

Please consider this programming problem:  
Given a positive integer N. Calculate the Fibonacci series till the number N. If N is a part of the series, include N as well.  
Provide an example solution for the problem in the Python language.

Vorschläge anzeigen ▾ 🔊

Python

```
def fibonacci_series(n):  
    """Calculates the Fibonacci series up to the given number.  
  
    Args:  
        n: The maximum number in the series.  
  
    Returns:  
        A list containing the Fibonacci series up to n.  
        """  
  
    if n <= 0:  
        return []  
  
    fib_series = [0, 1]  
    while fib_series[-1] < n:  
        next_fib = fib_series[-1] + fib_series[-2]  
        fib_series.append(next_fib)  
  
    # If n is in the series, keep it  
    if fib_series[-1] == n:  
        return fib_series  
    else:  
        return fib_series[:-1]  
  
# Example usage:  
n = 20  
result = fibonacci_series(n)  
print(result)
```

Verwende den Code [mit Vorsicht](#).

This code defines a function `fibonacci_series` that takes a positive integer `n` as input and returns a list containing the Fibonacci series up to `n`.

Here's how the function works:

1. It checks if `n` is less than or equal to 0. If so, it returns an empty list since the Fibonacci series starts from 0 and 1.
2. It initializes a list `fib_series` with the first two Fibonacci numbers, 0 and 1.
3. It uses a `while` loop to calculate the next Fibonacci number until the last number in the series is greater than or equal to `n`.
4. If the last number in the series is equal to `n`, it returns the entire series. Otherwise, it returns the series without the last element.

In the example usage, we call the `fibonacci_series` function with `n = 20` and print the result. This will output the Fibonacci series up to 20, including 20 since it's part of the series.

Prompt hier eingeben

