

Der Call-Stack und die Rekursion

Ein populäres Beispiel für rekursive Algorithmen ist die Fakultätsfunktion:

```

5! = 5*4*3*2*1
fakultaet(5) = 120
fakultaet(3) = 3*2*1 = 6

```



(A1) Iterativ

Implementiere in BlueJ eine iterative Version der Fakultätsfunktion, die als Argument die Zahl entgegennimmt, deren Fakultät berechnet werden soll.



(A2) Rekursiv

Implementiere anhand des folgenden Pseudocodes eine rekursive Version fak_rekursiv.

```

fak_rekursiv(int n):
  wenn n=1:
    return 1
  sonst:
    return n*fak_rekursiv(n-1)

```

- Was ist der Rekursionsfall, was der Basisfall?
- Teste deine rekursive Methode

Detaillierte Betrachtung des Call-Stacks bei der Rekursion

Was passiert	Wie sieht der Stack aus?
fak_rekursiv(3) wird aufgerufen. Auf dem Stack wird Speicher für diesen Aufruf reserviert. Es gibt keine Rücksprungadresse. Innerhalb dieses Aufrufs wird fak_rekursiv(2) (nächster Schritt) aufgerufen, da die Fallunterscheidung nicht zum Basisfall führt sondern zum Rekursionsfall.	<pre> graph TD subgraph Stack direction TB Frame[fak_rekursiv(3)] Frame --- Param[n=3] Frame --- Ret[ruecksprungadresse=keine] end </pre>

Was passiert	Wie sieht der Stack aus?				
fak_rekursiv(3) wird aufgerufen. Auf dem Stack wird Speicher für diesen Aufruf reserviert. Es gibt keine Rücksprungadresse	<table border="1"><tr><td colspan="2" data-bbox="788 248 1471 327">fak_rekursiv(3)</td></tr><tr><td data-bbox="788 327 1031 394">n=3</td><td data-bbox="1031 327 1471 394">ruecksprungadresse=keine</td></tr></table>	fak_rekursiv(3)		n=3	ruecksprungadresse=keine
fak_rekursiv(3)					
n=3	ruecksprungadresse=keine				

From: <https://www.info-bw.de/> -

Permanent link: https://www.info-bw.de/faecher:informatik:oberstufe:algorithmen:rekursion:callstack_rekursion:start?rev=1642073319

Last update: 13.01.2022 11:28

