Exercise 1 (Operationen auf Mengen)

Gegeben sind die Mengen $Starters = \{salad, soup\}$, $Main = \{pizza, steak\}$, and $Dessert = \{fruit, icecream\}$, sowie die Relationen $Father = \{(adam, bert), (bert, carl)\}$ und $Car = \{(adam, audi), (bert, bmw), (carl, chevy)\}$.
Berechnen Sie:

1. $Starters \times Main \times Dessert$
2. $Father.Father$
3. $Father.Car$
4. $Car.Father$

Exercise 2 (Aussagen über Mengen)

Welche der folgenden Aussagen sind wahr? Begründen Sie jeweils, warum. Wenn eine Aussage falsch ist, dann zeigen Sie das mit einem Gegenbeispiel. Die Mengen A, B, C seien Teilmengen einer beliebigen endlichen Grundmenge U.

1. Wenn $A \cup B = A \cup C$ dann $B = C$.
2. Es gilt $A \subseteq B$ genau dann wenn $\bar{B} \subseteq \bar{A}$.
3. $A \times B = B \times A$
4. $|A \times B| = |A| \cdot |B|$

Exercise 3 (Mengen als Listen)

Mengen werden häufig mit Hilfe von Listen implementiert. Geben Sie in einer Programmiersprache ihrer Wahl oder in Pseudocode eine Funktion `boolean equals(list l1,list l2)` an, die genau dann `true` liefert, wenn die Mengen der Elemente der beiden Listen gleich sind. Gegeben sind die Funktionen `int head(list l)` und `list tail(list l)`, die für nichtleere Listen jeweils das erste Element und die Restliste liefern, sowie die Funktion `boolean empty(list l)`. Geben Sie möglichst einfachen, kurzen und offensichtlich korrekten Code an, die Laufzeit spielt keine Rolle. Definieren Sie sich geeignete Hilfsfunktionen.

Exercise 4 (Mengen von Natürlichen Zahlen als Boolean Arrays)

Eine endliche Menge S von natürlichen Zahlen kann man als Array `boolean[] s` repräsentieren, wie folgt: so dass gilt:

`s[n] = true` falls $n \in S$
`s[n] = false` falls $n \notin S$.

Die Menge $S = \{1, 3, 4\}$ ist dann durch folgendes Array repräsentiert:

```
s[0] = false
s[1] = true
s[2] = false
s[3] = true
s[4] = true  .
```

Geben Sie möglichst einfache Algorithmen in Pseudocode an für folgende Funktionen:

1. `boolean subset(boolean[] s, boolean[] t)`, die genau dann `true` liefert, wenn S eine Teilmenge von T ist
2. `boolean[] union(boolean[] s, boolean[] t)`, die die (Repräsentation der) Mengenvereinigung von S und T liefert.

Exercise 5 (Relationen als Boolean Arrays)

Eine endliche binäre Relation R auf den natürlichen Zahlen sei als ein zweidimensionales Array `boolean[][] r` repräsentiert, so dass gilt:

`r[m][n] = true` falls $(m, n) \in R$
`r[m][n] = false` falls $(m, n) \notin R$.

Geben Sie einen möglichst einfachen Algorithmus in Pseudocode an für die Funktion
`boolean[][] compose(boolean[][] r, boolean[][] t)` ,
die die (Repräsentation der) Komposition der Relationen R und T liefert.

Exercise 6 (Anzahl Relationen zwischen Mengen)

Gegeben seien zwei Mengen A und B mit $|A| = m$ und $|B| = n$. Wieviele Relationen zwischen A und B gibt es? Betrachten Sie Beispiele. Begründen Sie Ihre Antwort.