Aufgabe 1

- Undo-Funktion in Anwendungsprogrammen (1 P)
- Prüfen, ob ein Klammerterm korrekt ist (1 P)
- History-Funktion in Browsern
- ...

Aufgabe 2

```
class Stack:

    def __init__(self):
        self.items = []

    def push(self, item):
        self.items.append(item) # 0.5P

    def pop(self):
        return self.items.pop() # 0.5P

    def peek(self):
        return self.items[-1]   # 0.5P

    def is_empty(self):
        return self.items == [] # 0.5P
```

Aufgabe 3

```
from stack import Stack

s = Stack()
s.push(3)
s.push(5)
s.push(1)
a = s.pop()
b = s.peek()
s.push(8)

print(a)
print(b)
print(s.size())
```

```
1
5
3
```

Aufgabe 4

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5 ~~6~~ ~~7~~ 8 ~~9~~

2 1 3 0 4 6 7 9 False

(b) ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

2 3 1 4 6 5 8 7 0 9 True

Aufgabe 5

(a) M D - C F + * ⇒ (M - D) * (C + F)

(b) - * P S + X Y ⇒ P * S - (X + Y)

Aufgabe 6

Input:

~~P~~	~~-~~	~~(~~	~~C~~	~~+~~	~~F~~	~~)~~	~~*~~	~~A~~	~~-~~	~~E~~	~~+~~	~~W~~	~~*~~	~~M~~

Stack: (nimmt → zu)

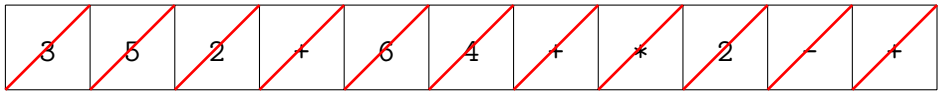
~~-~~	~~(~~	~~+~~	~~)~~	~~*~~	~~-~~	~~+~~	~~*~~							

Output:

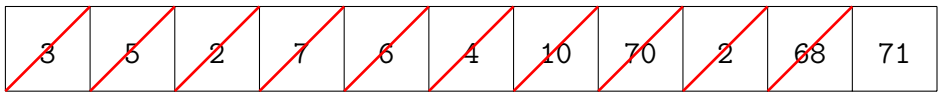
P	C	F	+	A	*	-	E	-	W	M	*	+		

Aufgabe 7

Input:



Stack: (nimmt → zu)



Output: 71

Aufgabe 8

(a) Ein Algorithmus ist ein Lösungsverfahren, das

- endlich
- deterministisch (der nächsten Schritt ist eindeutig bestimmt)
- effektiv (die Wirkung jeder Anweisung ist eindeutig)

ist.

(b)
- Ist ein Kochrezept ein Lösungsverfahren? Ja
- Ist ein Kochrezept endlich? Ja
- Ist ein Kochrezept deterministisch? Ja, wenn die Reihenfolge vorgegeben ist.
- Ist ein Kochrezept effektiv? Nicht unbedingt, wenn z.B. Mengenangaben oder Backzeiten nicht genau angegeben werden („Salz nach belieben“ oder „40–50 Minuten bei mittlerer Hitze backen“).

Aufgabe 9

$\text{ggT}(35, 14) = (21, 14) = (7, 14) \overset{\text{s}}{=} (14, 7) = (7, 7) = (0, 7) \overset{\text{s}}{=} (7, 0) = (7, 0) = 7$

Aufgabe 10

$\text{ggT}(35, 14) = (14, 7) = (7, 0) = 7$

Aufgabe 11

(a) $T(n) = 4n^2 + 2n + 5n^3 + 1 \in O(n^3)$

Kontrolle: $\lim\limits_{n\to\infty} \dfrac{5n^3 + 4n^2 + 2n + 1}{n^3} = \lim\limits_{n\to\infty} \left(5 + \dfrac{4}{n} + \dfrac{2}{n^2} + \dfrac{1}{n^3}\right) = 5 < \infty$

(b) $T(n) = 5 \cdot 3^{n-1} = 5 \cdot 3^n \cdot 3^{-1} = \frac{5}{3} \cdot 3^n \in O(3^n)$

Kontrolle: $\lim\limits_{n\to\infty} \dfrac{5 \cdot 3^{n-1}}{3^n} = \lim\limits_{n\to\infty} \left(5 \cdot 3^{-1}\right) = \dfrac{5}{3} < \infty$

(c) $T(n) = 27 \in O(1)$

Kontrolle: $\lim\limits_{n\to\infty} \dfrac{27}{1} = 27 < \infty$

(d) $T(n) = (4n^2 + 3)(5n - 4)(7n^3 - 6) \in O(n^6)$

(e) Logarithmengesetze:

$T(n) = \log_2(4n^2) = \log_2(4) + 2\log_2(n) \in O(\log_2(n))$