

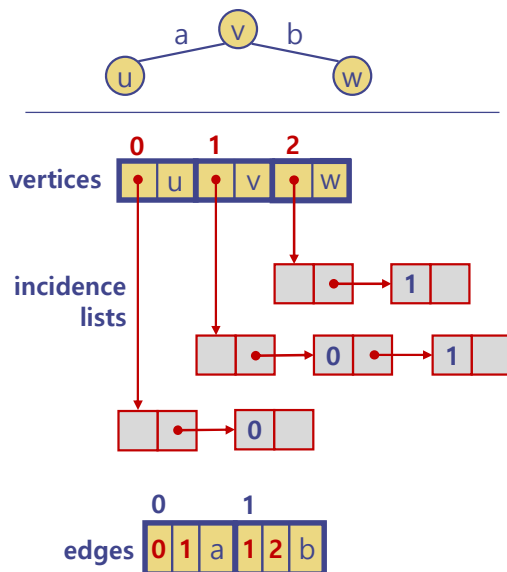
〈알고리즘 실습〉 - 그래프 표현 (보충-간단 버전)

※ 입출력에 대한 안내

- 특별한 언급이 없으면 문제의 조건에 맞지 않는 입력은 입력되지 않는다고 가정하라.
- 특별한 언급이 없으면, 각 줄의 맨 앞과 맨 뒤에는 공백을 출력하지 않는다.
- 출력 예시에서 □는 각 줄의 맨 앞과 맨 뒤에 출력되는 공백을 의미한다.
- 입출력 예시에서 \mapsto 이 후는 각 입력과 출력에 대한 설명이다.

[1] 인접리스트 구조 (문제1 참고)

구현 개념도 :

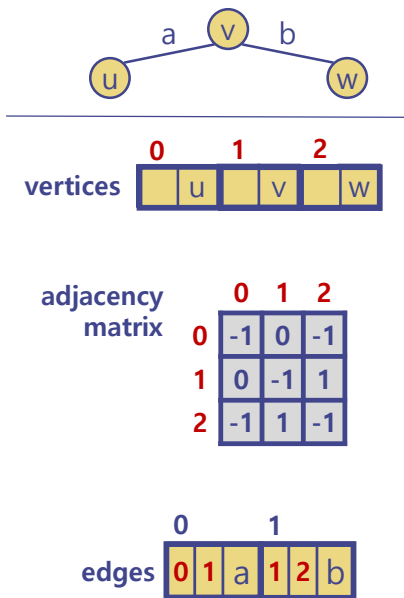


구현 :

- **structure graph :**
 - vertices(정점리스트) : an array[0:n-1] of structures
 - edges(간선리스트) : an array[0:m-1] of structures
- **structure vertex :**
 - name(이름) : char or string
 - edges(부착간선리스트) : a linked-list of integers
- **structure edge :**
 - name(이름) : char or string
 - endpoint1(끝점) : integer
 - endpoint2(끝점) : integer

[2] 인접행렬 구조 (문제 2 참고)

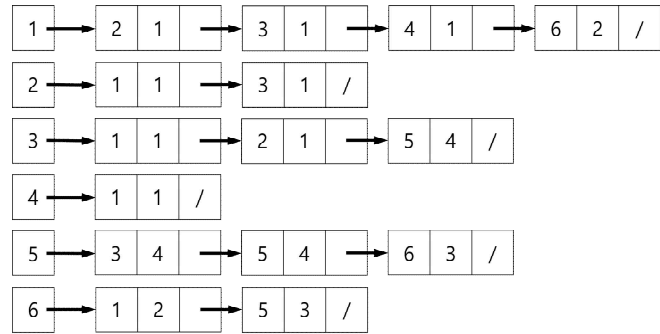
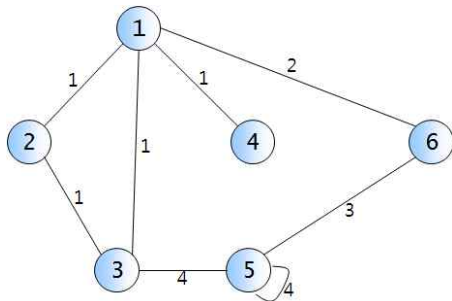
구현 개념도 :



구현 :

- **structure graph :**
 - vertices(정점리스트) : array[0:n-1] of structures
 - edges(간선리스트) : an array[0:m-1] of structures
 - adjacencyMatrix(인접행렬) : an array [0:n-1, 0:n-1] of integers
- **structure vertex :**
 - name : char or string
- **structure edge :**
 - name(이름) : char or string
 - endpoint1(끝점) : integer
 - endpoint2(끝점) : integer

[문제 1] (인접리스트 구현) 아래의 무방향 가중그래프를 인접리스트로 표현하고, 다음 명령어에 따라 그래프 정보를 출력하거나 그래프를 수정하는 프로그램을 작성하시오.



대화식 프로그램에 주어지는 명령어는 **a**, **m**, **q** 세 가지며 각 명령에 대해 다음과 같이 수행해야 한다.

a <node number> : <node number>를 가지는 node와 인접한 node와 그 노드까지의 간선 가중치를 모두 인쇄. 단, node number의 **오름차순**으로 인쇄하되, space 외의 구분자 없이 노드번호 가중치 노드번호 가중치 ... 형식으로 인쇄한다. 그래프에 정점 **a**가 존재하지 않으면 아무 것도 하지 않고 **-1**을 출력한다.

m a b w : 간선 (**a**, **b**)의 가중치를 **w**로 변경한다. 그러한 간선이 존재하지 않을 때는 가중치 **w**인 새로운 간선 (**a**, **b**)를 생성한다. **w = 0**이면 간선 (**a**, **b**)를 삭제한다. 그래프에 정점 **a** 혹은 **b**가 존재하지 않으면 아무 것도 하지 않고 **-1**을 출력한다.

q : 프로그램 종료

입력 예시 1

출력 예시 1

a 2	→ 노드번호	□ 1 1 3 1	→ (1번 노드, 가중치 1), (3번 노드, 가중치 1)
m 4 2 3	→ 간선 (4,2)에 가중치 3	□ 1 1 3 1 4 3	→ (1번 노드, 가중치 1), (3번 노드, 가중치 1), (4번 노드, 가중치 3)
a 2			
q			

입력 예시 2

출력 예시 2

a 5	→ 노드번호	□ 3 4 5 4 6 3
m 3 5 0	→ 간선 (3,5) 삭제	□ 5 4 6 3
a 5		-1
a 7		
q		

[문제 2] (인접행렬 구현) 아래의 무방향 가중그래프를 인접행렬로 표현하고, 명령어에 따라 그래프 정보를 인쇄하거나 그래프를 수정하는 프로그램을 작성하시오. 명령어 정의와 입출력 예시는 문제 1과 같다.

