Aufgabe 1: Sind die folgenden Abbildungen jeweils injektiv, surjektiv und/oder bijektiv?

(a) $f_1(x) = \sqrt{x}$, mit $f_1 : \mathbb{R}^+ \to \mathbb{R}^+$

(b) $f_2(x) = x^2$, mit $f_2 : \mathbb{R} \to \mathbb{R}$

(c) $f_3(x) = |x|$, mit $f_3 : \mathbb{R} \to \mathbb{R}$

(d) $f_4(x) = 3x$, mit $f_4 : \mathbb{R} \to \mathbb{R}$

(e) $f = \{(1,2),(2,2),(3,4),(5,3)\}$ mit $f : A \to A$ und $A = \{1,\ldots,5\}$.

Schwierigkeit: 4

Aufgabe 2: Wählen Sie eine Relation auf der Grundmenge $A = \{a,b,c,d\}$, die ...

(a) ... weder symmetrisch noch anti-symmetrisch ist.

(b) ... sowohl symmetrisch als auch anti-symmetisch ist.

(c) ... transitiv ist, und die Tupel $(b,c),(c,a),(a,b),(b,c)$ enthält.

(d) ... die transitiv und reflexiv ist und möglichst wenig Elemente enthält.

Schwierigkeit: 3

Aufgabe 3: "Übersetzen" Sie die folgenden natürlich-sprachlichen Aussagen in die Sprache der Mathematik , d.h. unter Verwendung der Symbole $\forall, \exists, \wedge, \vee, \Rightarrow, \Longleftrightarrow, \neg, \ldots, \in, \mathbb{N}, \mathbb{Z}, \mathbb{R}, |\ldots|$

(a) Wenn man aus der Aussage A die Aussage B folgern kann und aus der Aussage B die Aussage C folgern kann, dann kann man auch aus der Aussage A die Aussage C folgern. (Das ist übrigens die Formulierung der Transititivität der Implikation).

(b) Aus der Aussage A folgt die Ausssage B genau dann, wenn aus der Aussage "nicht B" die Aussage "nicht A" folgt.

(c) Nicht alle natürlichen Zahlen sind ungerade.

(d) Es gibt keine natürliche Zahl, die sowohl gerade als auch ungerade ist.

Hiweis: Sie düfen die Prädikate *even* und *odd* verwenden, die jede natürliche Zahl auf einen Wahrheitswert abbildet, der angibt ob diese gerade oder ungerade ist. **Schwierigkeit: 6**

Aufgabe 4: Formulieren Sie die folgenden Aussagen (unabhängig davon, ob die falsch oder wahr sind) in der "Sprache der Mathematik", d.h. unter Verwendung der Symbole $=, |\ldots|, \forall, \exists, \wedge, \vee, \Rightarrow, \Longleftrightarrow, \neg, \in, \mathbb{N}, \mathbb{Z}, \mathbb{R}$

(a) "Falls der Schnitt zweier Mengen nicht leer ist, dann enthalten Sie mindestens zwei gemeinsame Elemente"

(b) "Alle natürlichen Zahlen sind entweder gleich, oder haben einen Abstand von mindestens 1."

(c) "Genau eine der Mengen A oder B ist leer."

(d) "Die Mengen A und B enthalten genau ein gemeinsames Element"

Schwierigkeit: 6

Aufgabe 5: Beweisen Sie die folgenden Aussagen durch direkten Beweis.

(a) "Sei n eine gerade natürliche Zahl. Dann ist auch n^2 gerade."

(b) "Die Summe von zwei ungerade Zahlen ist gerade."

(c) "Sei n eine beliebige natürliche Zahl. Dann ist $n^2 + n$ gerade."

Schwierigkeit: 6

Aufgabe 6: Beweisen oder widerlegen Sie folgende Aussage:
„Zwei Mengen sind genau dann identisch, wenn Ihre Potenzmengen identisch sind.“ oder formaler geschrieben:

$$A = B \quad \Longleftrightarrow \quad \mathcal{P}(A) = \mathcal{P}(B)$$

Achten Sie darauf, beiden "Richtungen" zu beweisen, also sowohl „$\Longleftarrow$“ als auch „$\Longrightarrow$“; wollen Sie die Aussage widerlegen, genügt es natürlich, nur eine Richtung zu widerlegen. **Schwierigkeit: 10**

Aufgabe 7: Berechnen Sie (ausnahmsweise) die konkreten Werte der folgenden Ausdrücke:

(a) $\{x \mid x \in \{1, 2, \ldots, 10\},\ ggT(x, 6) = 1\ \}$

(b) $\{x \mid x \text{ teilt } 30 \text{ und } x \text{ ist Primzahl } \}$

(c) $4021923^3 \mod 9$

(d) $1123456789 \mod 3$

(e) $|\{x \mid x \mod 3 = 1 \text{ und } x \mod 7 = 1 \text{ und } x \in \{1, \ldots, 20\}\}|$

(f) $\mathcal{P}(\{1, \{1\}\})$

(g) $\{1, 2, 3, 4, 5\} \cap \mathcal{P}(\{1, 2, 3, 4, 5\})$

(h) $\{x \mid x \in \mathbb{N} \wedge x \leq 100 \wedge x \equiv 1 \pmod{10}\}$

(i) $\{x \mid x \in \mathcal{P}(\{1, \ldots, 10\})\ \wedge\ |x| = 3\ \wedge 4 \in x\ \wedge 1 \in x\ \}$

Schwierigkeit: 4

Aufgabe 8: Gegeben sei das Alphabet $A = \{a, b, c\}$.

(a) Wie viele Wörter der Länge 10 kann man aus den drei Buchstaben bilden?

(b) Wie viele Wörter der Länge 10 gibt es, die genau 2 mal a enthalten?

(c) Wie viele Wörter der Länge 10 gibt es, die genau 3 mal a und genau drei mal b enthalten?

Schwierigkeit: 3

Aufgabe 9: Aus einer Gruppe von 20 Informatik-Studenten (davon 13 männliche und 7 weibliche) werden zufällig 4 ausgewählt, die zusammen ein Projekt bearbeiten sollen.

(a) Wie viele verschiedene Möglichkeiten gibt es insgesamt, 4 Studenten auszuwählen?

(b) Wie viele verschiedenen Möglichkeiten gibt es, 4 Studenten auszuwählen, so dass 2 davon männlich und 2 davon weiblich sind?

(c) Wie groß ist die Wahrscheinlichkeit, dass der zuerst ausgewählte Student männlich ist?

(d) Wie groß ist die Wahrscheinlichkeit, dass alle ausgewählten Studenten männlich sind?

Schwierigkeit: 5

Aufgabe 10: Ein Dominostein ist ein Rechteck bestehend aus zwei Quadraten, wobei in jedem Quadrat durch Punkte eine Zahl von 1 bis n dargestellt ist.
Wie viel verschiedene Dominosteine gibt es? (Abhängig von n) **Schwierigkeit: 3**

Aufgabe 11:

(a) Zeigen Sie über vollständige Induktion, dass für alle $n \in \mathbb{N}$ gilt, dass $4n^3 - n$ immer durch 3 teilbar ist.

(b) Zeigen Sie über vollständige Induktion, dass die Ungleichung

$$(1+x)^n > 1 + n \cdot x$$

falls $x > 0$ ist für alle $n \in \{2, 3, \ldots\}$ gilt.

Schwierigkeit: 4

Aufgabe 12:

(a) Wie viele verschiedene „Wörter“ lassen sich durch Umstellen der Buchstaben aus den Wörtern `Mississippi` bilden?

(b) Angenommen, ein Wort aus den in `Mississippi` enthaltenen Buchstaben wird zufällig gebildet. Wie groß ist die Wahrscheinlichkeit, dass alle Buchstaben `i` direkt hintereinander stehen?

(c) Angenommen, ein Wort aus den in `Mississippi` enthaltenen Buchstaben wird zufällig gebildet. Wie groß ist die Wahrscheinlichkeit, dass am Wortende der Buchstabe `i` steht?

Schwierigkeit: 5

Aufgabe 13: Sie würfeln 5 mal hintereinander mit einem 6-seitigen Würfel. Wie groß ist die Wahrscheinlichkeit, dass ...

(a) ... genau zweimal eine 1 fällt?

(b) ... alle Würfe gleich sind?

(c) ... alle Würfe verschieden sind?

Schwierigkeit: 2

Aufgabe 14: Mit zwei idealen Würfeln werde einmal gewürfelt. Wir wählen also als Wahrscheinlichkeitsraum $\Omega = \{1, \ldots, 6\}^2$. Gegeben seien die folgenden Ereignisse:
A: Die Augensumme ist gerade
B: Es fällt ein Pasch
C: Es fällt mind. eine 6

(a) Geben Sie die Ereignisse A, B und C *konkret* als Teilmengen von Ω an.

(b) Berechnen Sie $Pr[A], Pr[B]$ und $Pr[C]$

(c) Berechnen Sie $Pr[A|\overline{B}]$, $Pr[C|B]$ und $Pr[A \cap B \cap C]$.

Schwierigkeit: 2

Aufgabe 15:

(a) Zeigen Sie über vollständige Induktion, dass für alle $n \in \mathbb{N}$ gilt, dass

$$3 + 7 + 11 + \ldots + (4n - 1) = 2n^2 + n$$

(b) Die Fibonacci-Folge ist folgendermaßen definiert:

$$F_1 = 1, \quad F_2 = 1, \quad F_{n+1} = F_n + F_{n-1}$$

1. Geben Sie die ersten 6 Folgeglieder F_1 bis F_6 der Fibonacci-Folge an.
2. Zeigen Sie durch vollständige Induktion, dass gilt:

$$1 + F_1 + F_2 + \ldots + F_n = F_{n+2}$$

Schwierigkeit: 6

Aufgabe 16: *Das Stein-Schere-Papier-Spiel* Das Spiel wird ausschließlich mit den Händen gespielt. Handhaltungen werden Symbole zugeordnet, die sich wechselseitig “schlagen” können. Die drei Hauptfiguren sind Schere, Stein und Papier. Das Papier wird durch eine flache Hand mit ungespreizten Fingern dargestellt, das Symbol der Schere ist der gespreizte Zeige- und Mittelfinger und der Stein wird durch eine Faust symbolisiert. Die Wertigkeit der Symbole gegeneinander ergibt sich aus dem jeweils Dargestellten: Die Schere schneidet das Papier (= Schere gewinnt), das Papier wickelt den Stein ein (= Papier gewinnt) und der Stein macht die Schere stumpf (= Stein gewinnt). Angenommen zwei Spieler, nennen wir sie Spieler A und Spieler B, spielen gegeneinander.
Entscheiden sich beide Spieler für dasselbe Symbol, wird das Spiel als “Unentschieden” gewertet und wiederholt.

(a) Wie groß ist die Wkeit, dass beim ersten Versuch einer der beiden Spieler gewinnt (d.h. dass das Spiel nicht wiederholt werden muss)?

(b) Wie groß ist die Wkeit, dass die beiden ersten Versuche “unentschieden” sind und erst der dritte Versuch eine Entscheidung bringt.

(c) Das Spiel wird 100 mal gespielt. Wie groß ist die Wkeit, dass die beiden Spieler dabei kein einziges Mal dieselben Symbole gewählt haben?

(d) Das Spiel wird 10 mal gespielt. Wie groß ist die Wkeit, dass die beiden Spieler dabei 3 Mal “unentschieden” gespielt haben?

Schwierigkeit: 3

Aufgabe 17:

(a) Wie viele 4-elementige Teilmengen von $\{1, \ldots, 100\}$ gibt es, die die 1 und die 10 enthalten?

(b) Wie viele 4-elementige Teilmengen von $\{1, \ldots, 100\}$ gibt es, die entweder die 1 oder die 10 enthalten, aber nicht beide gleichzeitig?

(c) Wie viele 10-elememtige Teilmengen von $\{1, \ldots, 10\}$ gibt es, die die 1 und die 10 enthalten?

(d) Wie viele 4-elementige Teilmengen von $\{1, \ldots, 100\}$ gibt es, die die 1 oder die 10 enthalten? (es können evtl auch beide gleichzeitig in der Teilmenge sein)

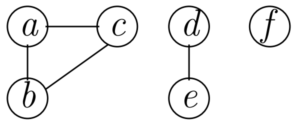
Schwierigkeit: 4

Algorithmik

Aufgabe 18: Verwenden Sie die Breitensuche, um alle Zusammenhangskomponenten eines Graphen zu bestimmen; implementieren Sie eine entsprechende Funktion `allComps`, die eine Liste aller Zusammenhangskomponenten zurückliefert. Eine Zusammenhangskomponenten soll hierbei wiederum als Menge (etwa repräsentiert als Liste oder `set`-Objekt) von Knoten repräsentiert sein, die die entsprechende Zusammenhangskomponente bilden. Beispiel:

```
>>> allComps(graph)
>>> [[a,b,c],[d,e],[f]]
```

`graph=`



Schwierigkeit: 8

Aufgabe 19: Erweitern Sie die in der Vorlesung vorgestellte Klasse `Graph` um die folgenden Methoden:

(a) Erweitern Sie die Klasse `Graph` um die Methode `Graph.w(i,j)`, die das Gewicht der Kante `(i,j)` zurückliefert (bzw. `None`, falls die Kante kein Gewicht besitzt).

(b) Eine Methode `Graph.isPath(vs)`, die eine Knotenliste `vs` übergeben bekommt und prüft, ob es sich hierbei um einen Pfad handelt.

(c) Eine Methode `Graph.pathVal(vs)`, die eine Knotenliste `vs` übergeben bekommt. Handelt es sich dabei um einen gültigen Pfad, so wird der „Wert“ dieses Pfades (d. h. die Summe der Gewichte der Kanten des Pfades) zurückgeliefert. Andernfalls soll der Wert ∞ (in Python: `float('inf')`) zurückgeliefert werden. Verwenden Sie hierbei das folgende „Gerüst“ und fügen Sie an der mit „...“ markierten Stelle die passende Listenkomprehension ein.

```
def pathVal(self, xs):
  if len(xs)<2: return 0
  return sum([...])
```

Schwierigkeit: 6

Aufgabe 20: Implementieren Sie eine (rekursive) Python-Funktion `bh2lst(bh)`, die die Schlüsselwerte eines Binomial-Heaps in eine Liste umwandelt. **Schwierigkeit: 9**

Aufgabe 21: Gegeben sei die folgende List von Pythonwerten:

```
[88, 1, 72, 60, 86, 82, 41, 69, 90, 37, 51, 61, 12, 27, 4]
```

(a) Fügen Sie die Werte dieser Liste nacheinander in einen zunächst leeren binären Heap ein.

(b) Fügen Sie die Werte dieser Liste nacheinander in einen zunächst leeren Binomial-Heap ein.

Schwierigkeit: 4

Aufgabe 22: Gegeben Sei folgende Klassendefinition, die Tries repräsentieren soll:

```
class Trie(object):
  def __init__(self):
    self.children = {}
    self.val = None
```

(a) Geben Sie eine rekursive Implementierung einer Suchmethode `Trie.search(key)` an.

(b) Geben Sie eine iterative Implenentierung einer Suchmethide `Trie.search(key)` an.

Schwierigkeit: 7

Aufgabe 23: Gegeben Sei folgende Klassendefinition, die Tries repräsentieren soll:

```
class Trie(object):
  def __init__(self):
    self.children = {}
    self.val = None
```

(a) Geben Sie eine rekursive Implementierung einer Einfügemetode `Trie.insert(key, val)` an.

(b) Geben Sie eine iterative Implenentierung einer Einfügemethode `Trie.insert(key, val)` an.

Schwierigkeit: 7

Aufgabe 24: Gegeben seien die folgenden Klassendefinitionen zur Repräsentation von Skiplisten:

```
class SLEntry(object):
  def __init__(self, key, ptrs=[], val=None):
    self.key = key ; self.ptrs = ptrs ; self.val = val

class SkipList(object):
  def __init__(self):
    self.tail = SLEntry(Infty)
    self.head = SLEntry(None,[self.tail for _ in range(MaxHeight+1)])
    self.height = 0
```

(a) Schreiben Sie eine Funktion `numHeights(h)`, die die Anzahl der Elemente mit Höhe `n` zurückliefert.

(b) Schreiben Sie eine Funktion `avgHeight(s)`, die die durchschnittliche Höhe eines Elementes der Skip-Liste `s` berechnet.

Schwierigkeit: 7

Aufgabe 25: Gegeben sei die folgende Liste von Werten:
`[88, 1, 72, 60, 86, 82, 41, 69, 90, 37, 51, 61, 12, 27, 4]`

(a) Fügen Sie die Werte der Liste nacheinander in einen anfänglich leeren AVL-Baum ein und führen Sie immer wenn notwendig die entsprechenden Rotations-Operationen aus.

(b) Welche Höhe hat der entstandene Baum? Welche Höhe könnte ein AVL-Baum mit 15 Werten maximal/minimal haben?

Schwierigkeit: 5

Aufgabe 26: Das folgende Sortierverfahren zum Sortieren einer Liste von Zahlen nennt man Bubblesort:
In jedem Durchlauf durch die Liste werden zwei benachbarte Werte genau dann getauscht, wenn der linke Wert größer ist als der rechte Wert. Es werden so viele Durchläufe ausgeführt, bis keine Vertauschungen mehr vorgenommen werden müssen.

(a) Implementieren Sie Bubblesort als Python-Prozedur

(b) Welche Laufzeit hat Bubblesort im Best-Case bzw. im Worst-Case.

Schwierigkeit: 7

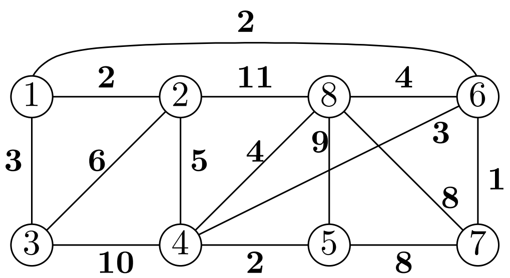
Aufgabe 27: Gegeben sei die folgende Liste von Werten:
`[88, 1, 72, 60, 86, 82, 41, 69, 90, 37, 51, 61, 12, 27, 4]`

(a) Sortieren Sie diese Werte so, wie es der Quicksort-Algorithmus machen würde. Stellen Sie die Sortierschritte graphisch als Baum dar.

(b) Sortieren Sie diese Werte so, wie es der Mergesort-Algorithmus machen würde. Stellen Sie die Sortierschritte graphisch als Baum dar.

Schwierigkeit: 4

Aufgabe 28: Führen Sie die ersten 4 Durchläufe (der äußersten Schleife) des Warshall-Algorithmus für den folgenden Graphen durch:



Schwierigkeit: 5