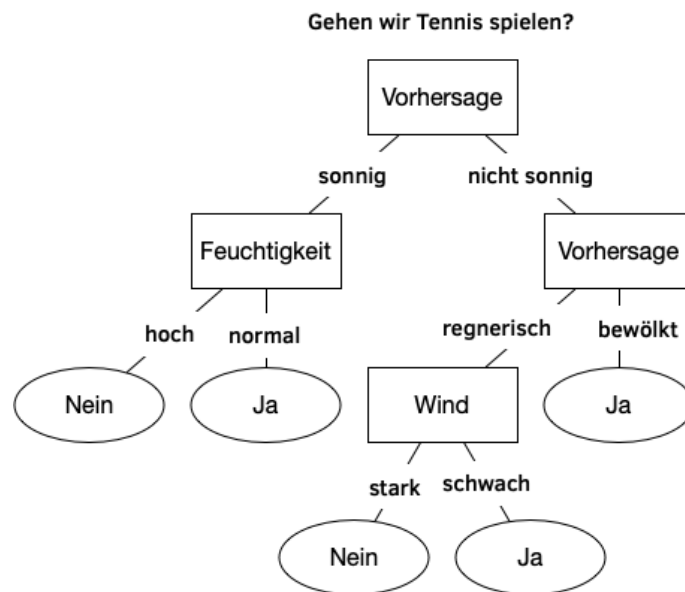


Nichtlineare Datenstrukturen: Bäume

Binäre Entscheidungsbäume



Aufgabe 1

Studiere die Dokumentation der Klasse `BinaryTree<ContentType>` und vergleiche sie mit der Verwendung der Klasse im Projekt `Entscheidungsbaum` im Tauschordner (in der Klasse `DecisionTreeBuilder`).

Versuche die folgenden Leitfragen für dich zu beantworten:

- Wieso wird mehr als ein Objekt der Klasse `BinaryTree` erstellt?
- In welcher Reihenfolge wird der Baum aufgebaut (werden die Objekte der Knoten instanziiert)?
- Wie werden die Inhalte vom generischen Typ `ContentType` (hier Unterklassen der Klasse `Decision`) im Baum gespeichert?

Aufgabe 2

Die Knoten enthalten Objekte der Unterklassen von `Decision`. Jede Unterklasse entscheidet für einen Datensatz, ob es im Baum links oder rechts weiter geht. Die Blattknoten enthalten keine Entscheidung, sondern die finale Festlegung einer Klasse für den Datensatz (hier „Ja“ oder „Nein“) in der Klasse `Classification`.

ANALYSIERE die Klassen `Decision` und `Classification` und BESCHREIBE ihre Funktionsweise. ERKLÄRE auch, wie die erste Entscheidung (in der Wurzel) umgesetzt ist (Klasse `DecisionTreeBuilder`, Zeile 17-19).

Aufgabe 3

IMPLEMENTIERE anhand des Beispiels im Konstruktor der Klasse `DecisionTreeBuilder` den vollständigen Entscheidungsbaum zur Tennis-Frage (vgl. Abbildung oben).

Teste deinen Baum mit dem *Objektinspektor* von BlueJ.

Aufgabe 4

Vervollständige die Methode `testeDatensatz` in `Entscheider`, so dass für einen Datensatz am Ende die korrekte Entscheidung (als `String`) zurückgegeben wird.

Teste mit verschiedenen Datensätzen.