

Rajan Gautam

19BCP101

Div. II, CE 19
SOT, PDEU

Pandit Deendayal Energy University
School of Technology

Design & Analysis of Algorithm (20CP209P)

B. Tech - Computer Science & Engineering (Sem-IV)

Table of Contents

| | |
|--|----------|
| Lab 4 Assignment: Implementation of Recursive/Non-Recursive Function and its Analysis..... | 2 |
| AIM 1: To write a C/C++ Program to implement Recursive/Non-Recursive Function and its Analysis..... | 2 |
| CODE:..... | 2 |
| OUTPUT: | 5 |
| Comparison for Factorial Function | 5 |
| Comparison for Fibonacci Function | 6 |

Lab 4 Assignment: Implementation of Recursive/Non-Recursive Function and its Analysis

AIM 1: To write a C/C++ Program to implement Recursive/Non-Recursive Function and its Analysis

Aim: To write a recursive and non- recursive function to compute (Any Two)

- nth Fibonacci
- Factorial
- Binary Search Algorithm.

Compare the time requirements of non-recursive function with those of recursive version.

CODE:

```
1. #include<bits/stdc++.h>
2. #include <iostream>
3.
4. using namespace std;
5.
6. int recursive_fibonacci(int n)
7. {
8.     if (n <= 1)
9.     {
10.         return n;
11.     }
12.     else
13.     {
14.         return recursive_fibonacci(n-1) + recursive_fibonacci(n-2);
15.     }
16. }
17.
18. int non_recursive_fibonacci(int n)
19. {
20.     int n1 = 0, n2 = 1, next;
21.     for(int i = 0; i < n - 1; i++)
22.     {
23.         next = n1 + n2;
24.         n1 = n2;
25.         n2 = next;
26.     }
```

```
27.     return next;
28. }
29.
30. int recursive_factorial(int n)
31. {
32.     if(n == 0)
33.     {
34.         return 1;
35.     }
36.     else
37.     {
38.         return n * recursive_factorial(n - 1);
39.     }
40. }
41.
42. int non_recursive_factorial(int n)
43. {
44.     int factorial = 1;
45.     for(int i = 1; i <= n; i++)
46.     {
47.         factorial = factorial * i;
48.     }
49.     return factorial;
50. }
51.
52. int main()
53. {
54.     int m, n;
55.     cout<<"\t Enter the value of Factorial and Fibonacci: ";
56.     cin>>m;
57.     cin>>n;
58.
59.     cout<<"\t Time Taken by the Recursive and Non-Recursive
        Factorial"<<endl;
60.     cout<<"\t Using Recursion: "<< m << "! = "<<
        recursive_factorial(m)<<endl;
61.     cout<<"\t Using Non-Recursion: "<< m << "! = "<<
        non_recursive_factorial(m)<<endl;
62.     cout<<"\tNumber \t Time Taken by Recursive \tTime Taken by Non-
        Recursive"<<endl;
63.     while(m > 5)
```

```

64.     {
65.         clock_t ft1 = clock();
66.         recursive_factorial(n);
67.         clock_t ft2 = clock() - ft1;
68.
69.         clock_t nft1 = clock();
70.         non_recursive_factorial(n);
71.         clock_t nft2 = clock() - nft1;
72.
73.         cout<<"\t"<<m<<" \t "<< fixed << setprecision(3)<< (float)ft2 /
        CLOCKS_PER_SEC * 1000 << " ms "<<"\t\t\t"<<(float)nft2 / CLOCKS_PER_SEC *
        1000<< " ms\n";
74.
75.         m = m - 1;
76.     }
77.     cout<<"\n\t Time Taken by the Recursive and Non-Recursive Fibonacci
        Series"<<endl;
78.     cout<<"\t nth Term Using Recursion: " <<recursive_fibonacci(n)<<endl;
79.     cout<<"\t nth Term Using Non-Recursion: " <<
        non_recursive_fibonacci(n)<<endl;
80.     cout<<"\tNumber \t Time Taken by Recursive \tTime Taken by Non-
        Recursive"<<endl;
81.     while(n > 10)
82.     {
83.         clock_t ft1 = clock();
84.         recursive_fibonacci(n);
85.         clock_t ft2 = clock() - ft1;
86.
87.         clock_t nft1 = clock();
88.         non_recursive_fibonacci(n);
89.         clock_t nft2 = clock() - nft1;
90.
91.         cout<<"\t"<<n<<" \t "<< fixed << setprecision(3)<< (float)ft2 /
        CLOCKS_PER_SEC * 1000 << " ms "<<"\t\t\t"<<(float)nft2 / CLOCKS_PER_SEC *
        1000<< " ms\n";
92.
93.         n = n - 1;
94.     }
95.
96.     return 0;
97.

```

OUTPUT:

```

main.cpp
input
Enter the value of Factorial and Fibonacci: 10 15
Time Taken by the Recursive and Non-Recursive Factorial
Using Recursion: 10! = 3628800
Using Non-Recursion: 10! = 3628800

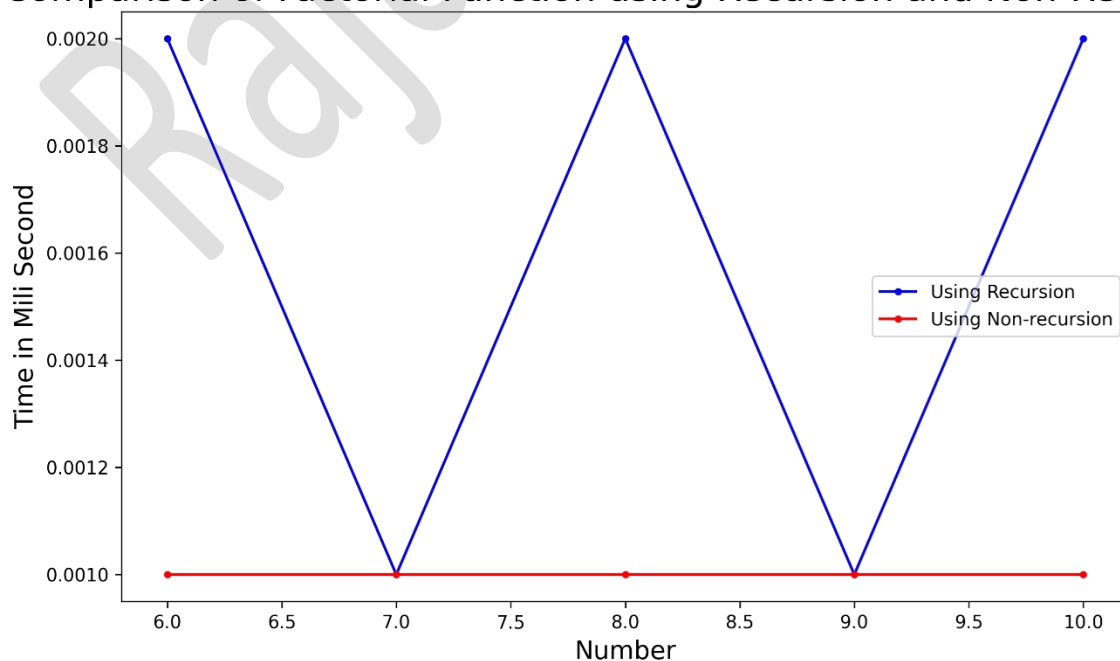
Number    Time Taken by Recursive    Time Taken by Non-Recursive
10        0.002 ms                   0.001 ms
9         0.001 ms                   0.001 ms
8         0.002 ms                   0.001 ms
7         0.001 ms                   0.001 ms
6         0.002 ms                   0.001 ms

Time Taken by the Recursive and Non-Recursive Fibonacci Series
nth Term Using Recursion: 610
nth Term Using Non-Recursion: 610

Number    Time Taken by Recursive    Time Taken by Non-Recursive
15        0.007 ms                   0.001 ms
14        0.004 ms                   0.001 ms
13        0.003 ms                   0.001 ms
12        0.003 ms                   0.001 ms
11        0.002 ms                   0.001 ms

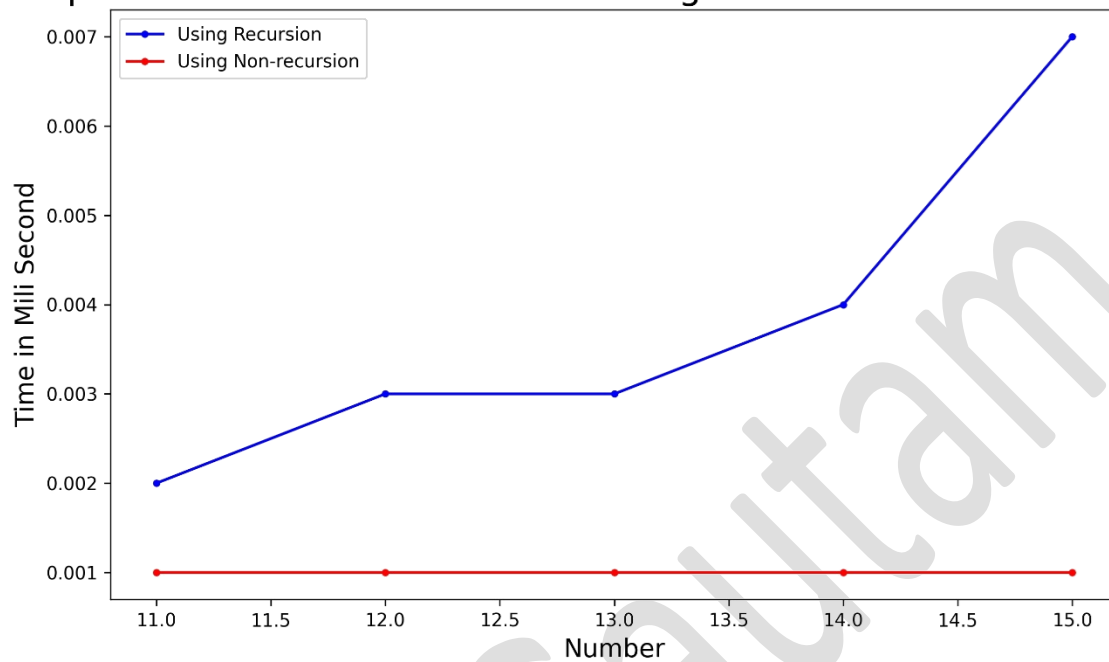
...Program finished with exit code 0
Press ENTER to exit console.

```

Comparison for Factorial Function**Comparison of Factorial Function using Recursion and Non-Recursion**

Comparison for Fibonacci Function

Comparison of Fibonacci Function using Recursion and Non-Recursion



Link: https://github.com/rgautam320/Design-and-Analysis-of-Algorithm-Lab/tree/master/Lab_4_Recursion