# CS 762 – Required background

All of the following topics should be familiar to you. If not: the appendix of the lecture notes covers many of them; and so would most textbooks on algorithms (e.g. Cormen, Leisersen, Rivest (& Stein)).

- **Graphs** (directed and undirected):
  - vertex/vertices, edges
  - incident edge, adjacent vertex, neighbour
  - degree, in-degree, out-degree
  - loop, multi-edge, simple graph
  - path/cycle of length $k$, both undirected and directed
  - connected graph, connected components
  - subgraph, induced subgraph
  - complete graph, bipartite graph, complete bipartite graph
  - forest, (free) tree, equivalent characterizations of trees
  - rooted tree, node, parent, child, ancestor, descendant, root, leaf, interior node
- **Analysis of algorithms**
  - $O$-notation, $\Omega$
  - Complexity classes $\mathcal{P}$ and $\mathcal{NP}$
  - NP-hard, NP-complete
  - reductions, how to prove NP-hardness
  - various NP-hard problems: Hamiltonian Cycle, 3-SAT (= 3-CNF-SAT), Clique, Vertex Cover, TSP
- **Basic graph algorithms**
  - Adjacency list representation of graphs, what operations can/can't be done in $O(1)$ time.
  - Breadth-first search (BFS), Depth-first search (DFS)
  - Analyzing graph algorithms, $\sum_{v \in V} \deg(v) = 2m$
  - Computing connected components in $O(m + n)$ time
  - Directed acyclic graphs (= dags), topological order
- **Various other topics**
  - pre-order and post-order traversal of a tree
  - complement of a graph
  - bucket sort
  - single-source shortest (weighted) path, Dijkstra's algorithm
  - dynamic programming, how to design an algorithm from a recursive formula, how to retrieve the solution.