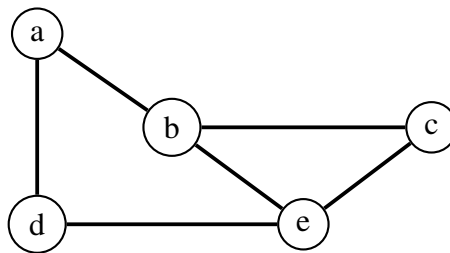


Algorithmen und Datenstrukturen

Aufgabe 1 Repräsentation von Graphen

a) Geben Sie zu dem nachfolgenden Graphen sowohl Adjazenz-Matrix als auch Adjazenz-Liste an!

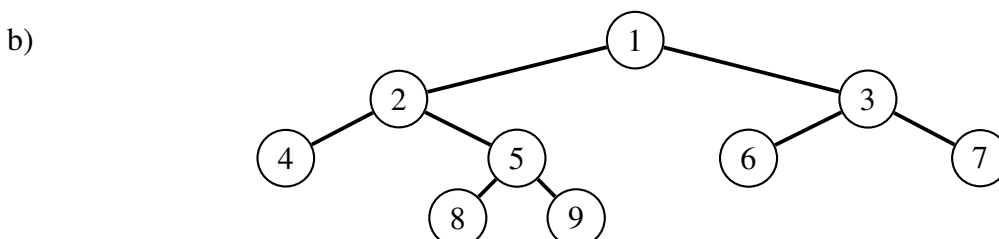
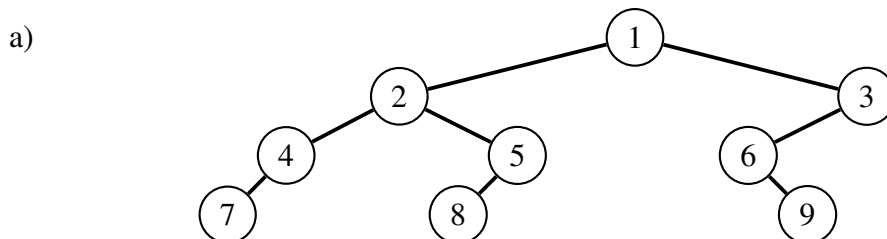


b) Skizzieren Sie den Graphen zu der folgenden Adjazenz-Matrix, und geben Sie die Adjazenz-Liste an!

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>
<i>a</i>	0	1	0	0	1	1
<i>b</i>	1	0	0	0	0	0
<i>c</i>	1	0	0	0	1	0
<i>d</i>	0	1	0	0	0	1
<i>e</i>	0	0	1	1	0	0
<i>f</i>	0	0	0	0	1	1

Aufgabe 2 Baum-Traversierung

Geben Sie für die folgenden Bäume jeweils die sequentiellen Listen an, die bei *Pre-Order*-, *In-Order*- und *Post-Order-Traversierung* entstehen!



Aufgabe 3 **Verifikation**

Gegeben sei ein Feld von Zahlen $X[0..n-1]$ der Länge n . Die Funktion $\mathbf{max}(X)$ soll den maximalen Wert des Feldes X berechnen. Hierzu verwenden wir die Maximums-Funktion zweier Zahlen a, b ,

$$\mathbf{max}(a, b) := \begin{cases} a & \text{falls } a \geq b, \\ b & \text{falls } a < b. \end{cases}$$

Nachfolgend der Pseudocode der Funktion $\mathbf{max}(X)$. Hierbei sind die in geschweiften Klammern eingeschlossenen Bezeichner $\{C_k\}$, wie aus Vorlesung und Übung gewohnt, Platzhalter für logische Bedingungen, die zur Verifikation benötigt werden.

```
max(X):  
{C0}  
    maxval = X[0];  
    i = 1;  
{C1}  
    while (i < n) {  
{C2}  
        maxval = max(maxval, X[i]);  
        i = i + 1;  
{C3}  
    }  
    return maxval;  
{C4}
```

Weiterhin definieren wir eine Hilfsfunktion f zur späteren Verwendung:

$$f(n) := \begin{cases} X[0] & \text{falls } n = 1, \\ \mathbf{max}(X[0..n-1]) = \mathbf{max}(f(n-1), X[n-1]) & \text{sonst.} \end{cases}$$

- Geben Sie jeweils eine geeignete Vor- und Nachbedingung C_0 bzw. C_4 der Funktion $\mathbf{max}(X)$ an!
- Geben Sie eine geeignete Schleifeninvariante P für die Schleife **while** $(i < n) \{ \dots \}$ an! Was ist die Vorbedingung C_1 dieser Schleife? *Hinweis:* verwenden Sie hier die Hilfsfunktion f .
- Welche drei Bedingungen müssen überprüft werden, damit die Korrektheit der **while** Schleife mittels der Invariante P gezeigt werden kann? Zeigen Sie, dass diese drei Bedingungen in diesem Fall erfüllt sind!

Hinweis: Sie können hierzu die Vorbedingungs-Platzhalter C_1, C_2, C_3, C_4 verwenden.