Chapter 2:
Graph representation in computers

Discrete Mathematics 2

Lecturer: Nguyen Kieu Linh
Posts and Telecommunications Institute of Technology

Hanoi, 2023
http://www.ptit.edu.vn

Contents

Adjacency Matrix of Undirected Graph

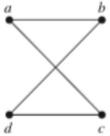
Definition

Suppose that $G = \langle V, E \rangle$ is a simple graph, where $|V| = n$. Suppose that the vertices of G are listed arbitrarily as $v_1, v_2, \ldots, v_n$. The *adjacency matrix* A of G, with respect to this listing of the vertices, is the $n \times n$ zero-one matrix $A = [a_{ij}]$, such that

$$a_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \text{ is an edge of } G, \\ 0 & \text{otherwise.} \end{cases}$$

Example

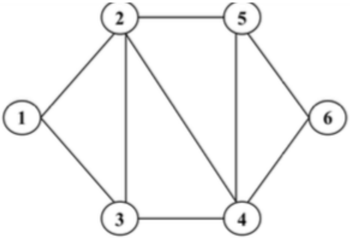
Use an adjacency matrix to represent the following graph



Adjacency Matrix of Undirected Graph

Example

Use an adjacency matrix to represent the following graph



Properties of Adjacency Matrix of Undirected Graph

* Symmetry: $a_{ij} = a_{ji}$.
* Sum of all the elements equals to two times of the number of edges

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} = 2|E|$$

* Sum of all the elements in row u equals to the degree of u

$$\sum_{j=1}^{n} a_{uj} = \deg(u)$$

Properties of Adjacency Matrix of Undirected Graph

- Sum of all the elements in column u equals to the degree of vertex u

$$\sum_{i=1}^{n} a_{iu} = \deg(u)$$

- Let $a_{ij}^{p} (i, j = 1, 2, \ldots, n)$ be the elements of matrix $A^p =$ A. $A...A$ (p times), then a_{ij}^{p} is the number of distinct paths from vertex i to vertex j through $p-1$ intermediate vertices

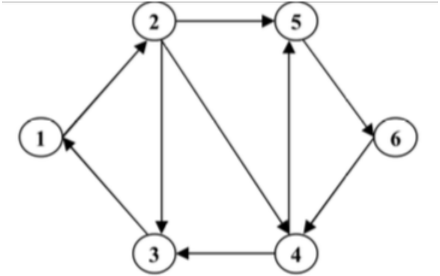
Adjacency Matrix of Directed Graph

Similar to the adjacency matrix of undirected graph, however

- There exists a direction on each directed egde
- Not symmetric

Example

Use an adjacency matrix to represent the following graph



Properties of Adjacency Matrix of Directed Graph

* Sum of all the elements equals to the number of edges

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} = |E|$$

* Sum of all the elements in row u equals to the outdegree of vertex u

$$\sum_{j=1}^{n} a_{uj} = \deg^{+}(u)$$

Properties of Adjacency Matrix of Directed Graph

* Sum of all the elements in column u equals to the indegree of vertex u
$$\sum_{i=1}^{n} a_{iu} = \deg^{-}(u)$$
* Let $a_{ij}^{p}(i,j = 1,2,\ldots,n)$ be the elements of matrix $A^p = A.A.\ldots A$ (p times), then a_{ij}^{p} is the number of distinct paths from vertex i to vertex j through $p-1$ intermediate vertices

Weighted (Adjacency) Matrix

- A real value $c = c(e) = c(u, v)$ is assigned to each edge $e = (u, v)$, called the weight of edge e.
- In this case the graph is called *weighted graph*.
- Weighted matrix $\mathrm{C} = [c_{ij}]$, with

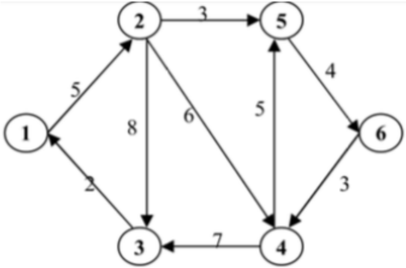
$$c_{ij} = \begin{cases} c & \text{if } \{v_i, v_j\} \text{ is an edge of } G, \\ \theta & \text{otherwise.} \end{cases}$$

where, θ can received values $0, \infty, -\infty$ depending on the problem

Weighted (Adjacency) Matrix

Example

Use an adjacency matrix to represent the following graph



Advantages & Disadvantages of Adjacency Matrix

Advantages

- Simple and easy to be implemented (Using a two dimensional array to represent)
- Easy to check whether two vertices u, v are adjacent or not (Only one comparison operation $a[u][v] \neq 0$?)

Disadvantages

- Waste of memory: we need n^2 memory units despite the number of edges
- Cannot represent a graph with a large number of vertices
- To find adjacency vertices of vertex u we need n comparison operations despite the degree of u.

Adjacency Matrix Storage

- The first line is the number of vertices.
- The next *n* lines store the adjacency matrix.
- Two elements are separated by one or more spaces.

```
10
0 1 1 1 0 0 0 0 0 0
1 0 0 1 1 0 0 0 0 0
1 0 0 1 0 1 0 0 0 0
1 1 1 0 1 1 0 0 1 0
0 1 0 1 0 0 0 1 0 0
0 0 1 1 0 0 1 0 0 0
0 0 0 0 0 1 0 0 1 1
0 0 0 0 1 0 0 0 1 1
0 0 0 1 0 0 1 1 0 1
0 0 0 0 0 0 1 1 1 0
```

Contents

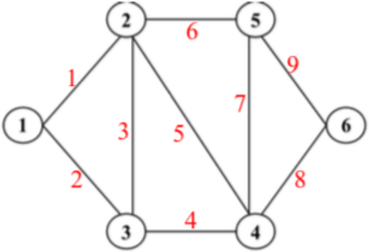
Incidence Matrix of Undirected Graph

Consider undirected graph $G = (V, E)$, $V = \{1, 2, \ldots, n\}$, $E = \{e_1, e_2, \ldots, e_m\}$. *Vertex-edge incidence matrix* of G is a $n \times m$ matrix $A = [a_{ij}]$:

$$a_{ij} = \begin{cases} 1, & \text{if vertex } i \text{ and edge } j \text{ are incident} \\ 0, & \text{otherwise} \end{cases}$$

Example

Use a vertex-edge incidence matrix to represent the graph



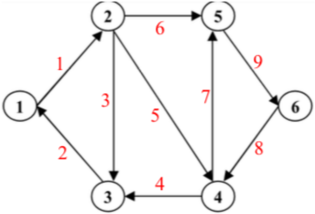
Incidence Matrix of Directed Graph

Consider directed graph $G = (V, E), V = \{1, 2, \ldots, n\}, E = \{e_1, e_2, \ldots, e_m\}$. *Vertex-directed edge incidence matrix* of G is a $n \times m$ matrix $A = a_{ij}$:

$$a_{ij} = \begin{cases} 1, & \text{if vertex } i \text{ is the initial vertex of directed egde } e_j \\ -1, & \text{if vertex } i \text{ is the terminal vertex of directed egde } e_j \\ 0, & \text{otherwise} \end{cases}$$

Example

Use a vertex-directed edge incidence matrix to represent the graph



Contents

Edge List

In the case of sparse graphs ($m \leq 6n$), we usually use edge (directed edge) list

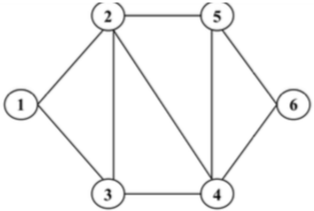
- We store all the edges (directed edges) of the undirected graph (directed graph).
- Each edge (directed edge) $e(x, y)$ is represented by two variables $start[e]$, $end[e]$.
- Advantages: we need only $2m$ memory units to store the graph
- Disadvantages: to know which vertices are adjacent to a vertex we need m comparison operations (examine all m edges)
- For weighted graph, we need more m memory units to store weights of edges

Edge List of Undirected Graph

- We only list (u, v), do not list (v, u)
- Should list edges in the ascending order of the starting points of edges
- The number of edges containing u (left or right) is the degree of vertex v.

Example

Use a edge list to represent the following undirected graph

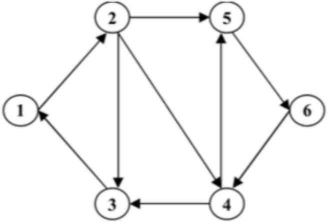


Edge List of Directed Graph

- List vertices in two columns, the first column lists the Initial vertices, the second column lists the terminal vertices corresponding to the first vertices of a directed edge.
- On each row, initial vertex does not have to be less than terminal vertex.

Example

Use a edge list to represent the following directed graph

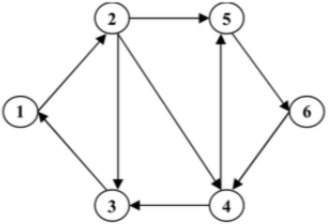


Edge List of Weighted Graph

- For weighted Graph, we need to add a column to list the corresponding weights of Edges.

Example

Use a edge list to represent the following weighted graph



Edge List Storage

- The first line stores M, N, the number of vertices and the number of edges of the graph
- In the next M lines, each line stores an edge of the graph.

6	9	
1	2	5
2	3	8
2	4	6
2	5	3
3	1	2
4	3	7
4	5	5
5	6	4
6	4	3

Advantages & Disadvantages of Edge List

Advantages

- Save memory in the case of sparse graphs $m < 6n$.
- Convenient for algorithms considering only edges of graphs.

Disadvantages

- To know which vertices are adjacent to vertex u we have to examine all the edges of the graph.
- High computational complexity.

Contents

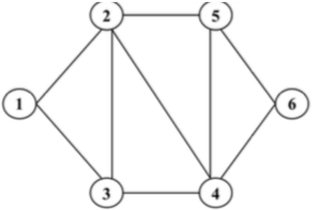
Adjacency List

- For each vertex u, we store the list of its adjacent vertices denoted by Adj(u), where

$$Adj(u) = \{v \in V \mid (u, v) \in E\}.$$

Example

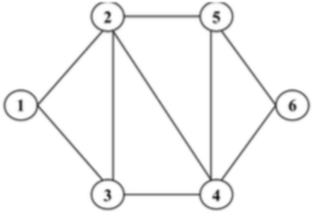
Use an adjacency list to represent the following graph



Adjacency List using Array

- Array consists of n segments.
- The i^{th} segment stores the adjacency list of the i^{th} vertex.
- To know where a segment starts or ends, we use another array storing the starting and ending elements of the segment

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
A[i]=?	2	3	1	3	4	5	1	2	4	2	3	5	6	2	4	6	4	5
	Segment 1		Segment 2				Segment 3			Segment 4				Segment 5			Segment 6	



Adj(1) = { 2, 3 }
Adj(2) = {1, 3, 4, 5}.
Adj(3) = {1, 2, 4}.
Adj(4) = {2, 3, 5, 6}.
Adj(5) = {2, 4, 6}.
Adj(6) = { 4, 5}.

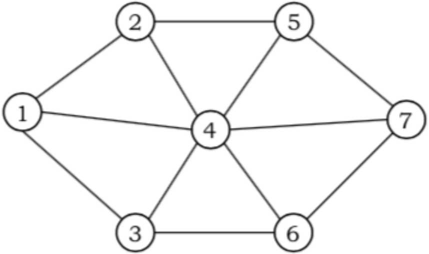
Adjacency List Storage

- The first line stores the number of vertices.
- Next N lines store adjacency lists of vertices as follows:
  - The first number is the ending position of the segment, other numbers are list of vertices of the linked list.
  - Numbers are separated by one or more spaces.

```
6
2   2   3
6   1   3   4   5
9   1   2   4
13  2   3   5   6
16  2   4   6
18  4   5
```

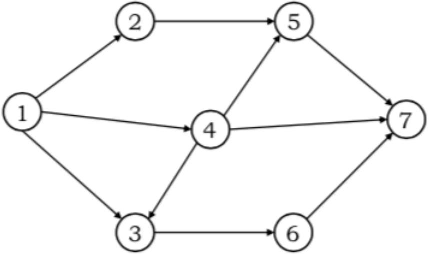
Exercises

Exercise 1. Represent the undirected graph below using: Adjacency matrix, Edge list, Adjacency list.



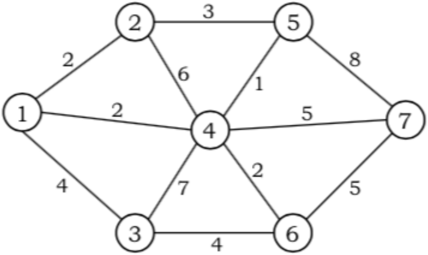
Exercises

Exercise 2. Represent the directed graph below using: Adjacency matrix, Edge list, Adjacency list.



Exercises

Exercise 3. Represent the weighted graph below using: Weighted matrix, Edge list.



Exercises

Exercise 4. Given an undirected graph $\mathrm{G} = \langle \mathrm{V}, \mathrm{E} \rangle$ consisting of 8 vertices is represented as an adjacency list as follows:

$$\begin{array}{lll} \mathrm{Adj}(1) = \{2,3\} & \mathrm{Adj}(4) = \{2,5,6\} & \mathrm{Adj}(7) = \{5,8\} \\ \mathrm{Adj}(2) = \{1,3,4\} & \mathrm{Adj}(5) = \{4,6,7\} & \mathrm{Adj}(8) = \{7\} \\ \mathrm{Adj}(3) = \{1,2\} & \mathrm{Adj}(6) = \{4,5\} & \end{array}$$

a) Find the degree of each vertex in the graph.
b) Represent the graph G as an adjacency matrix, edge list.

Exercises

Exercise 5. Given a directed graph $\mathrm{G} = \langle \mathrm{V}, \mathrm{E} \rangle$ consisting of 8 vertices is represented as an adjacency list as follows:

$$\begin{array}{lll} \mathrm{Adj}(1) = \{3\} & \mathrm{Adj}(4) = \{5, 8\} & \mathrm{Adj}(7) = \{4, 6\} \\ \mathrm{Adj}(2) = \{1, 4\} & \mathrm{Adj}(5) = \{2\} & \mathrm{Adj}(8) = \{7\} \\ \mathrm{Adj}(3) = \{2, 4\} & \mathrm{Adj}(6) = \{5\} & \end{array}$$

a) Find the in-degree and the out-degree of each vertex on the graph.
b) Represent the graph G as an adjacency matrix.
c) Find the number of paths of length 2 on the graph G from vertex 1 to vertices 3, 7 and 8?

Exercise 6. Given an undirected graph $\mathrm{G} =< \mathrm{V}, \mathrm{E} >$ consisting of 10 vertices and 20 edges is represented as an edge list as follows:

Initial Vertex	Terminal Vertex	Initial Vertex	Terminal Vertex
1	2	5	7
1	5	5	9
1	8	5	10
1	10	6	7
2	3	6	10
2	4	7	8
2	6	7	9
4	6	7	10
4	8	8	9
5	6	9	10

a) Find $\deg(u)$ for all $u \in V$?

b) Express the graph $\mathrm{G} =< \mathrm{V}, \mathrm{E} >$ as an adjacency matrix?

c) Represent the graph $\mathrm{G} =< \mathrm{V}, \mathrm{E} >$ as an adjacency list?

Exercise 7. Given an undirected multigraph $\mathrm{G} = \langle \mathrm{V}, \mathrm{E} >$ consisting of 10 vertices is represented as an adjacency matrix as follows:

$$\begin{vmatrix} 0 & 2 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 2 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 1 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 \end{vmatrix}$$

a) Find $\deg(\mathrm{u})$ for all $u \in V$? (disconnected)
b) Represent the graph $\mathrm{G} =< \mathrm{V}, \mathrm{E} >$ as an edge list?
c) Find the number of paths from vertex 4 to vertices 1,5 and 9 ?