

Datenstrukturen (Stack)

Prüfungsvorbereitung

Aufgabe 1

Nenne drei Anwendungen für Stacks in der Informatik.

Aufgabe 1

Hier ein paar Anwendungen für Stacks:

- ▶ History-Funktion (zurück/vor) in einem Browser
- ▶ Undo-Funktion in einem Anwendungsprogramm
- ▶ Überprüfung von Klammerausdrücken
- ▶ Umwandlung von Infix- in Postfix-Ausdrücke
- ▶ Berechnung von Postfix-Ausdrücken

Aufgabe 2

Wofür steht die Kurzformel LIFO bei Stacks?

Aufgabe 2

Last In - First Out

Aufgabe 3

Welche Ausgaben macht das folgende Python-Modul?

```
1 from stack import Stack
2
3 s = Stack()
4 s.push(5)
5 s.push(2)
6 s.push(4)
7 a = s.pop()
8 s.push(7)
9 s.push(1)
10 b = s.peek()
11 s.push(8)
12
13 print(a)
14 print(b)
15 print(s.size())
```

Aufgabe 3

```
1  from stack import Stack
2
3  s = Stack()      # []
4  s.push(5)       # [5]
5  s.push(2)       # [5, 2]
6  s.push(4)       # [5, 2, 4]
7  a = s.pop()     # [5, 2] a=4
8  s.push(7)       # [5, 2, 7]
9  s.push(1)       # [5, 2, 7, 1]
10 b = s.peek()   # [5, 2, 7, 1] b=1
11 s.push(8)       # [5, 2, 7, 1, 8]
12
13 print(a)        # => 4
14 print(b)        # => 1
15 print(s.size()) # => 5
```

Aufgabe 4

Es sei s eine Instanz der Klasse `Stack`. Was ist

- (a) die Gemeinsamkeit
- (b) der Unterschied

zwischen den Ausdrücken `s.peek()` und `s.pop()`?

Aufgabe 4

- (a) Gemeinsamkeit: beide Methoden geben das oberste Element des Stacks zurück.
- (b) Unterschied: `s.pop()` entfernt das oberste Element, `s.peek()` entfernt es nicht.
to peek: forschend blicken, spähen, gucken

Aufgabe 5

Implementiere die fehlenden Methoden der Klasse Stack für den gegebenen Konstruktor.

```
1 class Stack:
2
3     def __init__(self):
4         self.items = []
5
6     def push(self, item):
7         ...
8
9     def pop(self):
10        ...
11
12    def peek(self):
13        ...
14
15    def is_empty(self):
16        ...
17
```

Aufgabe 5

```
1 class Stack:
2
3     def __init__(self):
4         self.items = []
5
6     def push(self, item):
7         self.items.append(item)
8
9     def pop(self):
10        return self.items.pop()
11
12    def peek(self):
13        return self.items[-1]
14
15    def is_empty(self):
16        return self.items == []
17
18    def size(self):
19        return len(self.items)
```

Aufgabe 6

Angenommen, ein Programm führt eine Folge von push- und pop-Operationen in einer bestimmten Reihenfolge auf einem Stack aus. Die push-Operationen legen dabei die ganzen Zahlen von 0 bis 9 in dieser Reihenfolge auf den Stack ab. Die pop-Operationen geben den Rückgabewert aus. Welche der folgenden Output-Sequenzen kann es dabei nicht geben?

- (a) 1, 5, 4, 6, 3, 2, 9, 8, 0, 7
- (b) 1, 3, 2, 0, 8, 7, 6, 5, 4, 9
- (c) 1, 5, 6, 4, 7, 9, 8, 3, 0, 2
- (d) 0, 2, 3, 7, 9, 8, 6, 5, 1, 4

Aufgabe 6

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

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

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

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

Aufgabe 6

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

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

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

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

Aufgabe 6

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

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

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

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

Aufgabe 6

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

0 ~~1~~

1

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

0 ~~1~~

1

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

0 ~~1~~

1

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

~~0~~ 1

0

Aufgabe 6

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

0 ~~1~~ 2

1

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

0 ~~1~~ 2

1

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

0 ~~1~~ 2

1

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

~~0~~ 1 2

0

Aufgabe 6

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

0 ~~1~~ 2 3

1

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

0 ~~1~~ 2 3

1

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

0 ~~1~~ 2 3

1

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

~~0~~ 1 ~~2~~

0 2

Aufgabe 6

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

1

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

1 3

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

1

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

0 2

Aufgabe 6

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

1

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

1 3 2

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

1

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

0 2 3

Aufgabe 6

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

1 5

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

1 3 2 0

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

1 5

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

0 2 3

Aufgabe 6

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~

1 5 4

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ 4

1 3 2 0

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

0 ~~1~~ 2 3 4 ~~5~~ 6

1 5

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

~~0~~ 1 ~~2~~ ~~3~~ 4 5

0 2 3

Aufgabe 6

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~ 6

1 5 4

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ 4 5

1 3 2 0

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

0 ~~1~~ 2 3 4 ~~5~~ ~~6~~

1 5 6

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

~~0~~ 1 ~~2~~ ~~3~~ 4 5 6

0 2 3

Aufgabe 6

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~ ~~6~~

1 5 4 6

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ 4 5 6

1 3 2 0

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~ ~~6~~

1 5 6 4

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

~~0~~ 1 ~~2~~ ~~3~~ 4 5 6 7

0 2 3

Aufgabe 6

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

0 ~~1~~ 2 ~~3~~ ~~4~~ ~~5~~ ~~6~~

1 5 4 6 3

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ 4 5 6 7

1 3 2 0

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~ ~~6~~ 7

1 5 6 4

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

~~0~~ 1 ~~2~~ ~~3~~ 4 5 6 ~~7~~

0 2 3 7

Aufgabe 6

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

0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~

1 5 4 6 3 2

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ 4 5 6 7 8

1 3 2 0

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~ ~~6~~ ~~7~~

1 5 6 4 7

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

~~0~~ 1 ~~2~~ ~~3~~ 4 5 6 ~~7~~ 8

0 2 3 7

Aufgabe 6

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

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

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

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

Aufgabe 6

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

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

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

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

Aufgabe 6

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

0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ 7 8 9

1 5 4 6 3 2

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ 4 5 ~~6~~ ~~7~~ ~~8~~

1 3 2 0 8 7 6

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

0 ~~1~~ 2 3 ~~4~~ ~~5~~ ~~6~~ ~~7~~ 8 ~~9~~

1 5 6 4 7 9

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

~~0~~ 1 ~~2~~ ~~3~~ 4 5 6 ~~7~~ ~~8~~ ~~9~~

0 2 3 7 9 8

Aufgabe 6

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

1 5 4 6 3 2 9

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

1 3 2 0 8 7 6 5

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

1 5 6 4 7 9 8

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

0 2 3 7 9 8 6

Aufgabe 6

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

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

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

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

Aufgabe 6

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

1 5 4 6 3 2 9 8

False

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

1 3 2 0 8 7 6 5 4

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

1 5 6 4 7 9 8 3

False

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

0 2 3 7 9 8 6 5

False

Aufgabe 6

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

0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ 7 ~~8~~ ~~9~~

1 5 4 6 3 2 9 8

False

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

~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

1 3 2 0 8 7 6 5 4 9

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

0 ~~1~~ 2 ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

1 5 6 4 7 9 8 3

False

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

~~0~~ 1 ~~2~~ ~~3~~ 4 ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

0 2 3 7 9 8 6 5

False

Aufgabe 6

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

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

(c) 1, 5, 6, 4, 7, 9, 8, 3, 0, 2
0 ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~
1 5 6 4 7 9 8 3 False

(d) 0, 2, 3, 7, 9, 8, 6, 5, 1, 4
~~0~~ ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~
0 2 3 7 9 8 6 5 False

Aufgabe 7

Erkläre, wie mit Hilfe eines Stacks geprüft werden kann, ob ein Term aus lauter Klammern syntaktisch korrekt ist.

Aufgabe 7

Erzeuge einen leeren Stack.

Durchlaufe den Term zeichenweise.

Ist das Zeichen eine öffnende Klammer, lege sie dem Stack ab.

Ist das Zeichen eine schliessende Klammer, prüfe, ob der Stack leer ist. Wenn ja, ist die Klammerung falsch → Stop. Wenn nein entferne eine öffnende Klammer vom Stack.

Prüfe am Schluss, ob der Stack leer ist. Wenn ja, dann ist die Klammerung richtig. Wenn nein, dann ist sie falsch.