Recursion

sandeep@beingzero.in

What is it?

- Function calling itself – directly or indirectly.
- One or more base cases.
- Recursive Call.

Function that returns factorial of N

Expected	Run
factorial(1) → 1	1
factorial(2) → 2	2
factorial(3) → 6	6
factorial(4) → 24	24
factorial(5) → 120	120
factorial(6) → 720	720
factorial(7) → 5040	5040
factorial(8) → 40320	40320
factorial(12) → 479001600	479001600

```
int factorial(int n)
{
        if(n==0)
                return 1;
        return n*factorial(n-1);
}
```

Nth Fibonacci Number

Expected	Run
fibonacci(0) → 0	0
fibonacci(1) → 1	1
fibonacci(2) → 1	1
fibonacci(3) → 2	2
fibonacci(4) → 3	3
fibonacci(5) → 5	5
fibonacci(6) → 8	8
fibonacci(7) → 13	13

```
int fib(int n)
{
        if(n==0)
                return 0;
        if(n==1)
                return 1;
        return fib(n-1) + +fib(n-2);
}
```

Function to return total fingers of N people

- We have people standing in a line, numbered 1, 2, ...
- The odd person (1, 3, ..) have the normal 10 fingers.
- The even people (2, 4, ..) we'll say have 11 fingers, because they each have an extra finger.
- Recursively return the number of “fingers" in the people line 1, 2, ... n

```
int totalFingers(int people)
{
        // Your Logic
}
```

Expected	Run
totalFingers(0) → 0	0
totalFingers(1) → 10	10
totalFingers(2) → 21	21
totalFingers(3) → 31	31
totalFingers(4) → 42	42
totalFingers(5) → 52	52
totalFingers(12) → 126	126
totalFingers(50) → 525	525
totalFingers(234) → 2457	2457

Sum of digits of number

Expected	Run
sumDigits(126) → 9	9
sumDigits(49) → 13	13
sumDigits(12) → 3	3
sumDigits(10) → 1	1
sumDigits(1) → 1	1
sumDigits(0) → 0	0
sumDigits(730) → 10	10
sumDigits(1111) → 4	4
sumDigits(11111) → 5	5
sumDigits(10110) → 3	3
sumDigits(235) → 10	10

```
int sumDigits(int n)
{
  if(n==0)
      return 0;
  return n%10 + sumDigits(n/10);
}
```

Practice Problems

- http://codingbat.com/java/Recursion-1
- http://codingbat.com/java/Recursion-2
- http://www.sparknotes.com/cs/recursion/examples/
- https://www.cs.utexas.edu/~scottm/cs314/handouts/PracticeProblems.htm
- http://www.cs.cornell.edu/courses/CS2110/2014fa/L07-Recursion/recursion_practice.pdf