```python
from numpy import zeros
from time import time


# fibonacci function with 3 variables
def fibonacci_3_variables(num):
    if (num < 3):
        return 1
    a, b, c = 1, 1, 0
    for i in range(num-2):
        c = a + b
        a,b = b,c
    return c


# fibonacci function with 2 variables
def fibonacci_2_variables(num):
    if (num < 3):
        return 1
    a, b = 1, 1
    for i in range(num-2):
        b += a
        a = b - a
    return b


# recursive fibonacci function
def recursive_fibonacci(num):
    if (num <= 2):
        return 1
    return recursive_fibonacci(num-1) + recursive_fibonacci(num-2)


# fibonacci function with array
def fibonacci_with_array(num):
    if (num <= 2):
        return 1
    array = zeros(num)
    array[0], array[1] = 1, 1
    for i in range(num-2):
        array[i+2] = array[i+1] + array[i]
    return int(array[i+2])


fibonacci_3_variables_time, fibonacci_2_variables_time, recursive_fibonacci_time,
fibonacci_with_array_time = 0, 0, 0, 0
```

```
input_data = open('D:/Education/Data
Structure/Projects/Fibonacci/InputData.txt','r')
output_data = open('D:/Education/Data
Structure/Projects/Fibonacci/OutputData.txt','w')
for i in range(20):
    num = int(input_data.readline())

    start = time()
    fibonacci_3_variables(num)
    end = time()
    fibonacci_3_variables_time += end - start
    output_data.write('fibonacci function with 3 variables: f( %d ) = %d , time =
%.20f '  % (num, fibonacci_3_variables(num), end - start) + '\n' )
    print('fibonacci function with 3 variables: f( %d ) = %d , time = %.20f '  %
(num, fibonacci_3_variables(num), end - start) + '\n' )

    start = time()
    fibonacci_2_variables(num)
    end = time()
    fibonacci_2_variables_time += end - start
    output_data.write('fibonacci function with 2 variables: f( %d ) = %d , time =
%.20f  '  % (num, fibonacci_2_variables(num), end - start) + '\n' )
    print('fibonacci function with 2 variables: f( %d ) = %d , time = %.20f  '  %
(num, fibonacci_2_variables(num), end - start) + '\n' )

    start = time()
    recursive_fibonacci(num)
    end = time()
    recursive_fibonacci_time += end - start
    output_data.write('recursive fibonacci function : f( %d ) = %d , time =
%.20f  '  % (num, recursive_fibonacci(num), end - start) + '\n' )
    print('recursive fibonacci function : f( %d ) = %d , time = %.20f  '  % (num,
recursive_fibonacci(num), end - start) + '\n' )

    start = time()
    fibonacci_with_array(num)
    end = time()
    fibonacci_with_array_time += end - start
    output_data.write('fibonacci function with array : f( %d ) = %d , time =
%.20f  '  % (num, fibonacci_with_array(num), end - start) + '\n'*2 )
    print('fibonacci function with array : f( %d ) = %d , time = %.20f  '  %
(num, fibonacci_with_array(num), end - start) + '\n'*2 )

output_data.write( "fibonacci function with 3 variables total time : %.20f
\nfibonacci function with 2 variables total time : %.20f \nrecursive fibonacci
function total time : %.20f\nfibonacci function with array total time : %.20f " %
(fibonacci_3_variables_time , fibonacci_2_variables_time,
recursive_fibonacci_time, fibonacci_with_array_time))
print("fibonacci function with 3 variables total time : %.20f \nfibonacci
function with 2 variables total time : %.20f \nrecursive fibonacci function total
time : %.20f\nfibonacci function with array total time : %.20f \n" %
(fibonacci_3_variables_time , fibonacci_2_variables_time,
recursive_fibonacci_time, fibonacci_with_array_time))
input_data.close()
output_data.close()
```