

## Objektorientierung:

### Entwurf und Implementierung von Polynomen

Schwierigkeit 2

Entwerfen Sie eine Klasse Polynom, die ein reelwertiges (double) Polynom repräsentiert. Die Koeffizienten des Polynoms sollten als Feld im Konstruktor übergeben werden. Zwei Polynome sollen addiert werden können. Der Grad eines Polynoms soll sich bestimmt lassen. Ebenso soll die erste Ableitung eines Polynoms berechnet werden können. Achten Sie bei den Tests darauf auch Randbedingungen wie Null-Polynom oder Polynome mit 0-Koeffizienten beim höchsten Exponenten zu berücksichtigen. Zu guter Letzt, soll der Funktionswert an einer Stelle des Polynoms berechnet und zurückgegeben werden können.

Wenn Sie auch [diese Aufgabe zur Nullstellensuche](#) gemacht haben oder noch machen wollen, dann bietet es sich an ein Polynom das Interface einer stetigen Funktion implementieren zu lassen.

#### Lösung

### Entwurf und Implementierung von Windgeschwindigkeiten

Schwierigkeit 2

Windgeschwindigkeiten werden oft in Kilometer pro Stunde oder Knoten (Seemeilen pro Stunde) angegeben. Eine weitere Masseinheit ist die Beaufortskala (ganzzahlige Werte von 0-12). Entwerfen Sie eine Java-Klasse, mit der eine Windgeschwindigkeit unter Angabe des Werts in Kilometer pro Stunde erzeugt werden kann. Für eine Windgeschwindigkeit sollen auch Knoten und Beaufort-Wert ermittelt werden können. Ebenso soll überprüft werden, ob es windstill ist oder ein Orkan.

Windgeschwindigkeiten von weniger als 2 km/h gelten als windstill. Ab 120 km/h spricht man von einem Orkan. Eine Seemeile sind 1,852 Kilometer. Der Beaufortwert ist definiert durch  $v = 3,01 * B^{3/2}$ , wobei  $v$

die Windgeschwindigkeit in Kilometer pro Stunde ist und  $v$  kaufmännisch auf einen ganzzahligen Wert gerundet werden muss. Es gibt keinen Beaufortwert über 12.

Hinweis: Mit `Math.pow(a,b)` können Sie  $a^b$  berechnen.

[Lösung](#)

## Rationale Zahlen entwerfen und implementieren

Schwierigkeit 2

Rationale Zahlen sind Zahlen die sich als Bruch  $p / q$  einer ganzen Zahl  $p$  und einer natürlichen Zahl  $q$  darstellen lassen (mit  $q \neq 0$ ).

Entwerfen und implementieren Sie eine Klasse `RationaleZahl` deren Objekte rationale Zahlen repräsentieren, die sich addieren und multiplizieren lassen. Zähler und Nenner der vollständig *gekürzten* rationalen Zahl sollen zurückgegeben werden können. Ebenso soll der `double`-Wert der Zahl zurückgegeben werden können.

Einen Bruch können Sie vollständig kürzen, in dem Sie Zähler und Nenner durch den grössten gemeinsamen Teiler des Zählers und Nenners dividieren.

Achten Sie darauf, dass die Null eine eindeutige Darstellung hat. Ebenso darf nicht durch Null geteilt werden.

Die Klasse sollte einen Konstruktor enthalten, der Zähler und Nenner als `int`-Werte übergeben bekommt.

[Lösung](#)

## Entwerfen und Implementieren Sie chemische Elemente

Schwierigkeit 2

Die chemischen Elemente des Periodensystems klassifizieren alle Atome mit der selben Anzahl von Protonen: die Kernladungszahl oder auch Ordnungszahl. Jedes chemische Element hat einen symbolischen Namen. Dieser besteht in der Regel aus den ersten oder ersten beiden Anfangsbuchstaben des Lateinischen Namens.

Beispiele von chemischen Elementen sind:

H (hydrogenium) bezeichnet Wasserstoff und hat die Kernladungszahl 1.

O (oxygenium) bezeichnet Sauerstoff und hat die Kernladungszahl 8.

K ist Kalium, Kernladungszahl 19. Es ist ein *Alkalimetall*.

Zn ist Zink, Kernladungszahl 30. Es ist ein *Übergangsmetall*.

Ga ist Gallium, Kernladungszahl 31. Es ist ein *Metall*.

Im Periodensystem werden die chemischen Elemente unter anderem wie folgt unterteilt:

*Alkalimetalle* sind alle chemischen Elemente mit Kernladungszahl 3, 11, 19, 37, 55, 87

*Übergangsmetalle* sind alle chemischen Elemente mit Kernladungszahl zwischen (jeweils einschliesslich) 21 bis 31, 39 bis 48, 72 bis 80 und 104 bis 112.

*Metalle* sind alle chemischen Elemente mit Kernladungszahl 13, 49, 50, 81, 82, 83, 113, 114, 115, 116.

Entwerfen und implementieren Sie eine Klasse `ChemischesElement` mit den nötigen Eigenschaften und Methoden, um die drei Metalleigenschaft abzufragen (jeweils eine Methode). Verwenden Sie für letzteres kein `if/else` sondern:

Einmal ein `switch`

Einmal ein einziger Boolescher Ausdruck

Einmal ein `boolean`-Feld bei dem der Index die Kernladungszahl entspricht und den zugehörigen Wert zurückliefert. Das Feld ist `static` und kann im `static`-Initializer initialisiert werden.

Achten Sie darauf, die für eine Metalleigenschaft, die am besten passende der drei aufgeführten Implementierungsvarianten zu wählen. Wie unterscheiden sich die drei Implementierungsvarianten hinsichtlich Wartbarkeit und Ausführungsgeschwindigkeit?

Die Klasse soll noch Konstanten für die oben fünf genannten chemischen Elemente besitzen.

Die Werte der Objekte soll unveränderlich sein: sobald ein ChemischesElement erzeugt wurde, darf der Zustand nicht mehr geändert werden können.

Beachten Sie auch die Erweiterung dieser Aufgabe zum [Einlesen der Elemente aus einer Textdatei](#) und die Implementierung des abstrakten Datentyps Comparable zum [vergleichen zweier Chemischen Elemente](#)

[Lösung](#)