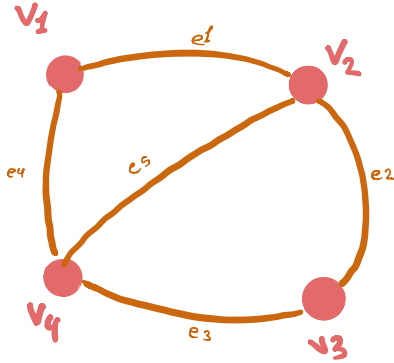


GRAPH

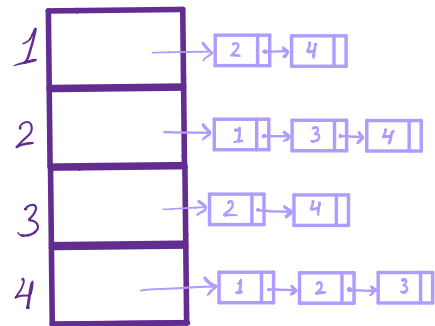
DFS \leftarrow Stack
BFS \leftarrow queue

* سلايات تعريف لها



adjacency list

* كل رأس والي مرتبط به

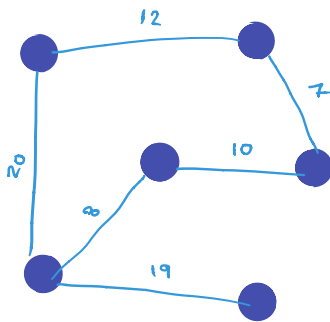


$O(e + n)$
عدم الرؤوس \leftarrow عدد الأضلاع

adjacency Matrix

عدد الرؤوس	1	2	3	4
1	0	1	0	1
2	1	0	1	1
3	0	1	0	1
4	1	1	1	0

* إذا بين edge خط 1



* Weighted graph

* إذا بنسوي لها matrix خط برال
ال 1 الرقم الموجود

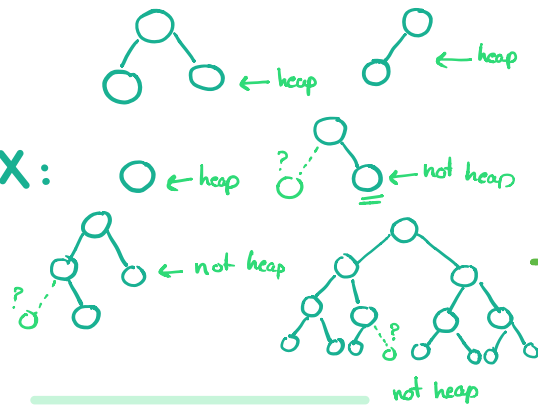
* تصوير متناظر (Symetric) إذا حبار
(الجدولين لمتناظرين نفس الشيء)
* إذا undirected \leftarrow داييم تصوير Symetric

* إذا بنسوي لها List خط
V, edge
الرسم \rightarrow الرأس

تاعده عامه
* إذا directed graph \leftarrow وعدد (n) \leftarrow عدد edge $(n-1)$
* إذا undirected \leftarrow عدد (n) \leftarrow عدد edge $\frac{n(n-1)}{2}$

HEAP

Ex:



عدد Nodes المسموح	المستوى	
$2^{1-1} = 1$	1	
$2^{2-1} = 2$	2	
$2^{3-1} = 4$	3	
$2^{4-1} = 8$	4	

- Complete binary tree
لا تبدأ إحتافه من اليسار دائماً

- Satisfy heap property (value)

* Min heap

* Root > Nodes

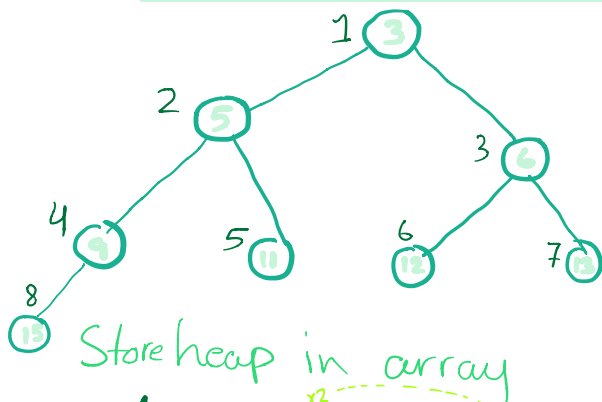
* Max heap

* Root < Nodes

$$2^{h-1} \leq n \leq 2^h - 1 \quad - h = \log(n+1)$$

عدد Nodes التي تقدر غطه المستوى

$$O(n) = \log(n)$$



Store heap in array

0	1	2	3	4	5	6	7	8
not use	3	5	6	9	11	12	13	15

- Sift up

- Sift down

- Sort

* ميب مكتوبه

ماعرفت اشرحها
كتابها غابله
مهم

finding the left child for a Node

Ex: What is The left child for 6?

6 index is $\rightarrow 3$ $2(k)$

So $3 \times 2 = 6$

the index for the child

* For the Right child $2(k)+1$