

Practical - 2

Implement functions to print nth Fibonacci number using iteration and recursive method. Compare the performance of two methods by counting number of steps executed on various inputs. Also draw a comparative chart. (Fibonacci series 1, 1, 2, 3, 5, 8..... Here 8 is the 6th Fibonacci number)

Code:

```
#include<stdio.h>
int count_it=0;
int count_rec=0;
void itfibonacci(int n)
{
    int first=0,second=1,third,i; count_it++;
    for(i=2;i<=n;i++){
        count_it++; third=first+second; count_it++;
        first=second;
        count_it++;
        second=third;
        count_it++;
    } printf("%d",third); count_it++;
}
int recfibonacci(int n){ if(n<=1){ count_rec++; count_rec++;
    return n;
}
Else
{
    count_rec++;
    return recfibonacci(n-1)+recfibonacci(n-2);
} }
void main()
{
    printf("21012021003_AMIT GOSWAMI\n");
    int n,i;
    printf("Enter the number of element : ");

    scanf("%d",&n);

    printf("\nFibonacci series using iteration");
    itfibonacci(n); printf("\nCount(iteration) : %d",count_it);
    printf("\n\nFibonacci series using recursion : ");

    printf("%d",recfibonacci(n));

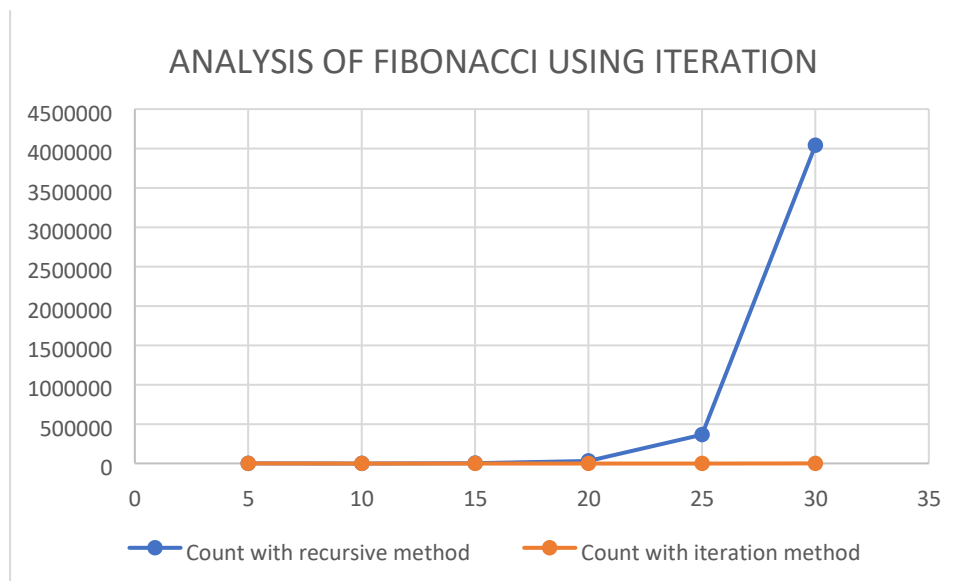
    printf("\n\nCount(recursion) : %d",count_rec); }
```

Output:

21012021003_AMIT GOSWAMI Enter the number of element : 5 Fibonacci series using iteration : 5 Count(iteration) : 18 Fibonacci series using recursion : 5 Count(recursion) : 23	21012021003_AMIT GOSWAMI Enter the number of element : 10 Fibonacci series using iteration : 55 Count(iteration) : 38 Fibonacci series using recursion : 55 Count(recursion) : 266
21012021003_AMIT GOSWAMI Enter the number of element : 15 Fibonacci series using iteration : 610 Count(iteration) : 58 Fibonacci series using recursion : 610 Count(recursion) : 2960	21012021003_AMIT GOSWAMI Enter the number of element : 20 Fibonacci series using iteration : 6765 Count(iteration) : 78 Fibonacci series using recursion : 6765 Count(recursion) : 32837
21012021003_AMIT GOSWAMI Enter the number of element : 25 Fibonacci series using iteration : 75025 Count(iteration) : 98 Fibonacci series using recursion : 75025 Count(recursion) : 364178	21012021003_AMIT GOSWAMI Enter the number of element : 30 Fibonacci series using iteration : 832040 Count(iteration) : 118 Fibonacci series using recursion : 832040 Count(recursion) : 4038806

Analysis:

Position of number	Count with recursive method	Count with iteration method
5	23	18
10	266	38
15	2960	58
20	32837	78
25	364178	98
30	4038806	118

**Conclusion:**

From the above practical we got to know that recursion is a fast process for smaller numbers, but as we increase the numbers, it can be quite a lengthy method, and to avoid that iteration can be a more reliable method.