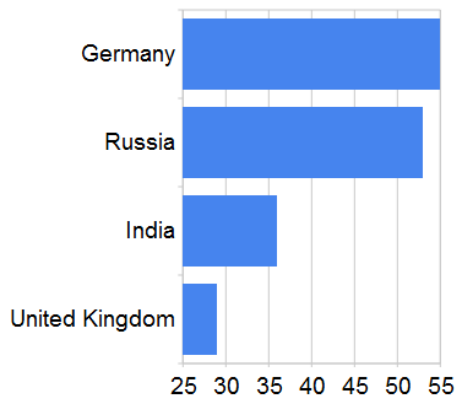


Prüfung 1: Lösungen

Programming: Visualisierung mit Google Code Playground (5 Punkte)

Aufgabe 1: Programmausführung simulieren (2 Punkte)

1 Punkt für korrekte Auswahl der Daten. 1 Punkt für qualitativ korrektes Diagramm: Entscheidend ist Sortierung, von oben nach unten oder umgekehrt spielt keine Rolle.



Aufgabe 2: Schnittstellen erklären (3 Punkte)

Ein Beispiel aus der Küche: Verschiedenste Arten von Dosenöffner können verschiedenste Arten von Dosen öffnen, weil die Konservendose quasi standardisiert ist. Das funktioniert, weil die "Schnittstelle" klar definiert ist: Dosenhersteller wissen, wie der Rand einer Dose beschaffen sein müssen, damit Dosenöffner den Rand aufschneiden können, und Hersteller von Dosenöffnern wissen, wie stark die Schneider sein müssen.

Damit Google uns die Daten visualisieren kann, müssen wir folgende Abmachungen einhalten:

- Die Daten müssen in tabellarischer Form abgelegt sein; die erste Zeile enthält die Spaltenbeschriftung.
- Wir müssen eine gültige Abfrage („Query“) formulieren, damit Google weiss, welche Daten ausgelesen werden sollen und wie sie sortiert werden sollen.
- Wir müssen Google sagen, was für ein Diagramm (BarChart) wir wollen.

Wenn wir unseren Teil der Abmachung einhalten, kann Googles Programm uns ein entsprechendes Diagramm zeichnen:

- Wir müssen uns nicht darum kümmern, wie Google die Daten holt.
- Wir müssen uns nicht darum kümmern, wie Google die Daten auswertet.
- Wir müssen uns nicht darum kümmern, wie Google die Daten auf den Bildschirm bringt.



1 Punkt für Verständlichkeit; 1 Punkt, wenn mindestens 2 der Dinge genannt sind, die der Client einhalten muss; 1 Punkt, wenn mindestens 1 der Dinge genannt ist, die Google erfüllt.

Automation: Schreiben Sie einen Automaten für Kara (5 Punkte)

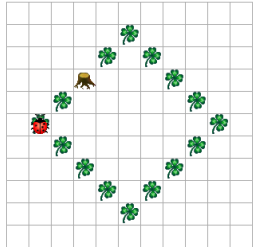
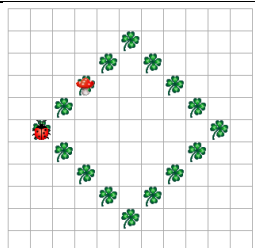
Aufgabe 1: Schreiben Sie Ihren Automaten in die Zustandstabelle unten

Die Lösung ist eine vereinfachte Version von „Pacman“ laufen: Kara weiss, dass immer diagonal rechts von ihm ein Kleeblatt sein muss; ist dort keines, muss er zurück zum letzten Kleeblatt und sich nach rechts drehen, und schon hat es wieder diagonal rechts von ihm ein Kleeblatt.

3 Punkte: 2 Punkte, wenn das Programm Kara korrekt laufen lässt; 1 Punkt für die korrekten Abbruchbedingungen.

laufen		Kara macht:	Nächster Zustand:
<input checked="" type="checkbox"/>	yes no	   	laufen
<input checked="" type="checkbox"/>	yes no	    	laufen
<input checked="" type="checkbox"/>	yes no		Stop
<input checked="" type="checkbox"/>	yes no		Stop

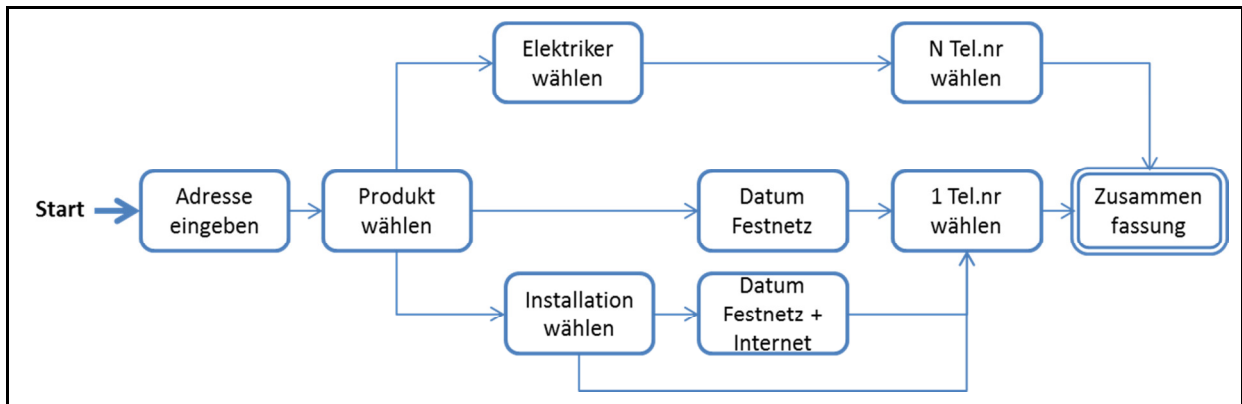
Aufgabe 2: Analysieren Sie Ihr Programm

	Kara läuft in den Baum hinein: Da Bäume nur angrenzend sein sollten, prüft das Programm nicht, ob der Weg von einem Kleeblatt zum nächsten frei ist.
	Da Pilze nicht auf Kleeblättern stehen sollten, verschiebt Kara den Pilz ein Feld nach rechts. Der Pilz steht anschliessend hinter Kara, und Kara läuft endlos das Quadrat ab.

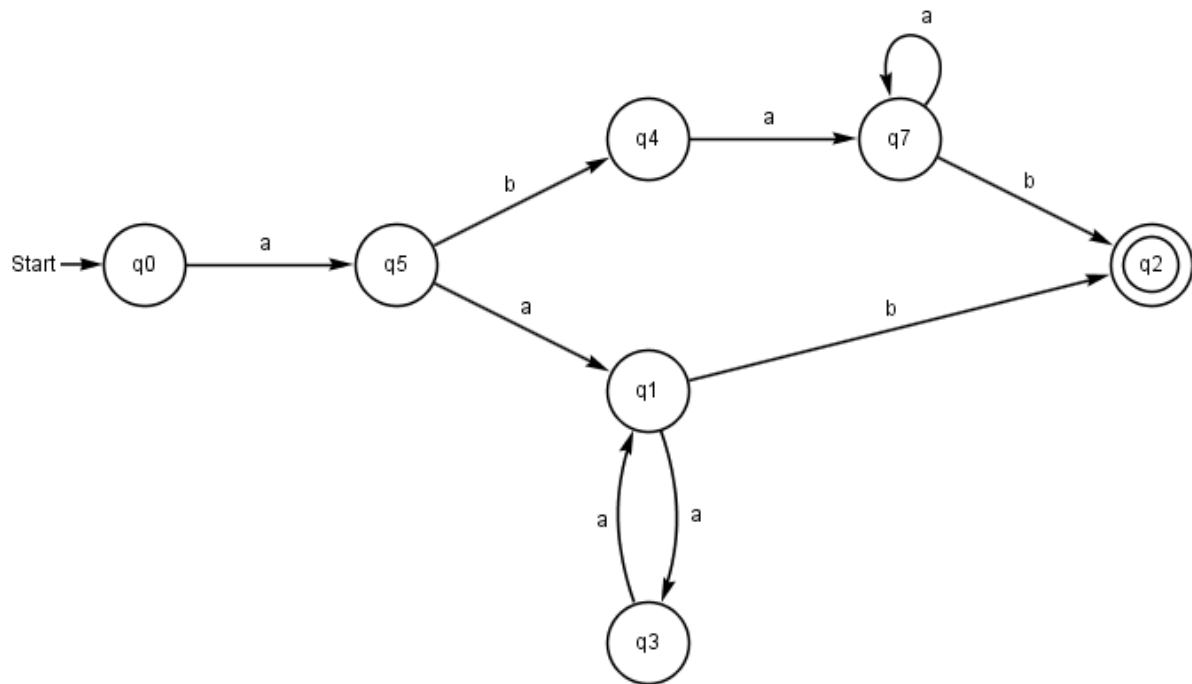
2 Punkte: 1 für Korrektheit, 1 für Verständlichkeit.

Modellieren mit Automaten (5 Punkte)

Mögliche Lösung. 3 Punkte für die Zustände (3 Punkte für ≥ 7 Zustände, 2 Punkte für ≥ 5 Zustände, 1 Punkte für ≥ 3 , sonst 0 Punkte), 2 Punkte für die Übergänge.



Computation: Reguläre Ausdrücke in endliche Automaten überführen (5 Punkte)



Computation: Reguläre Ausdrücke (5 Punkte)

Aufgabe (3 Punkte)

- 1 Punkt, wenn mindestens 4 Beispiele korrekt und unterschiedlich sind.
2 Punkte für Beschreibung: 1 Punkt für 0..255; 1 Punkt für Zahlenformat

Machen Sie drei möglichst unterschiedliche Beispiele von Zeichenketten, die von diesem Ausdruck akzeptiert werden:

255.255.255.255

0.0.0.0

222.33.1.0

Machen Sie drei möglichst unterschiedliche Beispiele von Zeichenketten, die von diesem Ausdruck *nicht* akzeptiert werden:

999.999.999.999

asdf

0..0.0.0

Beschreiben Sie präzise in Worten, welche Eingaben der Ausdruck akzeptiert:

Der Ausdruck akzeptiert Eingaben, die genau vier Zahlen zwischen 0 und 255 enthalten; diese Zahlen sind durch drei Punkte abgetrennt. Die vier Zahlen können dreistellig angegeben werden, selbst wenn es sich um Zahlen zwischen 0..100 handelt.

Hinweis: Bei diesen Eingaben handelt es sich um IP Adressen.

Aufgabe (2 Punkte)

- 1 Punkt, wenn `[1-5]|6` korrekt, und die Nachkommastellen optional sind.
1 Punkt, wenn korrekt `.5`, `.25`, `.75` akzeptiert wird.

Geben Sie einen regulären Ausdruck an, der sowohl ganze Noten als auch halbe Noten (zum Beispiel 5.5) und Viertelnoten (zum Beispiel 5.75, 5.25) akzeptiert:

`[1-5](\.[27]?5)?|6`

Hinweis: Wir ignorieren hier, dass die Eingabe von Anfang bis Ende eines Strings gehen müsste, dass der Ausdruck in der Praxis also `^[1-6](\.[27]?5)?$` lauten würde.