Github: https://github.com/Rasheedfahmi

Name:Rasheed Fahmi

ID: 2016907

Data structure

tree heirartichal structure

notes

1 - parents vs children

2 - siblings



 3 - ancestor vs descentants

binary tree

BST binary search tree

**tree**

1- pre-order

2-in-order

3-Post-order - BFS

- DFS

*1- Bfs*

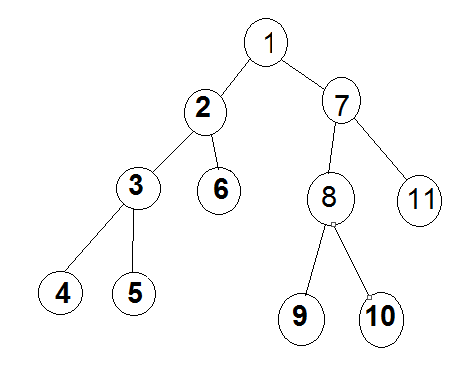
*-Floyedd = shorter path*

*-Beli man-ford*

*Algorithm*

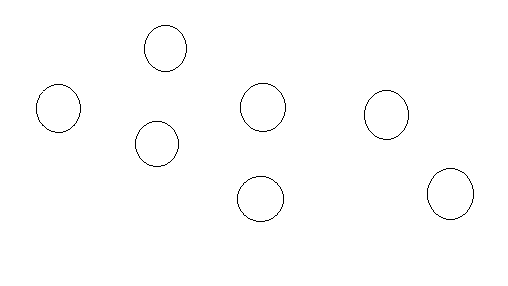
*Liner search binary sontag buble sort*

**Data structure as; array… linked list**



***2-DFS***

***Depth First search***

***Tree*  special type from data type called graph**

***Tree*  is the simplest from the graph**

**Graph**

**- connected vs not connect**

**- directed or undirected ( have no dirction for the edges)**

**- weighted vs no weighted**

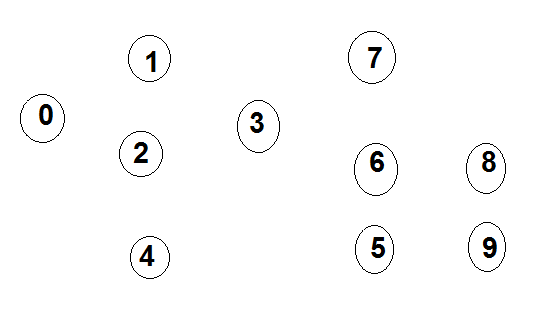
**- edge have value**

**- cyclic vs a cyclic**

**- dens vs spowse**

**1- how to implement the graph undirected and weighted by 2D array any matrix array ?**

**2 - undirected and weight put the value of the edges on matrix ?**

**3 – if it is directed and weighted graph by using adjancency matrix ?**

**4 – if it is sparse graph have no edges by adjancency list ?**

Code int main(void){

Intg [10] [1] = 5j

g [0] [1] = 5j

g [0] [2] = 2j

g [1] [0] =5j

g [2] [0] =2j

g [5] [6] =3j

g [6] [5] = 3j

adjancecy list

int max (void) {using 52 bytes in memory

structure node {4

int key ;

int weight ; struct node \* next ;};

struct node g [10]; g[i]=nul ptr;

struct node g1; g1 key =1

g1 weight=5;

g[0]=g1 struct node g2;

g2key=2

g2 weight =2;

g1.next=g2;}

Time

0(n^4)

0(n^3)

0(n^2)

Linear 0 (n)

0(n log n)

0 ( log n )

Constant 0 ( 1 )

Input Size.

**Example:**

**constant insert/remove array**

**(log n) binary search**

**(n log n) Heap sort/quick sort**

**linear  complexity increase as input size**