

Task-4Number Theory Level-1

Aim :- To write a program to find factorial of the numbers.

Algorithm

Read the integer N

Initialize a variable called factorial to 1.

Initialize an array called digits of size 10 too.

• Loop from 1 to N and for each iteration, multiply the current factorial with the loop index.

Convert the factorial to a string.

Loop through each character of the string to integer b. Increment the corresponding element of the digits array.

Count the number of non-zero elements in the digits array.

Print the count of non-zero elements as the result.

End the program.

Task 4: Number Theory – Level 1 (Includes Factorial, Fibonacci Series, Odd or Even, Sum of Digits, ...)

rogram:

```
include <stdio.h>
```

```
int countDigitsInFactorial(int n); int
```

```
main() {
```

```
    int n;
```

```
    printf("Enter a positive integer: "); scanf("%d",  
    &n);
```

```
    int count = countDigitsInFactorial(n);
```

```
    printf("Number of digits in %d! = %d\n", n, count); return  
    0;
```

```
int countDigitsInFactorial(int n) {
```

```
    if (n < 0) { return 0;
```

```
    } if (n <= 1)
```

```
    { return 1;
```

```
    } double digits =  
    0;
```

```
    for (int i = 2; i <= n; i++) { digits  
        += log10(i);
```

```
    }
```

```
    return
```

18/25

Task - 4b

Given a number positive number N , find value of $f_0 + f_1 + f_2 + \dots + f_N$ where f_i indicates i^{th} Fibonacci number. Remember that $f_0 = 0, f_1 = 1, f_2 = 1, f_3 = 2, f_4 = 3, f_5 = 5, \dots$. Since the answer can be very large, answer modulo 1000000007 should be returned.

Algorithm

- Read the positive integer N .
- Initialize variables f_0 and f_1 to 0 and 1 respectively.
- Initialize a variable sum to f_0 .
- Loop from 1 to N and for each iteration:
 - a. Calculate the current Fibonacci number f_n as $f_0 + f_1$.
 - b. update the values of f_0 and f_1 as f_1 and f_n respectively
 - c. Add the current Fibonacci number f_n to sum .
- Return sum modulo 1000000007 .
- End the program.

Program:

```
#include <stdio.h>

#define MOD 1000000007

int fibonacciSum(int n) { int prev = 0,
    curr = 1, next, sum = 0, i; for (i = 0; i
    <= n; i++) {
        sum = (sum + curr) % MOD; next
        = (prev + curr) % MOD;
        prev = curr; curr
        = next;
    }
    return sum;
}

int main() { int N;
    scanf("%d", &N);
    printf("%d\n", fibonacciSum(N));
    return 0;
}
```

Output

input:

N=3

Output:

4

VNL TEL-1 - CSE	
EX NO.	4
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	2
RECORD (4)	4
TOTAL (15)	15
DATE	28/11

Result :-

Thus, the program is executed and verified successfully.