# MT21MCS013 Jay Chachapara

# Recurision

### Finding the number of calls to a function

In [4]:

```
def function(n : int,counter : int) -> int :
    counter += 1
    if n <= 0 :
        return counter
    if n % 2 == 0 :
        return function(n-1,counter)
    else :
        return function(n-2,counter)
var = int(input("Enter an integer value : "))
times = function(var,0)
print("The function is called",times,"times")
```

```
Enter an integer value : 5
The function is called 4 times
```

### Print binary value of given integer

In [19]:

```
def binary(n : int) -> None :
    if n > 0 :
        #This condition is used to not to print the preceding zeros in the binary v
        if n//2 != 0 :
            binary(n//2)
        print(n%2,end="")
    if n == 0 :
        print("0",end="")

while True :
    try :
        x = int(input("\nEnter an integer : "))
        binary(x)
    except ValueError :
        break
```

```
Enter an integer : 0
0
Enter an integer : 1
1
Enter an integer : 2
10
Enter an integer : 3
11
Enter an integer : 4
100
Enter an integer : 5
101
Enter an integer : 6
110
Enter an integer : 7
111
Enter an integer : 8
1000
Enter an integer : 9
1001
Enter an integer : 10
1010
Enter an integer : 11
1011
Enter an integer : 12
1100
Enter an integer : 13
1101
Enter an integer : 14
1110
Enter an integer : 15
1111
Enter an integer : 16
10000
Enter an integer : 17
10001
Enter an integer : 18
10010
Enter an integer : 19
10011
Enter an integer : 20
```

```
10100
Enter an integer : v
```

In [ ]: