CENTRAL UNIVERSITY OF HARYANA

Department of Computer Science & Engineering under SOET

Analysis and Design of Algorithm

Assignment-01

Submitted by
Abhishek Rao
Roll No- 191882

Submitted to
Dr. Manish Shailani
Assistant Professor
Department of Computer Science and Engineering
Central University of Haryana (SOET)

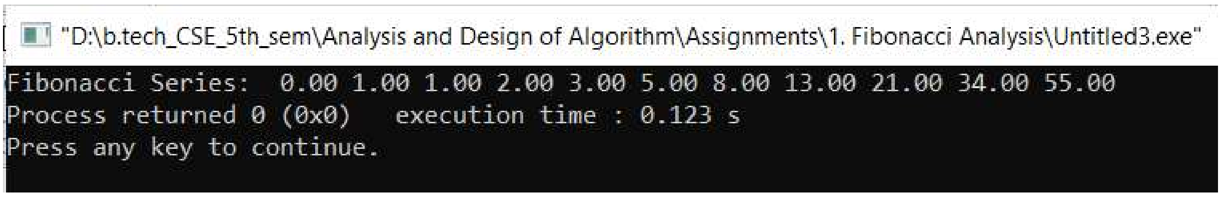
Program - 1: Write a program to implement a Fibonacci series and plot its run time for different number of terms.

Code:

Method-1: Recursive Method

```
#include<stdio.h>

// fibonacci function
double fib(double n)
{
    // if value of n is 0 or 1 (i.e. less than or equal to 1) then if will return 0 or 1
    if (n <=1 )
           return n;
   // if value of n is greater than 1 then it will return the summation of previous two terms
    else
          return fib(n-1) + fib(n-2);
}


// driver code
int main ()
{
    // taking the value of n
    double n=5;

    // printing the fibonacci series using for loop starting from value of n from 0 to n
    printf("Fibonacci Series: ");
    for(int i=0; i<=n; i++){
        printf(" %0.2lf", fib(i));
    }

return 0;
}
```

Output:



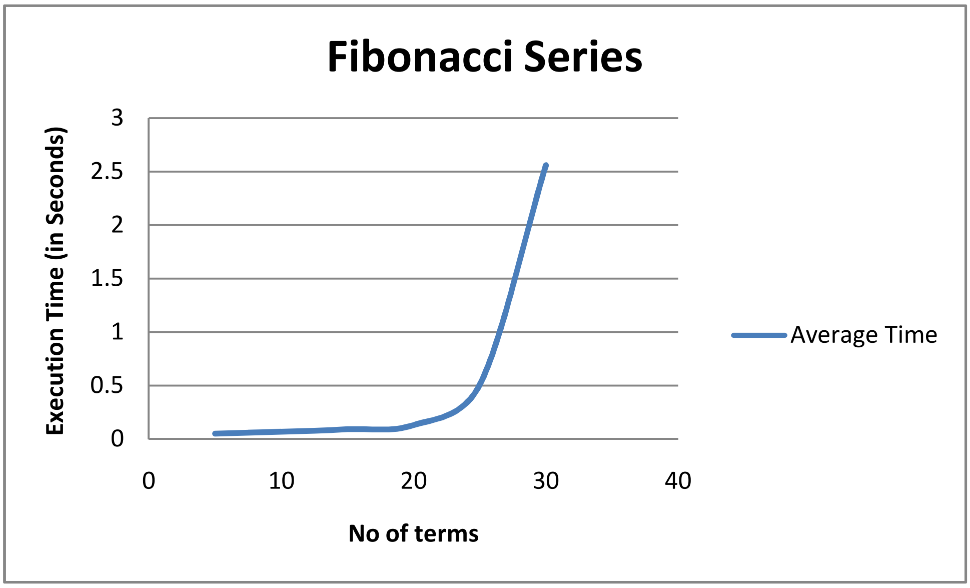
"D:\b.tech_CSE_5th_sem\Analysis and Design of Algorithm\Assignments\1. Fibonacci Analysis\Untitled3.exe"

```
Fibonacci Series:  0.00 1.00 1.00 2.00 3.00 5.00 8.00 13.00 21.00 34.00 55.00
Process returned 0 (0x0)   execution time : 0.123 s
Press any key to continue.
```

Time Analysis:

No of Terms	Time Taken during 10 different times										Average Time
5	0.051	0.066	0.036	0.048	0.038	0.055	0.037	0.032	0.05	0.081	0.0494
10	0.068	0.085	0.069	0.055	0.06	0.056	0.092	0.056	0.056	0.071	0.0668
15	0.127	0.064	0.067	0.073	0.094	0.124	0.06	0.073	0.132	0.09	0.0904
20	0.117	0.14	0.167	0.116	0.137	0.124	0.111	0.138	0.127	0.117	0.1294
25	0.511	0.504	0.508	0.47	0.482	0.461	0.628	0.536	0.536	0.424	0.506
30	2.477	2.512	2.595	2.455	2.352	2.578	2.527	2.642	2.527	2.926	2.5591

Graph Plot:



Github Link:

Code for method 1:

https://github.com/Abhishek-Rao-191882/ADA-5th-sem/blob/main/Assignment-01/Fibonacci-with-recursive.c

Code for method 2:

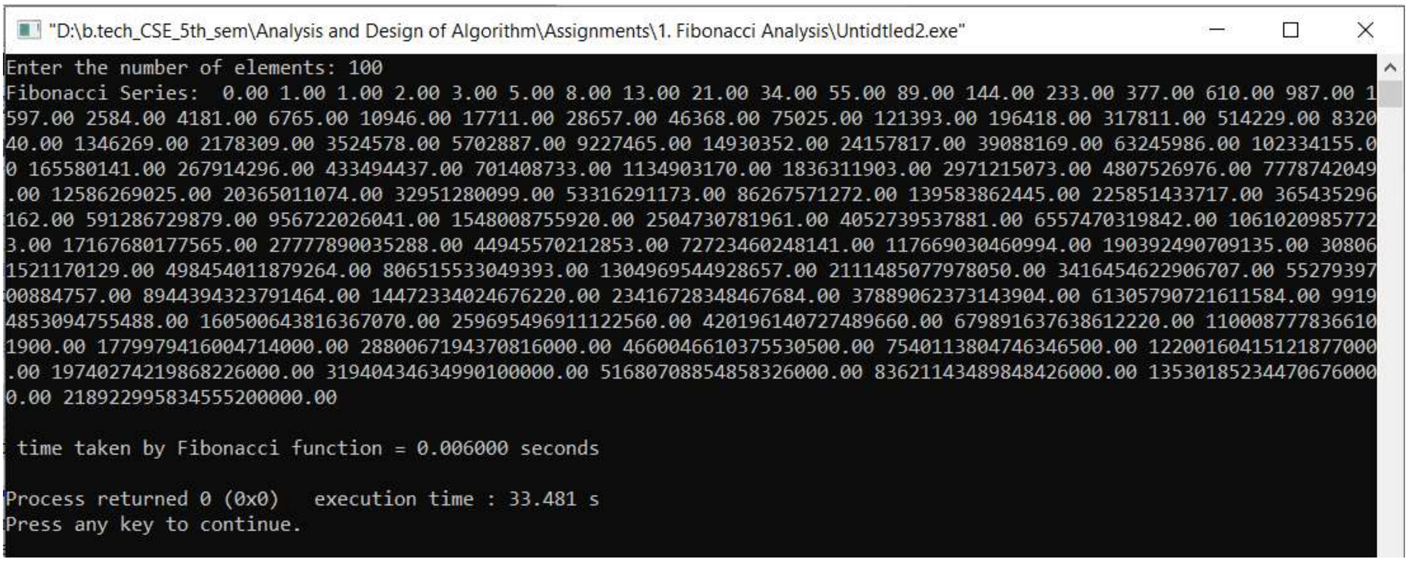
https://github.com/Abhishek-Rao-191882/ADA-5th-sem/blob/main/Assignment-01/Fibonacci-with-Recursive(DP_Concept).c

Method-2: Recursive with Dynamic Programming

Code :

```
#include<stdio.h>
#include<time.h>
// Fibonacci Function
void printFibonacci(int n){
    double i, first, second, next;
    // these are first two terms
    first=0;
    second=1;
    // for loop using to print the terms
    for(i=1; i<=n; i++)
    {
      printf(" %0.2lf",first);
      next=first+second;
      first=second;
      second=next;
    }
}
int main(){
    int n;
    // taking user input  for the number of terms
    printf("Enter the number of elements: ");
    scanf("%d",&n);
    // declaring variable for time calculating function
    clock_t t;
    t = clock();
    // printing the fibonacci series by calling the above defined function
    printf("Fibonacci Series: ");
    printFibonacci(n);
    // function for time calculation of the given function
    t = clock() - t;
    double time_taken = ((double)t)/CLOCKS_PER_SEC; // in seconds
    printf("/n");
    printf("/n time taken by Fibonacci function = %f seconds /n", time_taken);
  return 0;
 }
```

Output:

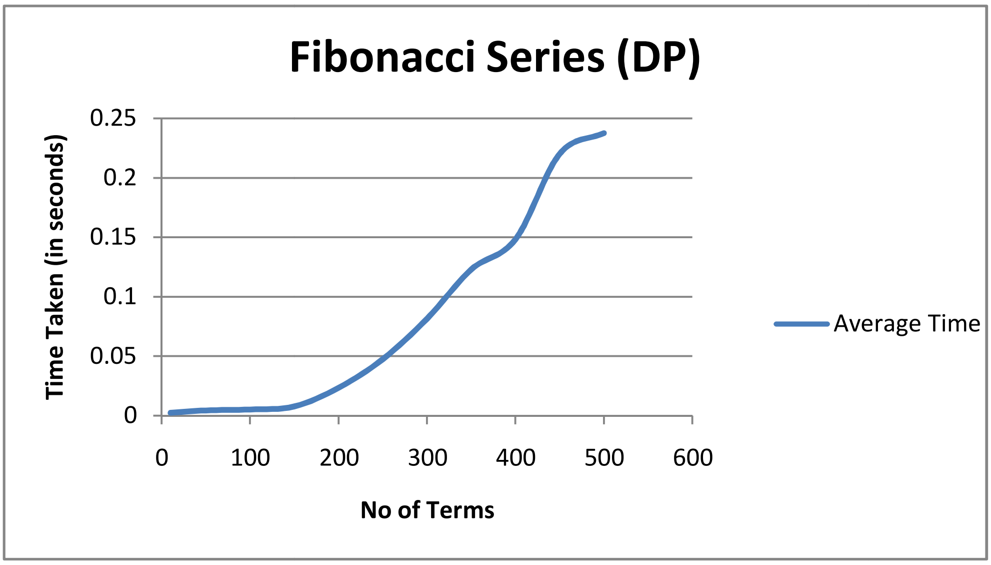

"D:\b.tech_CSE_5th_sem\Analysis and Design of Algorithm\Assignments\1. Fibonacci Analysis\Untidtled2.exe"

```
Enter the number of elements: 100
Fibonacci Series:  0.00 1.00 1.00 2.00 3.00 5.00 8.00 13.00 21.00 34.00 55.00 89.00 144.00 233.00 377.00 610.00 987.00 1
597.00 2584.00 4181.00 6765.00 10946.00 17711.00 28657.00 46368.00 75025.00 121393.00 196418.00 317811.00 514229.00 8320
40.00 1346269.00 2178309.00 3524578.00 5702887.00 9227465.00 14930352.00 24157817.00 39088169.00 63245986.00 102334155.0
0 165580141.00 267914296.00 433494437.00 701408733.00 1134903170.00 1836311903.00 2971215073.00 4807526976.00 7778742049
.00 12586269025.00 20365011074.00 32951280099.00 53316291173.00 86267571272.00 139583862445.00 225851433717.00 365435296
162.00 591286729879.00 956722026041.00 1548008755920.00 2504730781961.00 4052739537881.00 6557470319842.00 1061020985772
3.00 17167680177565.00 27777890035288.00 44945570212853.00 72723460248141.00 117669030460994.00 190392490709135.00 30806
1521170129.00 498454011879264.00 806515533049393.00 1304969544928657.00 2111485077978050.00 3416454622906707.00 55279397
00884757.00 8944394323791464.00 14472334024676220.00 23416728348467684.00 37889062373143904.00 61305790721611584.00 9919
4853094755488.00 160500643816367070.00 259695496911122560.00 420196140727489660.00 679891637638612220.00 110008777836610
1900.00 1779979416004714000.00 2880067194370816000.00 4660046610375530500.00 7540113804746346500.00 12200160415121877000
.00 19740274219868226000.00 31940434634990100000.00 51680708854858326000.00 83621143489848426000.00 135301852344706760000
0.00 218922995834555200000.00

 time taken by Fibonacci function = 0.006000 seconds

Process returned 0 (0x0)   execution time : 33.481 s
Press any key to continue.
```

Time Analysis:

No of Terms	Time Taken during 10 different times										Average Time
10	0.003	0.003	0.003	0.002	0.002	0.002	0.002	0.003	0.003	0.002	0.0025
50	0.005	0.004	0.004	0.005	0.005	0.005	0.004	0.004	0.005	0.004	0.0045
100	0.005	0.005	0.005	0.004	0.007	0.005	0.006	0.005	0.005	0.006	0.0053
150	0.009	0.008	0.008	0.007	0.007	0.007	0.009	0.007	0.009	0.007	0.0078
200	0.024	0.03	0.017	0.017	0.016	0.021	0.024	0.025	0.025	0.036	0.0235
250	0.052	0.08	0.035	0.035	0.039	0.04	0.04	0.07	0.046	0.039	0.0476
300	0.14	0.063	0.091	0.116	0.062	0.064	0.064	0.101	0.059	0.055	0.0815
350	0.083	0.093	0.185	0.09	0.11	0.137	0.086	0.099	0.163	0.183	0.1229
400	0.157	0.175	0.171	0.118	0.168	0.119	0.198	0.12	0.135	0.121	0.1482
450	0.277	0.159	0.318	0.27	0.337	0.137	0.163	0.248	0.15	0.141	0.22
500	0.296	0.272	0.177	0.181	0.38	0.194	0.23	0.176	0.285	0.183	0.2374

Graph Plot:


Fibonacci Series (DP)
Time Taken (in seconds)
No of Terms
Average Time