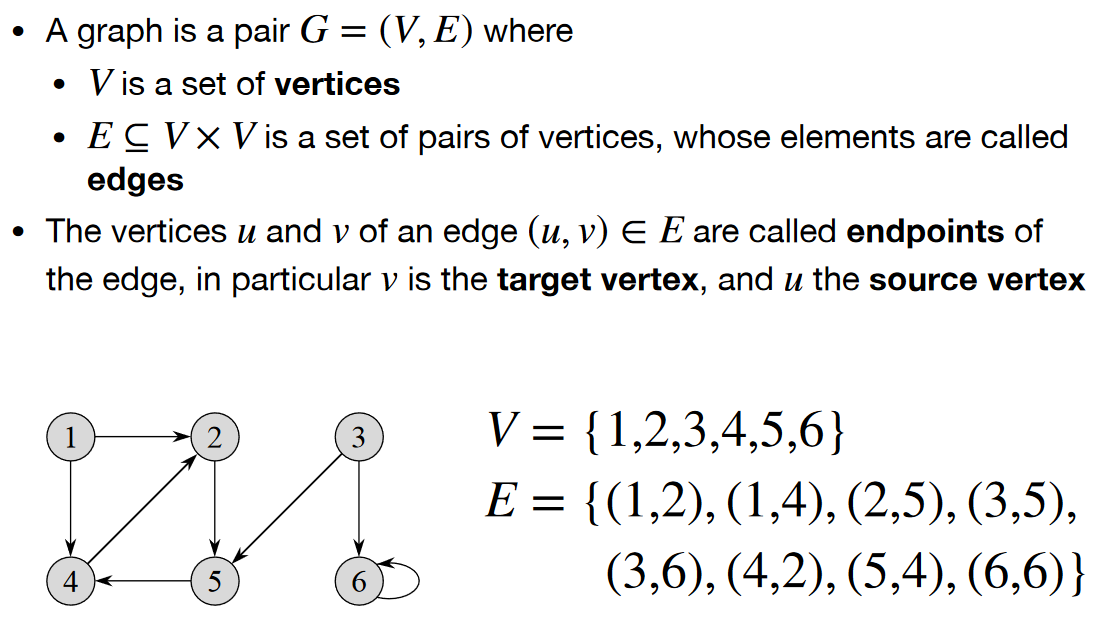
# Representation of graphs

## Undirected graphs

## 

## Directed graphs

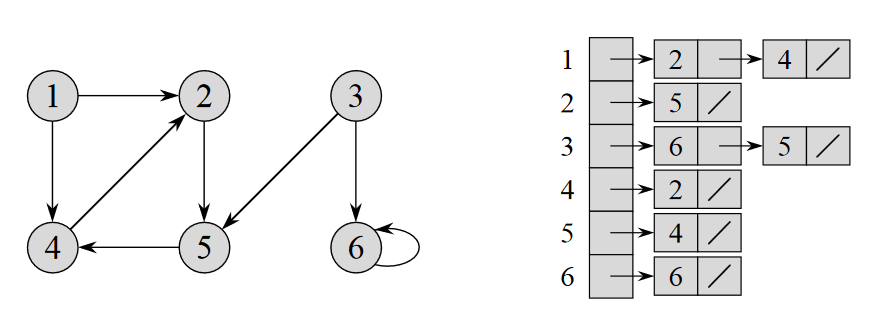


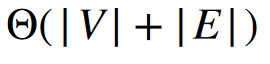
There are two classical representations of graphs.

## Adjacency-list

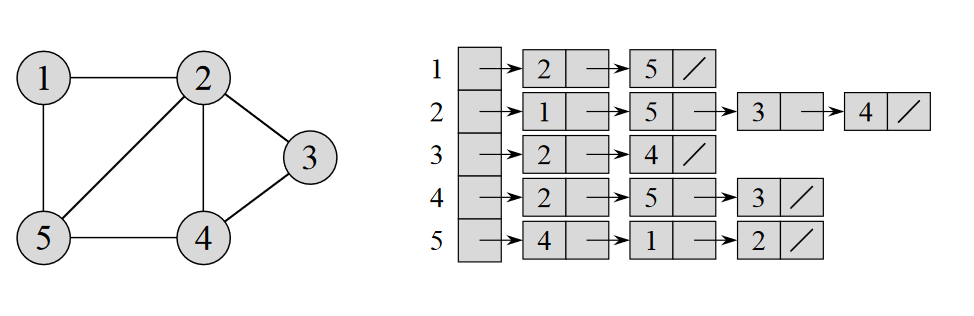
For the graph *G = (V,E)* there exist an array called *Adj* of │*V*│.

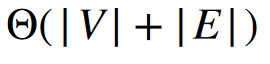
For every vertex *U* in *V* , Adj[U] consists of all the adjacent vertices to U.  
This example is for directed graphs:



The space required is . In this case it is 6 + 8 = 14.

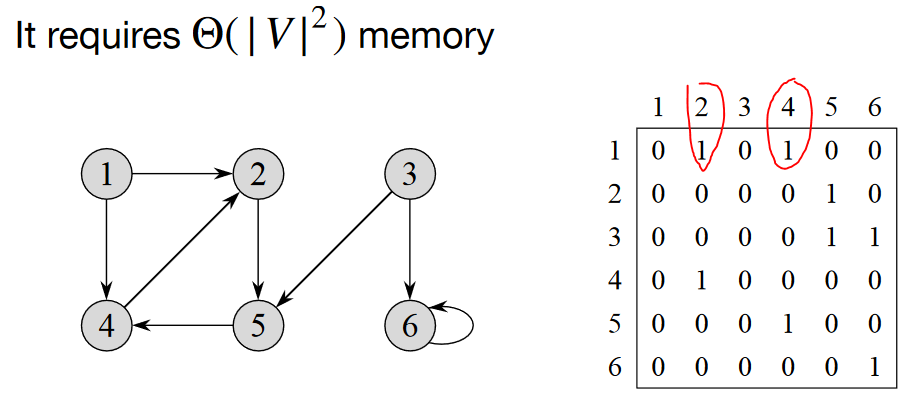
This example is for undirected graphs:



The space consumed is stil .

## Adjencency matrix

You can write the adjacencies in a matrix



You can se in the matrix that vertice 1 connects to 2 and 4 with the red circles. Works the same way with undirected graphs.

## Depth-first search (DFS) Properties

There are several properties of the DFS algorithm.

