

9. Codeanalyse Haskell [ID: 64929]

Ihre Antwort:

Betrachten Sie den folgenden Haskell-Code:

```
1  take 4 $ [(x,y) | x <- [1..], y <- [1..]]
2
3  take 4 $ [x+1 | x <- [1..], odd x]
4
5  take 4 $ [(n, even n) | n <- [1..]]
6
7  head (tail [1..4])
8
9  (head . tail) [1..4]
10
11 head . tail $ [1..4]
12
13 head . tail [1..4]
14
15 zip [1 .. 2] ["one", "two", "three"]
16
17 [(x, y) | x <- [1 .. 2], y <- ["one", "two", "three"]]
```

Geben Sie für alle Zeilen die Ergebnisse der Auswertung der jeweiligen Ausdrücke an.

- Zeile 1 (1P): [(1,1),(1,2),(1,3),(1,4)] ✓
- Zeile 3 (1P): [2,4,6,8] ✓
- Zeile 5 (1P): [(1,false), (2, true), (3, false), (4, true)] ✗
- Zeile 7 (0.5P): 2 ✓
- Zeile 9 (0.5P): 2 ✓
- Zeile 11 (0.5P): 2 ✓
- Zeile 13 (1P): error ✗
- Zeile 15 (1P): [(1,"one"),(2,"two")] ✓
- Zeile 17 (1.5P): [(1,"one"),(1,"two"),(1,"three"),(2,"one"),(2,"two"),(2,"three")] ✓

Hinweis: Wenn eine Zeile nicht ausgewertet werden kann, etwa durch Syntax- oder Typ-Fehler, geben Sie bitte als Antwort "**Fehler**" ein.