

DAA - Assignment no. 01

Write a program non-recursive and recursive program to calculate Fibonacci numbers and analyze their time and space complexity.

Input:

```
Assignment1.py X
Assignment1.py > ...
1  def non_recursive_fibonacci(n):
2
3      if n < 0:
4          raise ValueError("n must be a non-negative integer.")
5      elif n == 0 or n == 1:
6          return n
7      else:
8          a = 0
9          b = 1
10         for i in range(2, n + 1):
11             c = a + b
12             a = b
13             b = c
14         return c
15
16
17  def recursive_fibonacci(n):
18
19      if n < 0:
20          raise ValueError("n must be a non-negative integer.")
21      elif n == 0 or n == 1:
22          return n
23      else:
24          return recursive_fibonacci(n - 1) + recursive_fibonacci(n - 2)
25
26
27  def analyze_time_complexity(n):
28
29      non_recursive_time_complexity = "O(n)"
30
31      recursive_time_complexity = "O(2^n)"
32
33      return (f"Non-recursive Fibonacci function time complexity: {non_recursive_time_complexity}\n"
34              | f"Recursive Fibonacci function time complexity: {recursive_time_complexity}")
35
36
```

Assignment1.py ●

Assignment1.py > ...

```
33     return (f"Non-recursive Fibonacci function time complexity: {non_recursive_time_complexity}\n"
34             | f"Recursive Fibonacci function time complexity: {recursive_time_complexity}")
35
36
37 def analyze_space_complexity(n):
38
39     non_recursive_space_complexity = "O(1)"
40
41     recursive_space_complexity = "O(n)"
42
43     return (f"Non-recursive Fibonacci function space complexity: {non_recursive_space_complexity}\n"
44             | f"Recursive Fibonacci function space complexity: {recursive_space_complexity}")
45
46
47 def main():
48
49
50     print("Non-recursive Fibonacci numbers:")
51     for i in range(10):
52         print(non_recursive_fibonacci(i))
53
54     print("\nRecursive Fibonacci numbers:")
55     for i in range(10):
56         print(recursive_fibonacci(i))
57
58     print("\nTime complexity analysis:")
59     print(analyze_time_complexity(10))
60
61     print("\nSpace complexity analysis:")
62     print(analyze_space_complexity(10))
63
64
65 if __name__ == "__main__":
66     main()
67
```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS D:\Tanmay Mohadikar\Sem 7 Practicals\DAA> & D:/Python/python.exe "d:/Tanmay Mohadikar/Sem 7 Practicals/DAA/Assignment1.py"
Non-recursive Fibonacci numbers:
0
1
1
2
3
5
8
13
21
34

Recursive Fibonacci numbers:
0
1
1
2
3
5
8
13
21
34

Time complexity analysis:
Non-recursive Fibonacci function time complexity: O(n)
Recursive Fibonacci function time complexity: O(2^n)

Space complexity analysis:
Non-recursive Fibonacci function space complexity: O(1)
Recursive Fibonacci function space complexity: O(n)
PS D:\Tanmay Mohadikar\Sem 7 Practicals\DAA>
```