

## CHƯƠNG 2

### BIỂU DIỄN ĐỒ THỊ TRÊN MÁY TÍNH

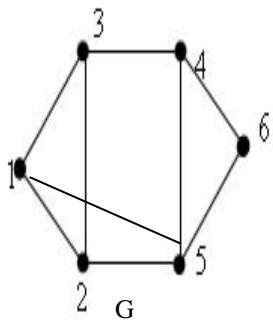
#### 1. Ma trận kề, ma trận trọng số

Gọi  $n$  là số đỉnh,  $m$  là số cạnh/cung của đơn đồ thị  $G=(V,E)$ ,

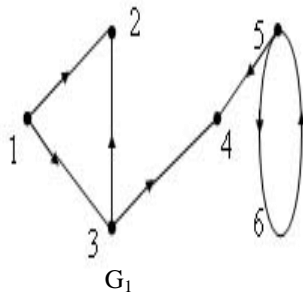
##### a) Ma trận kề:

- Sử dụng mảng hai chiều  $a[n][n]$ .
- $a[i][j] = 0$ , nếu cạnh/cung  $(i,j) \notin E$  và  $a[i][j] = 1$  nếu cạnh/cung  $(i,j) \in E$

Ví dụ:



Hình 1



	1	2	3	4	5	6
1	0	1	1	0	1	0
2	1	0	1	0	1	0
3	1	1	0	1	0	0
4	0	0	1	0	1	1
5	1	1	0	1	0	1
6	0	0	0	1	1	0

Ma trận kề của G

	1	2	3	4	5	6
1	0	1	1	0	0	0
2	0	0	0	0	0	0
3	0	1	0	1	0	0
4	0	0	0	0	0	0
5	0	0	0	1	0	1
6	0	0	0	0	1	0

Ma trận kề của  $G_1$

\* Tính chất của ma trận kề của đồ thị vô hướng:

- Tính đối xứng:  $a[i][j] = a[j][i]$
- Tổng các phần tử trên dòng  $i$  (cột  $j$ ) bằng bậc của đỉnh  $i$  (đỉnh  $j$ ).
- Gọi  $a_{ij}^k$  là phần tử của ma trận  $A^k = A \cdot A \cdot \dots \cdot A$  ( $k$  thừa số)  
Khi đó:  $a_{ij}^k$  là số đường đi khác nhau từ đỉnh  $i$  đến đỉnh  $j$  qua  $k-1$  đỉnh trung gian.

\* Tính chất của ma trận kề của đồ thị có hướng:

- Không có tính đối xứng
- Tổng các phần tử trên dòng  $i$  bằng bán bậc ra của đỉnh  $i$  ( $\deg^+(i)$ ) và tổng các phần tử trên cột  $j$  bằng bán bậc vào của đỉnh  $j$  ( $\deg^-(j)$ ).
- Giống t/ch 3 của vô hướng

\* Ma trận kề của đa đồ thị:  $a[i,j] =$  số cạnh (cung) nối hai đỉnh  $i, j$ .

##### b) Ma trận trọng số:

Đồ thị có trọng số: mỗi cạnh/cung  $(i,j)$  có một giá trị  $c_{i,j}$  gọi là trọng số của cạnh/cung.

- Sử dụng Mảng hai chiều  $a[n][n]$ .
- $a[i][j] = c_{i,j}$  nếu cạnh/cung  $(i,j) \in E$ ;  $a[i][j] = \theta$ , nếu cạnh/cung  $(i,j) \notin E$  ( $\theta = 0 / +\infty / -\infty$ )

Nhận xét:

- Ma trận kề và ma trận trọng số sử dụng  $n \times n$  đơn vị bộ nhớ.
- Khi xét đỉnh  $j$  có kề đỉnh  $i$  hay không, ta chỉ cần xét phần tử  $a[i][j]$ .
- Nếu  $m \ll n$  thì rất lãng phí bộ nhớ. Khi đó nên sử dụng ma trận liên thuộc

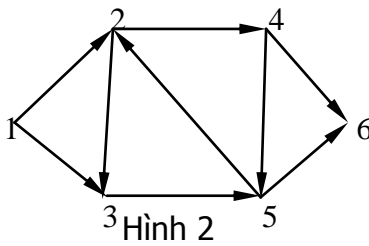
#### 2. Ma trận liên thuộc

- Sử dụng mảng hai chiều  $a[n][m]$ .
- Mỗi cạnh/cung lưu vào một cột
- Ma trận liên thuộc có dạng:

$$a_{ij} = \begin{cases} 1, & \text{nếu } i \text{ là đỉnh đầu của cung } e_j \\ -1, & \text{nếu } i \text{ là đỉnh cuối của cung } e_j \\ 0, & \text{nếu } i \text{ không là đầu nào của cung} \end{cases}$$

(Nếu vô hướng thì đầu và cuối đều bằng 1)

Ví dụ:



$$A = \begin{matrix} & \begin{matrix} (1,2) & (1,3) & (2,3) & (2,4) & (3,5) & (4,5) & (4,6) & (5,2) & (5,6) \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{matrix} & \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & -1 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 \end{pmatrix} \end{matrix}$$

Nhận xét:

- Sử dụng  $n \times m$  đơn vị bộ nhớ.
- Khi xét đỉnh  $j$  có kề đỉnh  $i$  hay không thì chậm
- Nếu  $m \ll n$  thì tiết kiệm bộ nhớ hơn ma trận kề.

Ví dụ: xem đồ thị  $G, G_1$  ở hình 1

### 3. Danh sách cạnh

- Sử dụng mảng hai chiều  $a[2][m]$ .
- Mỗi cạnh/cung lưu vào một hàng

Nhận xét:

- Sử dụng  $2 \times m$  đơn vị bộ nhớ.
- Khi xét đỉnh  $j$  có kề đỉnh  $i$  hay không thì chậm
- Tiết kiệm bộ nhớ nhiều nhất khi  $m < n$ .
- Nếu đồ thị có trọng số, cần thêm  $m$  đơn vị bộ nhớ để lưu trữ trọng số của các cạnh/cung.

D/s cạnh của  $G$    D/s cung của  $G_1$

Đầu	Cuối		Đầu	Cuối
1	2		1	2
1	3		1	3
1	5		3	2
2	3		3	4
2	5		5	4
3	4		5	6
4	5		6	5
4	6			
5	6			

#### 4. Danh sách kề

- Sử dụng n danh sách liên kết:  $\text{int } *a[n]$
- Danh sách liên kết thứ i lưu những đỉnh kề với đỉnh i

Ví dụ: Danh sách kề của G (hình 1)

1	→	2	→	3	→	5	nil
2	→	1	→	3	→	5	nil
3	→	1	→	2	→	4	nil
4	→	3	→	5	→	6	nil
5	→	1	→	2	→	4	→ 6 nil
6	→	4	→	5	nil		

Danh sách kề của G1 (hình 1)  
Đỉnh đầu

1	→	2	→	3	nil
2	nil				
3	→	2	→	4	nil
4	nil				
5	→	4	→	5	nil
	nil				
6	→	5			

Nhận xét:

- Sử dụng  $n+m$  (có hướng) hoặc  $n+2m$  (vô hướng) đơn vị bộ nhớ,
- Khi xét đỉnh j có kề đỉnh i hay không thì khá nhanh vì ta chỉ xét danh sách thứ i
- Khá tiết kiệm bộ nhớ

\* Tóm lại

	MTK	MTLT	DSC	DSK
Kích thước	$n*n$	$n*m$	$2*m$	$n+m; n+2m$

- Nếu không quan tâm tới tính tiết kiệm bộ nhớ thì sử dụng MT Kề
- Nếu đồ thị số cạnh ít, số đỉnh nhiều ( $m < n$ ) thì sử dụng DS cạnh/cung
- Nếu đồ thị lớn thì sử dụng DS Kề

## BÀI TẬP CHƯƠNG 2

### Bài 1:

Cho file đồ thị vô hướng dạng ma trận kề, đọc file vào cấu trúc ma trận kề, tính bậc của mỗi đỉnh.

Ví dụ file đồ thị dạng sau:

```
3 //3 là số đỉnh
0 0 1
0 0 1
1 1 0
```

### Bài 2:

Cho file đồ thị vô hướng, dạng danh sách cạnh, đọc file vào cấu trúc ma trận kề, tính bậc của mỗi đỉnh.

ví dụ:

```
4 //4 là số đỉnh
1 2 //cạnh (1,2)
1 3
```

1 4  
2 3

### Bài 3:

Cho file đồ thị có hướng, dạng danh sách cung, đọc file vào cấu trúc ma trận kề, tính bán bậc vào/ra của mỗi đỉnh.

ví dụ:

4 //4 là số đỉnh  
1 2 //cung (1,2)  
2 3  
3 2

### Bài 4:

Cho file đồ thị có hướng, dạng danh sách cung, đọc file vào cấu trúc ma trận liên thuộc, tính bán bậc vào/ra của mỗi đỉnh.

### Bài 5:

Cho file đồ thị có hướng, dạng danh sách cung, đọc file vào cấu trúc danh sách cạnh/cung, tính bán bậc vào/ra của mỗi đỉnh.

### Bài 6:

Cho file đồ thị có hướng, dạng danh sách cung, đọc file vào cấu trúc danh sách kề, tính bán bậc vào/ra của mỗi đỉnh.

### Bài 7:

Một bàn cờ 8×8 được đánh số như sau:

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

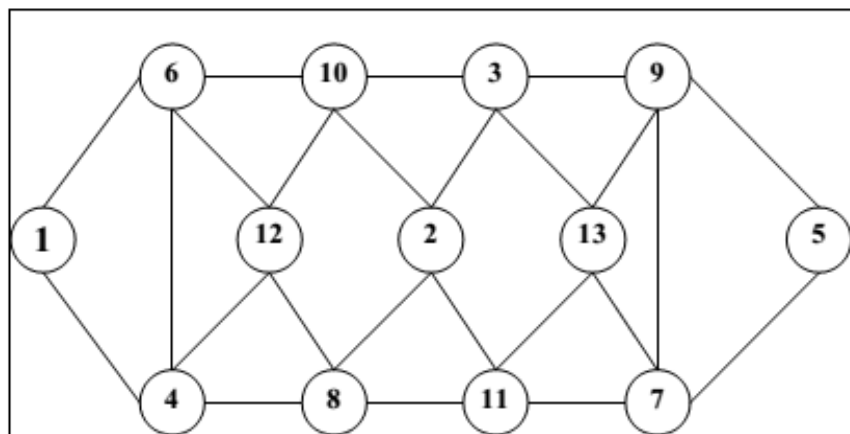
Mỗi ô có thể coi là một đỉnh của đồ thị. Hai đỉnh được coi là kề nhau nếu một con vua đặt ở ô này có thể nhảy sang ô kia sau một bước đi. Ví dụ : ô 1 kề với ô 2, 9, 10, ô 11 kề với 2, 3, 4, 10, 12, 18, 19, 20. Hãy viết chương trình tạo ma trận kề của đồ thị, kết quả in ra file king.out.

### Bài 8:

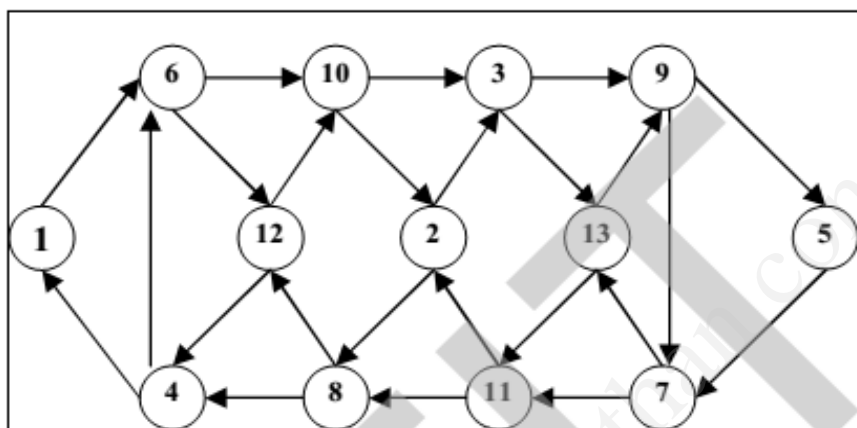
Bàn cờ 8×8 được đánh số như bài 7. Mỗi ô có thể coi là một đỉnh của đồ thị. Hai đỉnh được gọi là kề nhau nếu một con mã đặt ở ô này có thể nhảy sang ô kia sau một nước đi. Ví dụ ô 1 kề với 11, 18, ô 11 kề với 1, 5, 17, 21, 26, 28. Hãy viết chương trình lập ma trận kề của đồ thị, kết quả ghi vào file horse.out.

**Bài 9:** Hãy biểu diễn đồ thị dưới đây dưới dạng ma trận kề/mt ts, danh sách cạnh, danh sách kề.

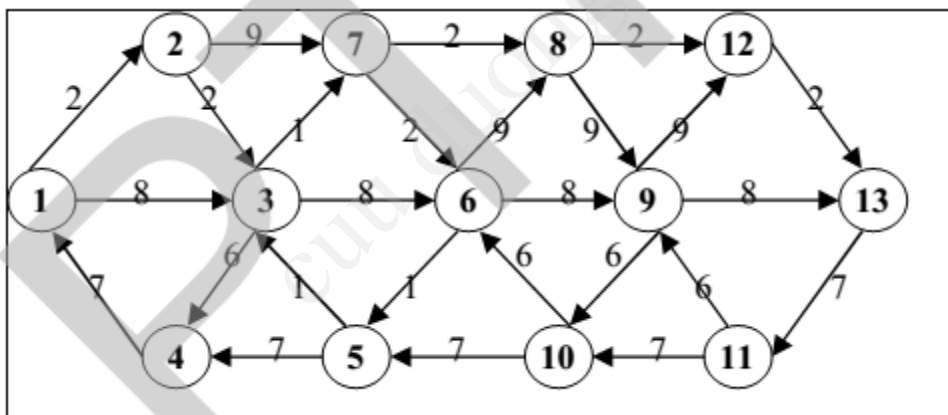
a/



b/



c/



- Hết -