TASK:11   Recursion and it's concepts.

The fibonacci number, commonly denoted F(n) form a sequence, called the fibonacci series such that each number is the sum of the two preceding ones starting from 0 and 1. That is.

$$F(0)=0, \quad F(1)=1$$

$$F(n)=F(n-1)+F(n-2), \text{ for } n>1$$

Given, calculate F(n).

e.g:

Input n=2

output:1

explanation : $F(2)=F(1)+F(0)=1+0=1$

algorithm:

1) If n is 0, return 0.

2) If n is 1, return 1.

3) Initialize variables a and b to 0 and 1, respectively

4) for i from 2 to n, calculate the next fibonacci number by setting a=b and b=a+b

5) Return b as the nth fibonacci number.

Program:

```
#include <stdio.h>
int fibonacci (int n){
    if (n<=1){
```

return n;
}
return fibonacci(n-1) + fibonacci(n-2);
}
int main() {
    int n=10;
    printf("The 1·d'th Fibonacci number is //·d\n", n, fibonacci(n));
    return 0;
}

output:

The 10th Fibonacci number is 55

problem: The Fribonacci sequence Tn is defined as follows:

$$T_0 = 0, T_1 = 1, T_2 = 1 \text{ and } T_{n+3} = T_n + T_{n+1} + T_{n+2} \text{ for } n \geq 0$$

Given n, return the value of T_n

example:

Input: n = 4
output: 4
explanation:

$$T_3 = 0 + 1 + 1 = 2$$

$$T_4 = 1 + 1 + 2 = 4$$

The answer is guaranteed to fit within a 32-bit in

i·e; answer <= $2^{31} - 1$.

Algorithm:

1. If n is 0, return 0
2. If n is 1 or 2, return 1
3. Initialize variables $t_0 = 0$, $t_1 = 1$, $t_2 = 1$ and $t_3 = 2$
4. Loop from 3 to n, and all at each iteration calculate the value of the next term in the sequence by setting $t_3 = t_0 + t_1 + t_2$, then update the values.
5. Return the value of t_3.

Program:

```c
#include <stdio.h>
int fibonacci(int n) {
    if (n <= 0) {
        return n;
    }

    int main() {
        int n = 10;
        printf("the 10th Fibonacci number is %d\n", n)
        fibonacci(n);
        return 0;
    }
```

Output:

enter the no. of elements: 15

0 1 1 2 3 5 8 13 21 34 55 89 144 233 377.

VEL TECH - CSE	
EX.NO.	
PERFORMANCE (5)	
RESULT AND ANALYSIS (3)	
VIVA VOCE (3)	
RECORD (4)	
TOTAL (15)	
SIGN WITH DATE	

Result: Thus the program is executed and verified successfully.