

## 4. PNK-Übungsblatt

### Aufgabe 12 Potenzberechnung

Berechnen Sie die Potenz  $pot(x, y) = x^y$  zweier natürlicher Zahlen  $x > 0$  und  $y \geq 0$  durch wiederholte Multiplikation. Implementieren Sie hierfür einen Algorithmus.

### Aufgabe 13 Gleichungsrätsel

Ein Informatiker möchte ein Rätsel für seine Kommilitonen erstellen. Dabei sollen Rechenzeichen zwischen den Zahlen von 1 bis 6 gesetzt werden, so dass das Ergebnis gleich 42 ist. Die Reihenfolge, in der die Zahlen stehen müssen, ist festgelegt mit: 6,5,4,3,2,1

Als Rechenzeichen sollen die Grundrechenarten '+', '-', '\*', '/' genügen. Für die Berechnung sollen die bekannten mathematischen Regeln gelten, also Division durch 0 ist verboten, Punktrechnung geht vor Strichrechnung. Klammern sind nicht erlaubt.

Zum Beispiel ergäbe  $6 * 5 / 4 + 3 * 2 - 1$  gerade  $12\frac{1}{2}$  und wäre daher keine Lösung des Problems.

- Finden Sie eine Lösung für dieses Problem.
- Implementieren Sie einen Algorithmus, der Ihnen alle Lösungen für das gegebene Problem ausgibt.
- Erweitern Sie Ihren Algorithmus, so dass er als zusätzliche Rechenart auch die Potenzrechnung  $x^y$  (in python: `x**y`) akzeptiert.

Hinweise: Die Anzahl der Möglichkeiten ist für aktuelle Computer „überschaubar“. Eine sogenannte *Brute-Force-Lösung*, bei der alle Möglichkeiten durchprobiert werden, ist also naheliegend.

Die Python-Funktion `eval(x)`, berechnet für einen String  $x$ , den zum Term  $x$  äquivalenten Wert, sie führt also die Berechnung des Wertes durch.

### Aufgabe 14 Das Sieb des Eratosthenes

Mit Hilfe des Siebs des Eratosthenes können sehr effizient alle Primzahlen bis zu einer gegebenen Obergrenze gefunden („herausgesiebt“) werden. Stellen Sie sich eine Tafel vor, auf der alle natürlichen Zahlen bis zu einer gewählten Obergrenze angeschrieben sind. Auf dieser werden folgendermaßen alle Nicht-Primzahlen gestrichen:

Beginnen Sie mit der 2. Alle Vielfachen von 2, beginnend mit 4, sind keine Primzahlen. Streichen Sie diese Zahlen auf der Tafel aus. Nehmen Sie nun die 3. Streichen Sie alle Vielfachen von 3, beginnend mit 6. Nun die 4. Sie ist bereits von der Tafel gestrichen und kann daher übersprungen werden. (Alle Vielfachen von 4 sind auch Vielfache von 2 und daher schon gestrichen.) Streichen Sie die Vielfachen von 5, beginnend mit 10.

Indem Sie so fortfahren und am Ende noch die 1 streichen, die per Definition keine Primzahl ist, bleiben auf der Tafel nur die Primzahlen ungestrichen. Schreiben Sie ein Programm, das diesen Algorithmus benutzt, um alle Primzahlen bis 1000 zu bestimmen.