 S와 정점 D가 연결되어 있다면 "True", 아니면 "False"를 출력하고 공백으로 구분하여 두 정점 중에서 정점의 차수(degree)가 더 큰 정점의 번호를 출력한다. S와 D는 현재 그래프에서 존재하는 정점으로만 주어지며, S와 D의 차수가 동일할 경우 번호가 더 큰 정점을 출력한다.
- **maxVertexInfo**: 현재 그래프에 존재하는 정점 중에서 번호가 가장 큰 정점의 번호와 해당 정점의 차수(degree)를 공백으로 구분하여 출력한다. 만약 출력할 정점이 없는 경우 -1을 출력한다.

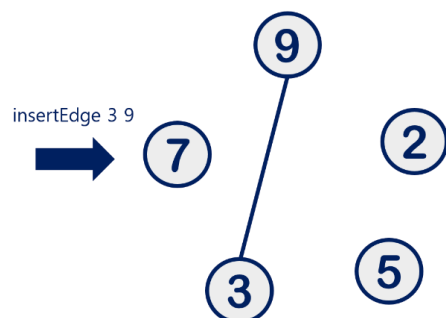
[그림 1]과 같이 3, 5, 7, 9 정점이 있고 간선이 하나도 존재하지 않는 그래프에서 insertVertex 2가 수행된 결과는 [그림 2]와 같으며, 이어서 insertEdge 3 9가 수행된 결과는 [그림 3]과 같다. 이때 maxVertexInfo를 수행한 출력 결과는 9 1이다.



[그림 1]



[그림 2]



[그림 3]

입력

첫 번째 줄에는 명령어의 개수 $T(1 \leq T \leq 10,000)$ 가 주어진다.

두 번째 줄부터 T 개의 줄에 걸쳐 명령어가 한 줄에 하나씩 주어진다. (단, 정점 S 와 D 가 주어질 때 두 정점은 서로 다르게 주어진다.)

출력

출력해야 하는 명령어가 주어질 때마다 그 결과를 한 줄에 하나씩 출력한다.

예제 입출력

예제 입력	예제 출력
22	-1
maxVertexInfo	10 0
insertVertex 3	Exist
insertVertex 10	True 2
insertVertex 2	False 2
insertEdge 2 3	None
insertVertex 1	10 0
insertEdge 1 3	False 1
maxVertexInfo	
insertVertex 3	
eraseVertex 3	
insertVertex 5	
insertEdge 10 2	
insertEdge 5 2	
insertEdge 2 1	
isAdjacent 2 10	
eraseEdge 2 10	
isAdjacent 10 2	
insertVertex 8	
insertEdge 8 1	
eraseEdge 10 2	
maxVertexInfo	
isAdjacent 1 5	