

Informatik (Übungsprüfung)

4.4.2025

Aufgabe 1 (2P)

Beschreibe zwei verschiedene Anwendungen von Stacks.

Aufgabe 1 (2P)

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

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 2 (2P)

```
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 (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 3 (2P)

```
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 (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 4 (2P)

(a)

Aufgabe 4 (2P)

(a) 0

Aufgabe 4 (2P)

(a) 0 1

Aufgabe 4 (2P)

(a) 0 1 2

Aufgabe 4 (2P)

(a) 0 1 ~~2~~

2

Aufgabe 4 (2P)

$$(a) \begin{array}{ccc} 0 & \cancel{1} & \cancel{2} \\ 2 & 1 & \end{array}$$

Aufgabe 4 (2P)

$$(a) \begin{array}{cccc} 0 & \cancel{1} & \cancel{2} & 3 \\ & 2 & 1 & \end{array}$$

Aufgabe 4 (2P)

$$(a) \quad \begin{array}{cccc} 0 & \cancel{1} & \cancel{2} & \cancel{3} \\ 2 & 1 & 3 & \end{array}$$

Aufgabe 4 (2P)

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~
2 1 3 0

Aufgabe 4 (2P)

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~ 4
2 1 3 0

Aufgabe 4 (2P)

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~
2 1 3 0 4

Aufgabe 4 (2P)

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5
2 1 3 0 4

Aufgabe 4 (2P)

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5 6
2 1 3 0 4

Aufgabe 4 (2P)

(a) ~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5 ~~6~~
2 1 3 0 4 6

Aufgabe 4 (2P)

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

Aufgabe 4 (2P)

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

Aufgabe 4 (2P)

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

Aufgabe 4 (2P)

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

Aufgabe 4 (2P)

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

Aufgabe 4 (2P)

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

2 1 3 0 4 6 7 9

False

(b)

(b) 0

(b) 0 1

(b) 0 1 2

(b) 0 1 ~~2~~

2

(b) 0 1 ~~2~~ 3

2

(b) 0 1 ~~2~~ ~~3~~

2 3

(b) 0 ~~1~~ ~~2~~ ~~3~~

2 3 1

(b) 0 ~~1~~ ~~2~~ ~~3~~ 4

2 3 1

(b) 0 ~~1~~ ~~2~~ ~~3~~ ~~4~~

2 3 1 4

(b) 0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5

2 3 1 4

(b) 0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5 6

2 3 1 4

(b) 0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ 5 ~~6~~

2 3 1 4 6

(b) 0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~

2 3 1 4 6 5

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

2 3 1 4 6 5

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

2 3 1 4 6 5

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

2 3 1 4 6 5 8

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

2 3 1 4 6 5 8 7

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

2 3 1 4 6 5 8 7 0

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

2 3 1 4 6 5 8 7 0

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

2 3 1 4 6 5 8 7 0 9

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

2 3 1 4 6 5 8 7 0 9 True

Aufgabe 5 (2P)

Stelle den Ausdruck korrekt in der Infix-Form dar.

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

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

Aufgabe 5 (2P)

$$(a) \quad M D - C F + * \Rightarrow (M - D) * (C + F)$$

Aufgabe 5 (2P)

$$(a) \quad M D - C F + * \quad \Rightarrow \quad (M - D) * (C + F)$$

$$(b) \quad - * P S + X Y \quad \Rightarrow \quad P * S - (X + Y)$$

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 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P														
---	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P														
---	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--

Output:

P														
---	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--

Output:

P	C													
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(+												
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--

Output:

P	C													
---	---	--	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(+												
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--

Output:

P	C	F												
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(+)											
---	---	---	---	--	--	--	--	--	--	--	--	--	--	--

Output:

P	C	F												
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(+												
---	---	--------------	--	--	--	--	--	--	--	--	--	--	--	--

Output:

P	C	F												
---	---	---	--	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(+)											
---	---	--------------	--------------	--	--	--	--	--	--	--	--	--	--	--

Output:

P	C	F	+											
---	---	---	---	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

-	(+)										
---	--------------	--------------	---	--	--	--	--	--	--	--	--	--	--

Output:

P	C	F	+										
---	---	---	---	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P	C	F	+											
---	---	---	---	--	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P	C	F	+	A										
---	---	---	---	---	--	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P	C	F	+	A	*									
---	---	---	---	---	---	--	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P	C	F	+	A	*	-								
---	---	---	---	---	---	---	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

P	C	F	+	A	*	-								
---	---	---	---	---	---	---	--	--	--	--	--	--	--	--

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

Aufgabe 6 (4P)

Input:

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

Stack: (nimmt → zu)

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

Output:

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

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 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3										
---	--	--	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5									
---	---	--	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2								
---	---	---	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2								
---	---	--------------	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2								
---	--------------	--------------	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7							
---	--------------	--------------	---	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6						
---	--------------	--------------	---	---	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4					
---	--------------	--------------	---	---	---	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4					
---	--------------	--------------	---	---	--------------	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4					
---	--------------	--------------	---	--------------	--------------	--	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10				
---	--------------	--------------	---	--------------	--------------	----	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10				
---	--------------	--------------	---	--------------	--------------	----	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10				
---	--------------	--------------	--------------	--------------	--------------	---------------	--	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70			
---	--------------	--------------	--------------	--------------	--------------	---------------	----	--	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2		
---	--------------	--------------	--------------	--------------	--------------	---------------	----	---	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2		
---	--------------	--------------	--------------	--------------	--------------	----	----	--------------	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2		
---	--------------	--------------	--------------	--------------	--------------	---------------	---------------	--------------	--	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	
---	--------------	--------------	--------------	--------------	--------------	---------------	---------------	--------------	----	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	
---	--------------	--------------	--------------	--------------	--------------	---------------	---------------	--------------	---------------	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	
--------------	--------------	--------------	--------------	--------------	--------------	---------------	---------------	--------------	---------------	--

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	71
--------------	--------------	--------------	---	--------------	--------------	----	----	--------------	---------------	----

Output:

Aufgabe 7 (2P)

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

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	71
--------------	--------------	--------------	---	--------------	--------------	----	----	--------------	---------------	----

Output: 71

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

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

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3										
---	--	--	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5									
---	---	--	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2								
---	---	---	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2								
---	---	--------------	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2								
---	--------------	--------------	--	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7							
---	--------------	--------------	---	--	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6						
---	--------------	--------------	---	---	--	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4					
---	--------------	--------------	---	---	---	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4					
---	--------------	--------------	---	---	--------------	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4					
---	--------------	--------------	---	--------------	--------------	--	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10				
---	--------------	--------------	---	--------------	--------------	----	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10				
---	--------------	--------------	---	--------------	--------------	---------------	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10				
---	--------------	--------------	--------------	--------------	--------------	---------------	--	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70			
---	--------------	--------------	--------------	--------------	--------------	---------------	----	--	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2		
---	--------------	--------------	--------------	--------------	--------------	---------------	----	---	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2		
---	--------------	--------------	--------------	--------------	--------------	----	----	--------------	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2		
---	--------------	--------------	--------------	--------------	--------------	----	----	--------------	--	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	
---	--------------	--------------	--------------	--------------	--------------	----	----	--------------	----	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	
---	--------------	--------------	--------------	--------------	--------------	----	----	--------------	---------------	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	
--------------	--------------	--------------	---	---	---	----	----	---	----	--

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	71
--------------	--------------	--------------	---	---	---	----	----	---	----	----

Output:

Aufgabe 7 (2P)

Input:

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

Stack: (nimmt → zu)

3	5	2	7	6	4	10	70	2	68	71
--------------	--------------	--------------	---	---	---	----	----	---	----	----

Output: 71

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 8 (2P)

(a) Ein Algorithmus ist ein Lösungsverfahren, das

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

ist.

(b) ▶ Handelt es sich um ein Lösungsverfahren? Ja

▶ Ist es endlich? Ja

▶ Ist es deterministisch? Nein, da es keine exakten Vorgabe gibt, wie die Artikel zu suchen sind.

▶ Ist es effektiv? Nein, da Artikel möglicherweise im Angebot des Supermarktes fehlen und nicht vorgegeben ist, was in diesem Fall zu tun ist.

Aufgabe 9 (2P)

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

Aufgabe 9 (2P)

$$\begin{aligned} \text{ggT}(30, 12) &= (18, 12) = (6, 12) \stackrel{\text{S}}{=} (12, 6) = (6, 6) \\ &= (0, 6) \stackrel{\text{S}}{=} (6, 0) = (6, 0) = 6 \end{aligned}$$

Aufgabe 10 (2P)

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

Aufgabe 10 (2P)

$$\text{ggT}(32, 18) = (18, 14) = (14, 4) = (4, 2) = (2, 0) = 2$$

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 11 (2P)

```
def function(n):  
    s = 0 # 1*c  
    for i in range(0, n): # n*c  
        for j in range(0, n): # n*n*c  
            s += i + j # 3*n*n*c  
        for k in range(0, 2*n): # 2*n*n*c  
            s += k # 4*n*n*c  
    return s # 1*c
```

Summe: $T(n) = 10cn^2 + cn + 2c \in O(n^2)$

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?

Aufgabe 12 (1P)

$$T(n) \in O(n^2) \Rightarrow T(n) = C \cdot n^2$$

$$T(100) = C \cdot 100^2 = 3 \text{ s}$$

$$T(300) = C \cdot 300^2 = 9 \cdot C \cdot 100^2 = 9 \cdot 3 \text{ s} = 27 \text{ s}$$