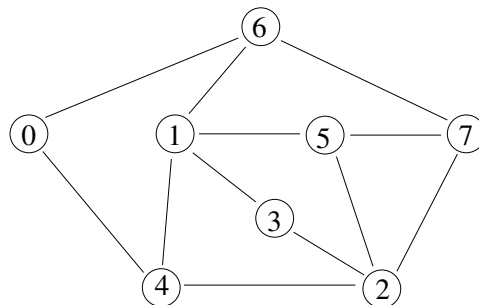


Übungen zu Informatik II Sommersemester 2007

13. Aufgabenblatt

Präsenzaufgabe 1 (Breitensuche)

- Ändern Sie den Algorithmus **BFS**(G, u) so ab, dass er ohne das Array c auskommt.
- Wenden Sie den Breitensuchalgorithmus der Vorlesung auf den folgenden Graphen an, dabei sei 0 der Startknoten:



Hausaufgabe 1 (Baumcharakterisierung) Zeigen Sie, dass die folgenden Aussagen für zusammenhängende, nicht leere ungerichtete Graphen G äquivalent sind:

- G ist ein Baum.
- Für jede Kante $e \in E_G$ gilt: Der Graph $(V_G, E_G \setminus \{e\})$ ist nicht zusammenhängend.
- Es gilt $|V_G| = |E_G| + 1$.

Hausaufgabe 2 (Graphen) In einem *ungerichteten Graphen* werden Objekte und Verbindungen zwischen ihnen gespeichert. Der ADT des ungerichteten Graphen (**UndirectedGraph**) soll die folgenden Operationen bereitstellen:

- hasEdge(i, j)**: Gibt an, ob eine Kante zwischen Knoten i und j vorhanden ist.
- arrayOfAdjacentVertices(i)**: Bestimmt ein Feld mit den zu Knoten i adjazenten Knoten.
- addEdge(i, j)**: Fügt die Kante zwischen Knoten i und j in den Graphen ein, sofern diese noch nicht vorhanden ist.

Geben Sie eine Implementierung **AdjacencyListUndirectedGraph** des ADT **UndirectedGraph** an.

Hausaufgabe 3 (Bipartite Graphen) Ein ungerichteter Graph G heißt *bipartit*, wenn es Knotenmengen V_1 und V_2 gibt mit:

- $V_G = V_1 \cup V_2$,
- $V_1 \cap V_2 = \emptyset$ und
- $\binom{V_i}{2} \cap E_G = \emptyset$ für $i \in \{1, 2\}$.

Geben Sie einen Algorithmus **BipartiteGraph**(G) an, der zu einem zusammenhängenden ungerichteten Graphen G entscheidet, ob er bipartit ist. Die Laufzeit Ihres Algorithmus soll $O(|V_G| + |E_G|)$ sein. Hinweis: Verwenden Sie eine ähnliche Strategie wie Sie bei der Breitensuche verwendet wird und färben Sie Knoten geschickt.

Ausgabe: 29.6.2007, **Abgabe:** bis 6.7.2007, 12 Uhr, im Schrein