United International University
Department of Computer Science and Engineering
CSI 124: Advanced Programming Laboratory
Lab Work # 09
Summer 2015

---

1 What is Recursion ?

Remember the movie *INCEPTION* ? Where Mr. Cobb used to enter a person's dream and from that person's dream to another person's dream to steal valuable information ? In recursion we will do almost the same thing save for stealing anything !
Recursion is a process by which a function calls itself repeatedly, until some specified condition has been satisfied. Many iterative (loop) problems can be solved using recursion. Have a look at this code:

```
#include<stdio.h>

void spendMoney(int n){
   printf("Your wallet balance: %d\n",n);
   n = n - 5;
   spendMoney(n); /* function calls itself */
}
int main(){
   printf("Hello Mr. COBB!\n");
   spendMoney(10000);
   return 0;
}
```

1. Compile and run this code and see the output.

2 But why doesn't it STOP ?

That was fun, right ? But in our world you will be barely asked to write a program that runs till Armageddon starts. So the billion dollar question is, how do we stop it ? Simple, we need to set a base condition. Its like I will continue to spend money until I have a single penny left in my wallet. So the above code should be like this:

```
#include<stdio.h>

void spendMoney(int n){
   printf("Your wallet balance: %d\n",n);
   n = n - 5;
   if(n < 100) return;
   spendMoney(n); /* function calls itself */
}
int main(){
```

```
   printf("Hello Mr. COBB!\n");
   spendMoney(1000);
   return 0;
}
```

1. Using a recursive function WAP that takes an integer 'n' as input and prints the numbers until n ≥ 0
   Sample input : 4
   Sample output : 4 3 2 1 0

3 Fun continues with Recursion !

Have a look at this code:

```
#include <stdio.h>

int factorial(int i)
{
   if(i <= 1) return 1;
   return i * factorial(i - 1);
}
int main()
{
    int i = 6;
    printf("Factorial of %d is %d\n", i, factorial(i));
    return 0;
}
```

1. Compile and run this code and see the output.
2. WAP that takes an integer 'n' as input and outputs the sum of integers until n ≥ 0
   Sample input : 4
   Sample output : 10
3. WAP that prints a string in reverse order.
   Sample input : abcd
   Sample output : dcba

4 Practise Matches :)

1. WAP that takes two integers as parameters and outputs their greatest common divisor (GCD)
2. WAP that takes integer a and integer b as input and outputs a % b (a mod b)
3. WAP that takes an integer 'n' as input and prints the 'n'th term of the fibonacci series.
   Sample input : 5
   Sample output : 5
   fibonacci series : 0 1 1 2 3 5 8 ....