

**Name:** .....

**Dauer:** 45 Minuten

**Umfang:** Stacks, Infix-, Präfix- und Postfix-Ausdrücke, Algorithmen, ggT, Laufzeitkomplexität

**Hilfsmittel:** Python Cheat Sheet

### **Aufgabe 1 (2P)**

Beschreibe zwei verschiedene Anwendungen von Stacks.

### **Aufgabe 2 (2P)**

Vervollständige die mit ... gekennzeichneten Methoden der Klasse `Stack`.

```
class Stack:
```

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

```
    def push(self, item):  
        ...
```

```
    def pop(self):  
        ...
```

```
    def peek(self):  
        ...
```

```
    def is_empty(self):  
        ...
```

### Aufgabe 3 (2P)

Welche Ausgaben macht das folgende Python-Modul?

```
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())
```

### Aufgabe 4 (2P)

Angenommen, ein Programm führt eine Folge von push- und pop-Operationen auf einem Stack aus, wobei die push-Operationen die ganzen Zahlen von 0 bis 9 in dieser Reihenfolge auf den Stack ablegen. Durch die Folge der pop-Operationen erhalten wir die folgenden Output-Sequenzen. Welche davon kann es nicht geben?

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

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

### Aufgabe 5 (2P)

Stelle den Ausdruck korrekt in der Infix-Form dar.

(a)  $M D - C F + *$

(b)  $- * P S + X Y$

### Aufgabe 6 (4P)

Wandle den Term mit dem Infix-to-Postfix-Algorithmus in die Postfix-Darstellung um.

Input:

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

Stack: (nimmt → zu)

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Output:

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

### Aufgabe 7 (2P)

Berechne den Wert des Postfix-Terms mit dem zugehörigen Algorithmus.

Input:

3	5	2	+	6	4	+	*	2	-	+
---	---	---	---	---	---	---	---	---	---	---

Stack: (nimmt → zu)

--	--	--	--	--	--	--	--	--	--	--

Output:

### Aufgabe 8 (2P)

Eine Person geht mit einer Einkaufsliste in den Supermarkt, sucht die Artikel auf der Liste, legt sie in den Einkaufswagen und streicht sie von der Liste ab. Untersuche kritisch, ob es sich bei dieser Tätigkeit um einen Algorithmus handelt und verwende Fachbegriffe.

### Aufgabe 9 (2P)

Zeige schrittweise, wie der klassische euklidische Algorithmus  $\text{ggT}(30, 12)$  berechnet.

### Aufgabe 10 (2P)

Zeige schrittweise, wie der moderne euklidische Algorithmus  $\text{ggT}(32, 18)$  berechnet.

### Aufgabe 11 (2P)

Bestimme nachvollziehbar die Laufzeitkomplexität der folgenden Python-Funktion in Abhängigkeit des Parameters  $n$ .

```
def function(n):
    s = 0
    for i in range(0, n):
        for j in range(0, n):
            s += i + j
        for k in range(0, 2*n):
            s += k
    return s
```

### Aufgabe 12 (1P)

Ein Programm mit der Laufzeitkomplexität  $O(n^2)$  liegt, benötigt für die Verarbeitung von  $n = 100$  Eingaben etwa 3 Sekunden. Wie lange benötigt dasselbe Programm auf demselben Computer für  $n = 300$  Eingaben?