

In [2]: `import ctypes`

```
# Load the shared library
libPMU = ctypes.CDLL('./libMyLib.so')

# Initialize the PMU counters (arguments: do_reset, enable_divider)
libPMU.init_pmu_counters(ctypes.c_int(1), ctypes.c_int(0))

# Get the cycle count
cycle_count = libPMU.get_cycle_count()
print("Cycle count:", cycle_count)
```

Cycle count: 1143062

In []:

```
1 import ctypes
2 import time
3 import numpy as np
4 import random
5
6 # Load the shared library
7 libPMU = ctypes.CDLL('./libPMU.so')
8
9 # Initialize PMU counters
10 def init_pmu_counters():
11     libPMU.init_pmu_counters(1, 0)
12
13 # Get cycle count
14 def get_cycle_count():
15     return libPMU.get_cycle_count()
16
17 # Recursive Fibonacci function
18 def recur_fibo(n):
19     return n if n <= 1 else recur_fibo(n-1) + recur_fibo(n-2)
20
21 # Run timing test
22 def run_fibo_timing_test(terms):
23     results = {'cycles': [], 'times': [], 'cycle_errors': [], 'time_errors': []}
24
25     for n in terms:
26         cycles, times = [], []
27         for _ in range(3): # Three trials
28             init_pmu_counters()
29
30             # Get 'before' time and cycle count
31             start_time = time.time()
32             start_cycle = get_cycle_count()
33
34             # Run the recur_fibo function
35             recur_fibo(n)
36
37             # Get 'after' cycle count and time
38             end_cycle = get_cycle_count()
39             end_time = time.time()
40
41             # Calculate the differences
```