

SHERLOCK Debug Report

Error: RecursionError

Root Cause

The `factorial` function is defined recursively but lacks a crucial base case. A recursive function must have a condition under which it stops calling itself to prevent infinite recursion. In this code, `factorial(n)` always calls `factorial(n-1)`, which in turn calls `factorial(n-2)`, and so on. This process continues indefinitely, with `n` eventually becoming negative, never reaching a point where the recursion terminates. Python's recursion limit is exceeded, leading to a `RecursionError`.

Fix

To fix this, a base case needs to be added to the `factorial` function. The factorial of 0 (0!) is 1, and the factorial of 1 (1!) is also 1. Therefore, if `n` is 0 or 1, the function should return 1 directly without making any further recursive calls. This provides the necessary termination condition for the recursion.

Corrected Code

```
def factorial(n): if n == 0 or n == 1: return 1 return n * factorial(n-1)
```

Confidence: 100%