

## Simulation of a recursive function:

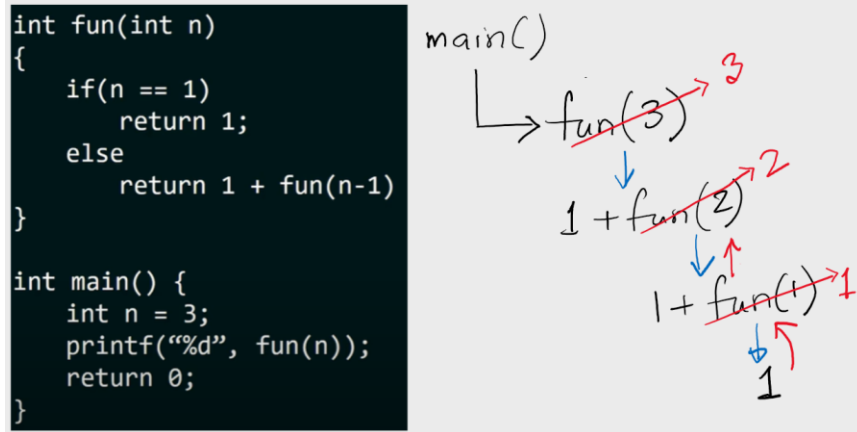


Figure 1: <https://youtu.be/kepBmgvWNDw>

## How to write Recursive Functions

<https://youtu.be/ggk7HbcnLG8>

1. Divide the problem into smaller subproblem
2. Specify the base condition to stop the recursion

### Divide the problem into smaller subproblem

```

Fact (1) = 1
Fact (2) = 2. 1 = 2. Fact (1)
Fact (3) = 3. 2. 1 = 3. Fact (2)
Fact (4) = 4. 3. 2. 1 = 4. Fact (3)
...
Fact (n) = n. Fact(n-1)

```

### Specify the base condition to stop the recursion

The base case returns a value without making any subsequent recursive calls. For example: `Fact (1) = 1`

### In summary:

<b>Function</b> rec: <b>if</b> base condition: ... <b>else</b> ... recursive formula	<b>Function</b> Fact(n) : <b>if</b> n==1 <b>then</b> <b>return</b> 1 <b>else</b> <b>return</b> n*Fact(n-1)
--	--

### Why Stack Overflow error occurs in recursion?

If the base case is not reached or not defined, then the stack overflow problem may arise. In the following code, Fact (5) or Fact (1) will cause stack overflow.

```
Function Fact(n) :
    if n==10 then
        return 10
    else
        return n*Fact(n-1)
```

### Practice problems:

Instructions:

1. Do not adopt unfair means. 10 marks will be deducted from the final marks for adopting unfair means.
2. No more than 40% marks for uncompileable codes.

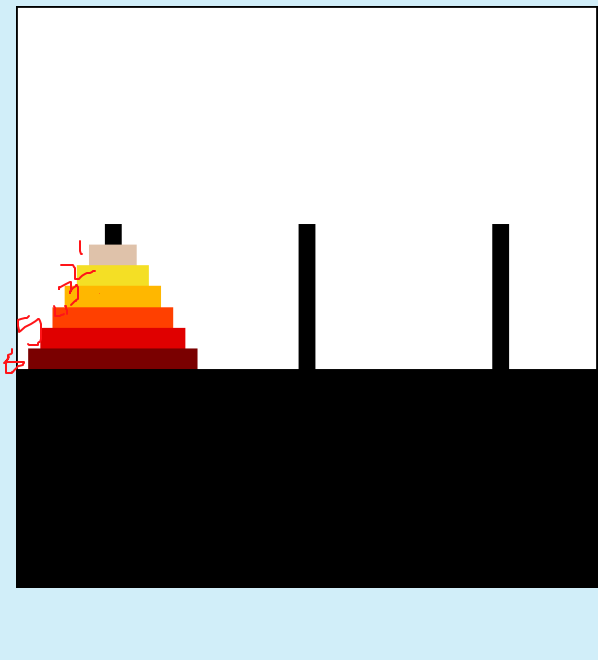
1. Write a recursive implementation of the factorial function.  
Recall that  $n! = 1 \times 2 \times \dots \times n$ , with the special case that  $0! = 1$ .
2. Write a recursive program to print the  $n^{\text{th}}$  Fibonacci number.
3. Write a recursive program to calculate the power of  $x$  ( $x^y$ ), where  $y$  is a non-negative integer.
4. Write a recursive program to count the number of digits of an integer.
5. Write a recursive program to find the sum of digits of an integer.

6. Write a recursive program to find the sum of the elements of an array of size  $n$ .
7. Write a recursive program to find the products of the elements of an array of size  $n$ .
8. Write a recursive program to find the maximum of the elements of an array of size  $n$ .
9. Write a recursive program to find the minimum of the elements of an array of size  $n$ .
10. Write a recursive program to find the average of the elements of an array of size  $n$ .
11. Write a recursive program to count/print the odd/even numbers of an array of  $n$  integers.
12. Write a recursive program to print an array of size  $n$  in reverse order.
13. Write a recursive program to print the even numbers in a given range.

Sample input	Sample output
3 10	4 6 8 10

14. Write a recursive program to check if a given string is a palindrome or not.
15. Write a recursive program to find the maximum of a 2d array.
16. Write a recursive program to count the prime numbers of a given 2d array.

17. Find the sum of the following series up to  $n^{\text{th}}$  position.
- $1 + 2 + 3 + \dots$
  - $1^2 + 2^2 + 3^2 + \dots$
  - $1 * 3 + 2 * 5 + 3 * 7 + 4 * 9 + \dots$
  - $2 * 3 + 4 * 5 + 8 * 7 + 16 * 9 + \dots$
  - $2 * 3 * 4 + 4 * 5 * 3 + 8 * 7 * 2 + 16 * 9 * 1 + \dots$
18. Write a recursive program to find the GCD of  $x$  and  $y$  where  $x, y$  are positive integers.  
(Hint: [use Euclid's algorithm](#). Two ways to solve this.)
19. Write a recursive program to find the LCM of  $x$  and  $y$  where  $x, y$  are positive integers.  
(Two ways to solve this)
20. Write a recursive program to solve the Tower of Hanoi problem for  $n$  disks.

	Sample input	Sample output
	4	Move disk 1 from A to B Move disk 2 from A to C Move disk 1 from B to C Move disk 3 from A to B Move disk 1 from C to A Move disk 2 from C to B Move disk 1 from A to B Move disk 4 from A to C Move disk 1 from B to C Move disk 2 from B to A Move disk 1 from C to A Move disk 3 from B to C Move disk 1 from A to B Move disk 2 from A to C Move disk 1 from B to C

21. Write a recursive implementation of binary search in a sorted array.
22. Given a set of parentheses check if they are balanced or not using a recursive function.
23. Write a recursive program to print all subsets of a set of  $n$  elements.
24. Write a recursive program to print all subsequence of a string.
25. DFS
26. In-order, preorder, postorder
27. Print path from node to the root of a binary tree