Overview

- [[Graph Terminology]]
- [[Graph Algorithms]]
- [[Graphentheorie]]

Storing Graph

- Size of a graph G = (V, E):
 - \circ Number of vertices: n = |V|
 - Number of edges: $m = |E|, \ 0 \le m \le n^2$
 - \Rightarrow In total: size $\Theta(n+m)$
 - ⇒ Analysis needs two parameters !
- o **dense** graphs: $m \approx n^2$, for example complete graphs
- o sparse graphs: $m \ll n^2$, for ex. trees (m = n 1) or hypercubes $(m = \frac{d}{2} \cdot n = O(n \log n)$, where d is the dimension of the hypercube)
- **Adjacency-matrix**: Matrix $A[1 \dots n, 1 \dots n]$ with

$$A[i,j] = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{else} \end{cases}$$

- Memory: $\Theta(n^2)$
- · convenient for dense graphs
- test for existence of an edge in $\Theta(1)$ time

Adjacency-List: Array $F[1 \dots n]$ with pointers,

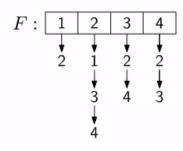
- F[i] points to a linear list with all vertices that are incident to the i^{th} vertex
- Memory: $\Theta(n+m)$
- test for existence of an edge in $\Omega(1), O(n)$ time

A small example:



$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

 \uparrow symmetric if graph is undirected



 $\leftarrow \text{1 Entry for every node,} \\ \Theta(n+m) \text{ memory for all edges.} \\ \text{The } k \text{ neighbours of a node can} \\ \text{be obtained in } \Theta(k) \text{ time.} \\$