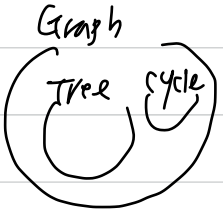
# lec10 Graph

- DFS/BFS
- MST (Kruskal / Prim algorithms)

Graph : data structure that represents the relationships btw connected objs.




Vertex (node) / Edge (link)

directed graph o→o

o—o

Degree of Graph. → $\sum degree(V) = 2|E|$ for undirected graph

- directed graph → $\sum in\text{-}degree(V) = \sum out\text{-}degree(V) = |E|$

Graph Representation

- Adjacency matrix (undirected → symmetric)
- Adjacency list.

DFS (depth first search.) → stack → LIFO / recursion.

```
void dfs_mat(GraphType *g, int v)
{
    int w;
    visited[v] = TRUE;
    printf("%d", v);
    for(w=0; w<g->n; w++)
        if(g->adj_mat[v][w] && !visited[w])
            dfs_mat(g, w); // dfs on node w
}

void dfs_list(GraphType *g, int v)
{
    GraphNode *w;
    visited[v] = TRUE;
    printf("%d", v);
    for(w=g->adj_list[v]; w; w=w->link)
        if(!visited[w->vertex])
            dfs_list(g, w->vertex);
}
```

$O(|V|+|E|)$